From: Tom Hastings
Date: 8/11~~07~~/99
File: ipp-notification-new-model-990811~~07~~.doc

This document is a new proposal to simplify the IPP Notification Model. It uses Harry Lewis's suggested terminology of Per-Job subscriptions and Per-Printer subscriptions. The Subscription object is defined only for use as Per-Printer subscriptions. Per-Job subscriptions are done entirely with Job attributes and use an exclude event mask (like SNMPv3 trap subscriptions) in order to allow different recipients to get different events. I've also added conformance requirements for operations and attributes, so that we can more properly reason about the model, before updating the specification document.

This document also includes the results of the 8/11~~4~~/99 IPP telecon.

There are 14 issues highlighted like this that need to be resolved.

This paper uses object modeling techniques to describe the current IPP Notification Specification object model (July 22, 1999):

    ftp://ftp.pwg.org/pub/pwg/ipp/new_NOT/ipp-notification-990722.pdf

Review of this object model before writing the complete spec will help speed up the production of the spec and will also help with the understanding of the semantics. I suggest that this modeling be kept in the final spec in some form similar to this paper.

This paper also gives an outline of how the final spec might be presented, including being divided into two parts. After describing the additional object attributes for the Job, Printer, and Subscription objects, the order of the spec is suggested to be:

1. the REQUIRED first part (sections 5, 6, and 7) defines the extensions to the Job and Printer object and its operations for Per-Job subscriptions of Job and Printer events and their notification content.

2. the OPTIONAL second part (section 8) defines the Per-Printer subscription mechanism and the Subscription object and its operations and attributes.

NOTE - Throughout this model document, the notation [ ] indicates that the sender (client in a request, IPP object in a response, Notification Sender in a notification) MAY omit the attribute inside the square brackets if there is not data to send. In addition, IPP Printer conformance requirements for operation and attribute support are indicated in this document with the words and codes:

REQUIRED (R) to support notification at all
CONDITIONALLY REQUIRED (C) - REQUIRED if Per-Printer subscription is supported
OPTIONAL (O) - not required for either Per-Job or Per-Printer Subscrpitions.

Therefore, the [ ] do NOT mean OPTIONAL support and should not be confused with support requirements.

36

Table of Contents

71

1   Current IPP Notification Operations

The current IPP Notification specification (July 22, 1999) consists of the Job object and the Printer object.
A Job object contains zero or one Per-Job Subscription which specifies Job and/or Printer events. The Job
events apply only to this Job. The Printer events apply no matter which job. A Printer object contains zero
or more Per-Printer Subscriptions which specify Job and/or Printer events. The Job and Printer events
apply no matter which job.

The following operations are defined for Per-Job Subscriptions and Per-Printer Subscriptions:

1.1   Per-Job Subscriptions - previous model

Job object related:	salient inputs besides printer-uri:
job creation operations	recipients, [events,] [user-data]
Subscribe-Job	job-id, recipients, [events]
Get-Job-Attributes	job-id, [requested-attributes]
Unsubscribe-Job	job-id, subscription-id

1.2   Per-Printer Subscriptions - previous model

Printer object related:	salient inputs beside printer-uri:
Get-Printer-Attributes	[requested-attributes]
Renew-Printer-Subscription	[lease time]
Get-Printer-Subscriptions	[my-subscriptions]

2   New IPP Notification Operations

This new proposed IPP object model add the Subscription object (to the Job and Printer objects), but only
for the Per-Printer subscriptions.

The following operations are defined for Per-Job Subscriptions and Per-Printer Subscriptions:

2.1   Per-Job Subscriptions - new model

Job object related:	salient inputs besides printer-uri
job creation operations	recipients, [events,] [user-data,] [exclude-event-mask]
Get-Job-Attributes	job-id, [requested-attributes]
~~Replace~~Set-Job-Subscription	job-id, [recipients,] [events,] [user-data,] [exclude-event-mask]

86

87 ## *2.2 Per-Printer Subscriptions - new model*

Printer object related:	salient inputs beside printer-uri:
Get-Printer-Attributes	[requested-attributes]
Subscription object related:	salient inputs besides printer-uri:
Create-Printer-Subscription	recipients, [events,] [user-data,] [exclude-event-mask,] [lease-time-requested]
Get-Printer-Subscription-Attributes	subscription-id, [requested-attributes]
Get-Printer-Subscriptions	[my-printer-subscriptions,] [requested-attributes]
Renew-Printer-Subscription	subscription-id, [lease time-requested]
Cancel-Printer-Subscription	subscription-id

88 ISSUE 1 - If IPP is generalized to multi-function devices, will we regret having "Printer" in the names of
89 these operations that could work for any type of device?

90 # 3   Job, Printer, and Subscription object relationships

91 This section describes the new notification object model that adds a Subscription object which together
92 with the Job and Printer object provide the complete notification semantics. The Subscription object is
93 OPTIONAL. The Subscription object MUST be supported if Per-Printer subscriptions are supported and
94 vice-versa.

95 Per-Job subscriptions do not use the Subscription object. Instead, a Job can have one Per-Job subscription
96 which is represented entirely as Job attributes. A "notify-event-exclude-mask" attribute allows some of the
97 specified events to be excluded from some of the specified Notification Recipients in the Job subscription.

98   The object relationships can be seen pictorially as:

```
99
100        Subscription objects (Per-Printer Subscriptions)          Printer object
101        +----+                                                  +------------+
102        | s1 |<------------------------------------------------>|            |
103        +----++                                                 |            |
104         | s2 |<------------------------------------------------>|     p1     |
105         +----++                                                |            |
106          | s3 |<------------------------------------------------>|            |
107          +----+                                                  +------------+
108                                Job objects
109                            +---------+
110                            |         |
111                            |    j1   |
112                            |         |
113                            +----+    |
114                            | s4 |    |        s4 is a Per-Job subscription
115                            ++---+----++
116                             |        |
117                             |   j2   |
118                             |        |
119                            +----+    |
120                            | s5 |    |        s5 is a Per-Job subscription
121                            ++---+----++
122                             |        |
123                             |   j3   |
124                             |        |
125                             |        |
126                            +---------+
```
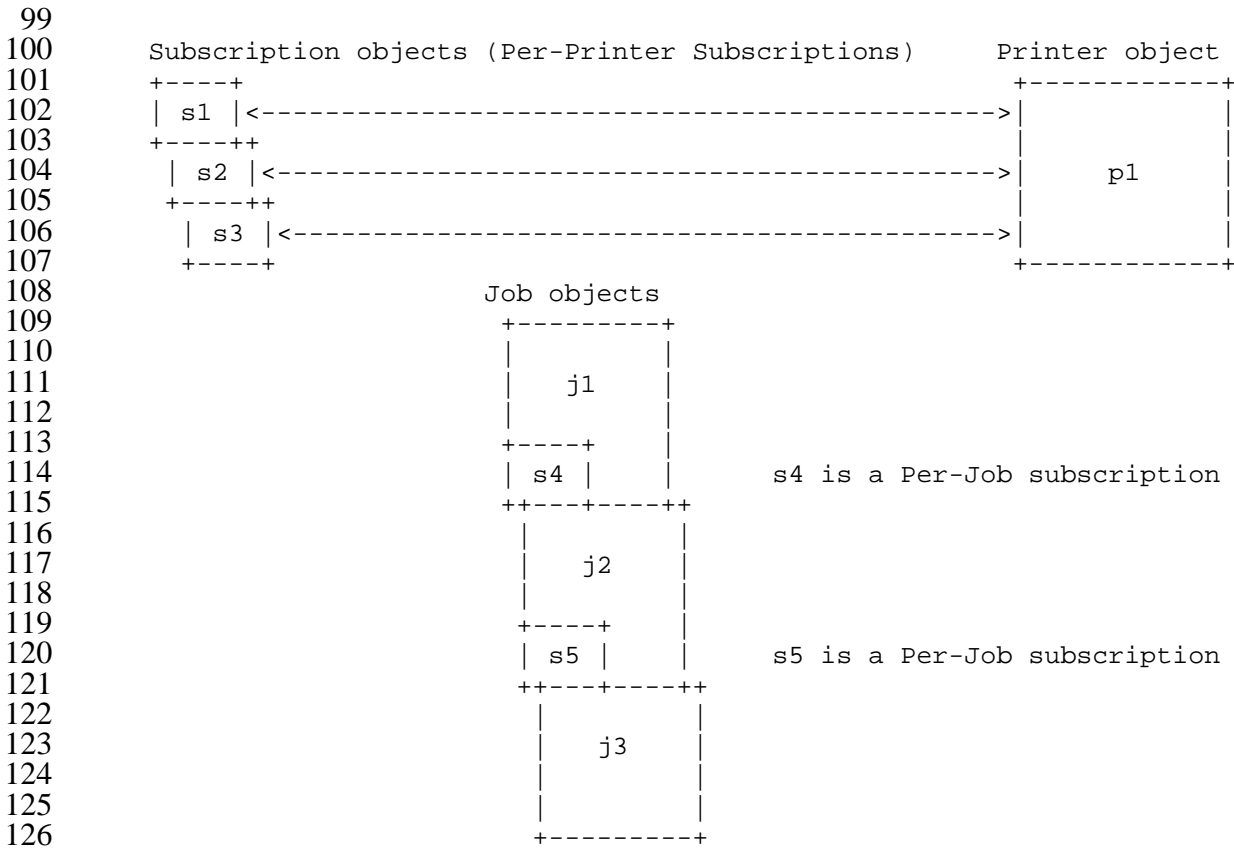
127   s1, s2, and s3 are Per-Printer Subscription objects and can identify Printer and/or Job events.

128   s4 and s5 are Per-Job subscriptions (not objects) and can identify Printer and/or Job events.

3.1   *Object relationships*

130   The object relationships can be stated as follows:

131   1. The Printer object contains zero or more Subscription objects (p1 contains s1-s3 Subscription
132   objects).

133   2. Each Subscription object (s1, s2, and s3) is contained in one Printer object (p1) and represents one
134   Per-Printer subscription.

135   ~~Each Subscription object (s1, s2, and s3) represents one Per-Printer subscription.~~

136   3. Each "Per-Printer" Subscription object identifies one or more Job and/or Printer events. Such Job
137   events are for *all* jobs on the Printer. Such Printer events are for *any* Printer event, no matter which
138   job is processing and when no jobs are processing.

139   4. The "Per-Job" subscription (not an object by itself, but is attributes of the Job object) identifies one
140   or more Job and/or Printer events. Such Job events are only for *this* job (different than "per-Printer"

141 Subscriptions). Such Printer events are for *any* Printer event, no matter which job and when no jobs
142 are processing (same as for "per-Printer" Subscriptions).
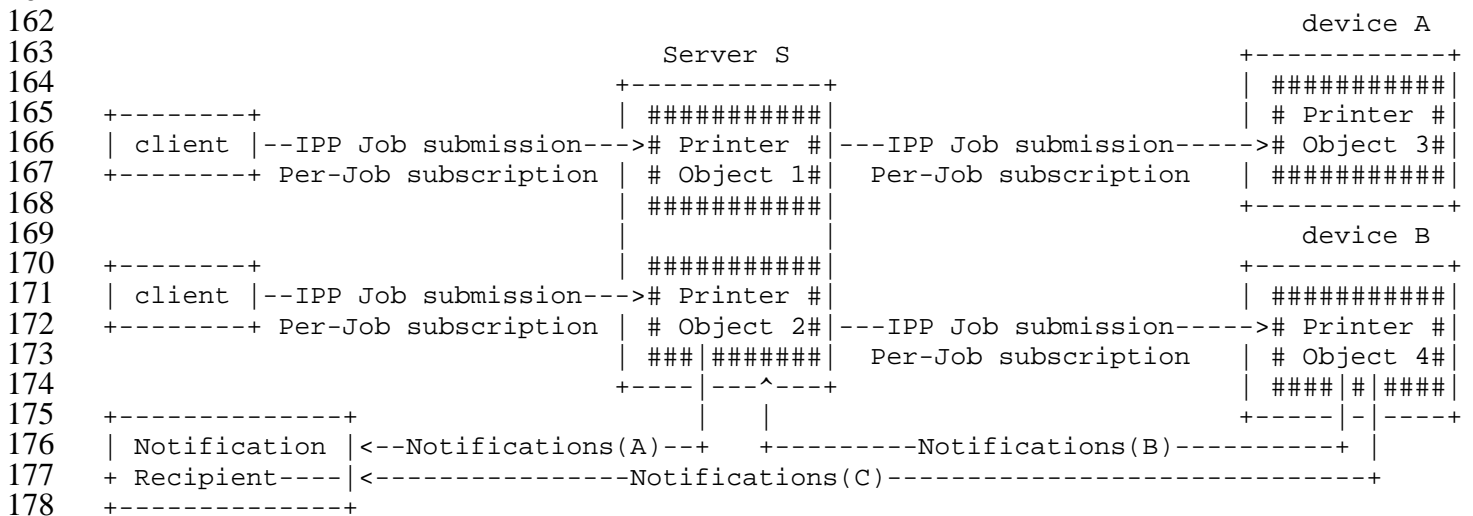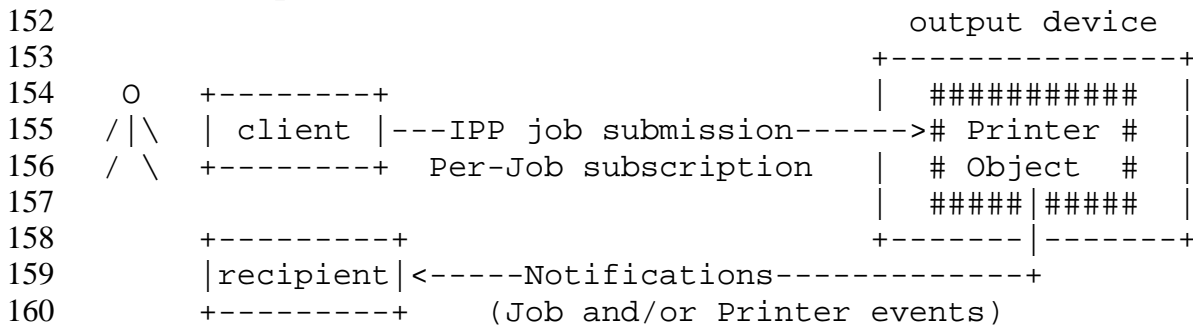
143   5. A Job object contains zero or one Per-Job subscriptions (not an object). (Job j1 contains Per-Job
144   subscription s4, Job j2 contains Per-Job subscription s5, and Job j3 does not contain a Per-Job
145   subscription.

146   6. A Subscription object cannot be contained in or associated with more than one Printer object.

3.2 Notification Configurations for Per-Job Subscriptions

148 This section describes several configurations of IPP Printer objects that are intended to be able to make use
149 of Per-Job subscriptions. Per-Printer Subscriptions may be used in addition, but are not shown to simplify
150 the pictures and understanding.

```
embedded printer:
                                           output device
                                       +---------------+
  O    +--------+                      |  ##########   |
 /|\   | client |---IPP job submission------># Printer #   |
 / \   +--------+  Per-Job subscription   |  # Object  #   |
                                       |  #####|#####   |
       +---------+                     +-------|-------+
       |recipient|<-----Notifications-------------+
       +---------+    (Job and/or Printer events)


                                                              device A
                            Server S                      +------------+
                          +------------+                  | ##########|
+--------+                | ##########|                  | # Printer #|
| client |--IPP Job submission--->#  Printer #|---IPP Job submission----># Object 3#|
+--------+ Per-Job subscription | # Object 1#   Per-Job subscription   | ##########|
                          | ##########|                  +------------+
                          |            |                     device B
+--------+                | ##########|                  +------------+
| client |--IPP Job submission--->#  Printer #|                  | ##########|
+--------+ Per-Job subscription | # Object 2#|---IPP Job submission----># Printer #|
                          | ###|#######|   Per-Job subscription   | # Object 4#|
                          +----|---^---+                  | ####|#|####|
+--------------+              |   |                       +-----|-|----+
| Notification |<--Notifications(A)--+   +---------Notifications(B)----------+ |
+ Recipient----|<---------------Notifications(C)----------------------------+
+--------------+
```

179 The first configuration show a simple Client - Printer relationship. The Printer delivers the Notifications to
180 the Notification Recipient(s) supplied by the Client.

181 The second configuration shows a (spooling or non-spooling) Server that implements two Printer objects (1
182 and 2) that represent two devices that in turn each implement an IPP Printer object (3 and 4, respectively).
183 The server implements one IPP Printer object (1 and 2) for each such device. The Server implementation
184 has three choices for how to support Per-Job subscriptions to the client (and itself):

6

185   1. forward the Per-Job subscriptions to the down stream IPP Printer and let it perform the
186   notification directly to the Notification Recipients supplied by the Client (Notifications(C)) and
187   use Per-Printer Subscriptions for the Server's own purposes.

188   2. save the client-supplied Per-Job subscription on the Job object in the server and substitute its
189   own Per-Job subscription with the Server as the Notification Recipient (Notifications(B)). Then
190   the Server relays Notifications to the client-supplied Notification Recipients (Notifications(A)).

191   3. A combination of 1 and 2 in which the Server adds its own Per-Job subscriptions to those
192   supplied by the client. Thus the IPP Job that goes to Printer object 4 has a combination of
193   subscription information from both the Client and the Server. This latter approach is sometimes
194   called "piggy-backing" because the Server is adding its Per-Job subscription information to that
195   supplied by the client. Piggy-backing is especially useful, if device B also accepts (IPP or non-
196   IPP) requests from other servers. Then when all the jobs from Server S have been completed by
197   device B, there will be no more Job events sent to Server S. (Server S could still maintain a
198   long term Per-Printer subscription with Printer D to that Server S can have Printer B's state track
199   (shadow) that of Printer D or Server S could poll Printer D when queried about Printer B).

200 4   Object attributes defined for Notification

201 The following notification attributes are defined for each of the Printer, Job, and (new, but OPTIONAL)
202 Subscription objects. The explanations about some of the object attributes are collected here so that they
203 can be referred to from the operations that set them. All Job, Printer, and Subscription object attributes are
204 READ-ONLY, i.e., they MUST NOT be settable with the Set-Job-Attributes or Set-Printer-Attributes
205 operations. Instead, all of them are set as a side effect of another operation or when an event notification is
206 delivered. Attributes and objects not in the current, July 22, spec are labeled as "new":

207 4.1.1 Job object attributes related to notification

Job object attributes:	Printer support	READ-ONLY, set by:
notify-recipients (1setOf uri)	R	job creation operation
notify-events (1setOf type2 keyword)	R	job creation operation
notify-user-data (octetString(63255))	R	job creation operation
notify-exclude-event-masks (1setOf octetString(8))	RO	job creation operation
notify-natural-languages" (1setOf naturalLanguage)	O	job creation operation

208

	Printer support	
job-trigger-event (type2 keyword)	R	Printer - event occurring
job-trigger-message (text(255))	R	Printer - event occurring
job-trigger-time (integer(MIN:MAX))	R	Printer - event occurring
job-trigger-date-time (dateTime)	O	Printer - event occurring
notify-request-ids (1setOf integer(0:MAX))	C	Printer - event occurring
previous-job-state (type1 enum)	R	Printer - event occurring
job-state-reasons-added (1setOf type2 keyword)	R	Printer - event occurring
job-state-reasons-deleted (1setOf type2 keyword)	R	Printer - event occurring

213 Attribute Notes:

214 "notify-user-data" (octetString(63255)) - information being sent from the Subscriber to the Notification
215   Recipient(s), such as the identify of the Subscriber or a path or index to some Subscriber
216   information or a key that the Notification Recipient needs in order to process the Notification. This
217   information is the same for all Notification Recipients. Per-Notification Recipient data should be
218   specified in the URL registration so that they may be sent as parameters in each URLcould identify
219   the real recipient, in case the Notification Recipient is a general recipient tool, such as Tool Talk or
220   an Instant Message Service.
221   ISSUE 1 - is 255 octets too long for "notify-user-data (octetString(255))", since it also has to be sent
222   in the notification content? How about 64 octets?

223 "notify-exclude-event-masks" (1setOf octetString(8)) - an OPTIONAL REQUIRED per-recipient event
224   mask whose values are parallel to the "notify-recipients" attribute. Each 1 bit indicates that the
225   corresponding recipient is not to receive the 'i'-th event specified in the "notify-events" attribute.
226   The first event in "notify-events" is numbered '1' which corresponds to the first bit (Big Endian
227   network order) of the event mask octetString. If an implementation only supports n events, then the
228   maximum number of octets that an implementation need support for this attribute is not really 8
229   octets, but is n mod 8.
230   The exclude event mask concept is also used in SNMPv3 trap registration (RFC2573].ISSUE 2 -
231   Should "notify-exclude-event-masks (1setOf octetString(8))" be REQUIRED if the number of
232   Notification Recipients supported in a Per-Job (or Per-Printer) Subscription in greater than 1 (as
233   specified in the Printer's "max-notification-recipients-supported" attribute)?

234 "notify-natural-languages" (1setOf naturalLanguage) - an OPTIONAL per-recipient natural language
235   specification whose values are parallel to the "notify-recipients" attribute.

238 "job-trigger-event" (type2 keyword) - the last Job event that was sent in a Notification for this Job's Per-Job
239   subscription. A value of 'none' is used if the job has not yet had a Job event Notification sent.

240 "job-trigger-time" (integer(MIN:MAX)) - the time (using the "printer-up-time" form) that the last Job event
241   was sent in a Notification for this Job's Per-Job subscription. A value of 0 is used if the job has not
242   yet had a Job event Notification sent.

243 "job-trigger-date-time" (dateTime) - the date-time that the last Job event was sent in a Notification for this
244   Job's Per-Job subscription. A value of 0 is used if the job has not yet had a Job event Notification
245   sent.

246 "notify-request-ids" (1setOf integer(0:MAX)) - the most recent request-ids delivered in a Job Notification
247   content for a Per-Job Subscription for this Job for each of the Job's Notification Recipients. In other

248 words, the "notify-request-ids" is a parallel attribute to the Job's "notify-recipients (1setOf uri)
249 attribute. A value of 0 indicates that no Notifications have been sent for this Job. The first request-
250 id sent for a job MUST be 1. The reason to have a separate request-id for each recipient is so that
251 when more than one Notification Recipient is specified in a subscription, each recipient will have its
252 own monotonically increasing set of request-ids, i.e., no gaps, in order to be able to detect a missing
253 notification

254 job-state-reasons-added (1setOf type2 keyword) and job-state-reasons-deleted (1setOf type2 keyword) -
255 ISSUE 3 - why not just "previous-job-state-reasons" instead of "job-state-reasons-added" and "job-
256 state-reasons-deleted" attributes as Job Description attributes?

257 ### 4.1.2 Printer object attributes related to notification

Printer object attributes:	Printer support	READ-ONLY, set by:
notify-schemes-supported (1setOf uriScheme)	R	Administrator/implementation
max-notification-recipients-supported (integer(1:MAX))	R	Administrator/implementation
notify-events-default (1setOf type2 keyword)	R	Administrator/implementation
notify-events-supported (1setOf type2 keyword)	R	Administrator/implementation
max-recipients-supported (integer(0:MAX))	R	Administrator/implementation
max-events-supported (integer(5:MAX))	R	Administrator/implementation
notify-lease-time-supported (rangeOfInteger(0:MAX))	C	Administrator/implementation
notify-lease-time-default (integer(0:MAX))	C	Administrator/implementation
max-printer-subscriptions-supported (integer(0:MAX))	C	Administrator/implementation
persistent-subscriptions-supported (boolean)	O	Administrator/implementation

258 Attribute Notes:

259 "notify-schemes-supported" (1setOf uriScheme) - the schemes that are supported in Per-Job and Per-Printer
260 "notify-recipients" operation attributes.

261 "max-notification-recipients-supported" (integer(1:MAX)) - the maximum number of Notification
262 Recipients that are supported in a Per-Job Subscription and in each . If Per-Printer Subscriptions, if
263 supportedare supported this same number applies. A value of 0 indicates no effective maximum.
264 An implementation MUST support at least 31 Notification Recipients per subscription, so that
265 clients can depend on at least that number of recipients in a single subscription. If notification is not
266 supported, this attribute MUST NOT be supported.

267 "max-events-supported" (integer(05:MAX)) - the maximum number of events that are supported in a Per-
268 Job subscription and in each Per-Printer Subscription object, if supported. A value of 0 indicates no
269 effective maximum. If Per-Printer subscriptions are supported this same number applies. An
270 implementation MUST support at least 5 events per subscription, so that clients can depend on at
271 least 5 events in a single subscription. If notification is not supported, this attribute MUST NOT be
272 supported.

273   "notify-lease-time-supported" (rangeOfInteger(0:MAX)) - the minimum and maximum lease time values
274   supported for "notify-lease-time-requested" operation attributes. When the lease time expires for a
275   Per-Printer Subscription without renewing, the Printer MUST delete the Subscription object. A
276   value of 0 indicates an infinite amount of time, i.e., no lease. No lease MAY require authorization.

277   "max-printer-subscriptions-supported" (integer(0:MAX)) - the maximum number of Per-Printer
278   subscription supported. 0 means there is no limit, if Per-Printer subscriptions are supported. An
279   implementation MUST support at least 8 Per-Printer Subscription object instances, if it supports
280   Per-Printer subscriptions, so that clients can depend on at least 8 Per-Printer Subscriptions. If
281   notification is not supported, this attribute MUST NOT be supported.

282   "persistent-subscriptions-supported" (boolean) - indicates whether or not the Printer supports persistent Per-
283   Printer Subscriptions, i.e., subscriptions that are preserved across power cycles. If the value is
284   'false' or that attribute is not supported, Per-Printer Subscriptions MUST NOT be persistent.

285   ## 4.1.3 Subscription object attributes (new and OPTIONAL)

Subscription object attributes:	Printer support	READ-ONLY, set by:
notify-recipients (1setOf uri)	C	client - Create-Printer-Subscription
notify-events (1setOf type2 keyword)	C	client - Create-Printer-Subscription
notify-user-data (octetString(255))	C	client - Create-Printer-Subscription
notify-exclude-event-mask (1setOf octetString(8))	RO	client - Create-Printer-Subscription
notify-natural-languages (1setOf naturalLanguage)	O	client - Create-Printer-Subscription

286

subscription-id (integer(1:MAX))	C	Printer - Create-Printer-Subscription
notify-lease-time-remaining (integer(0:MAX))	C	Printer - Create-Printer-Subscription
notify-lease-time-granted (integer(0:MAX))	C	client Printer - Create-Printer-Subscription
subscriber-user-name (name(MAX))	C	Printer - Create-Printer-Subscription

287

most-recentnotify-request-ids (1setOf integer(0:MAX))	C	Printer - event occurring
delivery-failure-count (integer(0:MAX))	C	Printer - event occurring

288   ISSUE 4 - It isn't necessary to add a "persistence (boolean)" attribute to the Subscription object in order to
289   record the value passed in the Create-Printer-Subscription operation and to allow it to be queried?

290   Attribute Notes:

291   "notify-user-data" (octetString(255)) - same as the "notify-user-data" Job Description attribute (see section
292   4.1.1).

293   "notify-exclude-event-mask" (1setOf octetString(8)) - same as the "notify-exclude-event-mask" Job
294   Description attribute (see section 4.1.1).

295 "notify-natural-languages (1setOf naturalLanguage)" - same as the "notify-natural-languages" Job
296   Description attribute (see section 4.1.1).

300 "subscription-id (integer(1:MAX)) - the subscription id of this Subscription object instance.

301 "notify-lease-time-remaining" (integer(0:MAX)) - the time in the future when the subscription lease will
302   expire, i.e., the "printer-up-time" value number of seconds remaining inat which the lease will
303   expire. When the Printer counts the value down to 0indicated time arrives, the Printer MUST delete
304   the Per-Printer Subscription object. A client is able to extend a lease using the Renew-Printer-
305   Subscription operation (see section 8.4). Internally, the implementation may have a more absolute
306   representation of when the least expires, but this relative value is returned to help the client know
307   how much more time is remaining in the lease. A value of 0 indicates an infinite nametime, if such
308   a policy is supported as indicated in the "notify-lease-time-supported (integer(0:MAX)) Printer
309   Description attribute and the subscriber is authorized to request an infinite lease.
310
311   Note: In order to compute the number of seconds remaining in a Per-Printer Subscription lease, a
312   client can subtract the "printer-up-time" Printer object attribute from the "notify-lease-time"
313   Subscription object attribute.

316 "notify-lease-time-granted" (integer(0:MAX)) - the number of seconds granted for the subscription when it
317   was created. It is used to restore the subscription on the Restart-Job and Reprocess-Job operations.
318   A 0 value indicates infinite time was granted.

319 "most-recentnotify-request-ids" (1setOf integer(0:MAX)) - the request-ids that was were delivered in the
320   most recent Notifications for the Per-Printer subscription for each of the Notification Recipients in
321   the corresponding parallel "notify-recipients" (1setOf uri) attribute. A value of 0 indicates that no
322   Notifications have been sent for this Per-Printer Subscription to the corresponding recipient. The
323   first request-id sent to a Recipient for this Per-Printer Subscription MUST be 1. The reason to have
324   a separate request-id for each recipient is so that when more than one Notification Recipient is
325   specified in a subscription, each recipient will have its own monotonically increasing set of request-
326   ids, i.e., no gaps, in order to be able to detect a missing notification. If an implementation doesn't
327   support more than one Notification Recipient then this attribute only has one value as well.

328 "delivery-failure-count" (integer(0:MAX)) - the number of transport delivery failures in delivering a
329   notification to any of the "notify-recipients" using the transport indicated. Note: for the current
330   delivery methods envisioned (see Notification Requirements [ipp-req], Notification Recipients do
331   not respond to the Notification Source, so this count is not the number of unacknowledged
332   notifications.

5 Events

Events are defined to be non-overlapping. So for example, 'job-state-changed' doesn't include creating the job or completing the job. This reduces the need to batch events together, since events tends to happen at different times by definition.

Both Job and/or Printer events may be specified for Per-Job or Per-Printer subscriptions:

Job events in a Per-Job subscription indicate Job events for *this* job only
Job events in a Per-Printer subscription indicates Job events for *all* jobs
Printer events in a Per-Job subscription indicate Printer events for *any* job
Printer events in a Per- ~~Job~~ Printer subscription indicate Printer events for *any* job
~~ISSUE 4 - Should we disambiguate the 'job-xxx' events and double the number by renaming to 'my-job-xxx' and 'all-job-xxx' events for use in both Per-Job and Per-Printer subscriptions?~~

5.1 Job events

'**none**': REQUIRED - no notifications of any events
~~ISSUE 5 - Add a 'job-announce' event which is sent to the Notification Recipient(s) when the subscription is accepted to prepare/warn them?~~
'**job-created**': REQUIRED - the Printer object has accepted Create-Job, Print-Job, or Print-URI.
'**job-completed**': REQUIRED - the job has reached 'completed**,** 'aborted', or 'canceled'.
'**job-state-changed**': REQUIRED - any job state or state-reason changed, except when the job is created or when the job moves to 'completed', 'aborted', or 'canceled'.
'**job-config-changed**' – "job-message-from-operator" or any of the non-READ-ONLY Job attributes changed, typically through the Set-Job-Attributes operation.
'**job-purged**' - Job-Purged operation on a 'not~~n~~-completed' job.
'**job-progress**' - sheet or copy completed. See separate [ipp-prog] spec.

5.2 Printer events

'**none**' - REQUIRED - no notification of any events
'**printer-announce**' - a Per-Job or Per-Printer subscription has been accepted. Specified by a Subscriber that wishes to warn the Notification Recipient that other Printer events are forth coming.
~~ISSUE 6 - Add a 'printer-announce' event which is sent to the Notification Recipient(s) when the subscription is accepted to prepare/warn them?~~
'**printer-restarted**' - device powered up or Restart-Printer operation is performed (see [ipp-adm]).
'**printer-shutdown**' - device is being powered down or the Shutdown-Printer operation [ipp-adm].
'**printer-state-changed**' - REQUIRED - "printer-state", "printer-state-reasons", and/or "printer-is-accepting-jobs" attributes changed, except when the Printer starts up or is shutdown.
'**printer-media-changed**' - when the media loaded ("media-ready") on a printer has been changed.
'**printer-config-changed**' - "printer-message-from-operator" or any non-READ-ONLY Printer attributes changed, except for "media-ready" (which has its own event), whether through the Set-Printer-Attributes operation or by other means and whether initiated by a human or not.
'**printer-queue-changed**' - the order of jobs in the Printer's queue has changed, so that an application that is monitoring the queue can perform a Get-Jobs operation to determine the new order. This event does not include when a job enters the queue (the 'job-created' event covers that) and does not inclded when a job leaves the queue (the 'job-completed' event covers that).

375	**'printer-no-longer-full'** - <u>the</u> Printer may still be printing but has acquired more buffer space to
376	accept jobs. This event only occurs when the Printer did not have room to accept jobs
377	previously and rejected a Print-Job, Print-URI, Create-Job, Send-Document, or Send-URI
378	operation from any client (desktop or Server).
379	**'printer-almost-idle'** - <u>the</u> Printer needs another Job in order to stay busy. Typically sent to a
380	Server that has subscribed.

6   Notification Content

ISSUE 7 - How is the User Friendly notification represented, such as is used with the 'mailto:' notification delivery scheme? What parts gets translated into the Natural Language of the subscriber (presumed to be the same as the recipient), besides the "job-trigger-message" and "printer-trigger-message"? The idea of using multipart/report has problems going through firewalls because of attachments, so maybe we need a single text message that can be read by humans, and still be parsed by programs for some of the (un-translated) keywords attribute names. Perhaps, the trigger-message contains user friendly stuff like job-name, printer-name, and the translated job-state and state-reasons values and the rest of the message is a series of lines of the form:

keyword: text

where 'keyword' is the un-translated (i.e, U.S. English) attribute keyword name and the text is the representation of the attribute values. Which attributes would be required beside the "job-trigger-message" and "printer-trigger-message"?

The notification content is the same for the current and new models. All attributes indicated are REQUIRED for *all* events, unless inside [ ] or the comment indicates otherwise:

6.1   Job Event Notification content

The following notification content is sent for both Per-Job and Per-Printer Subscriptions when a Job event occurs (see section 5.1 for Job events):

399	ISSUE 8 - Most transports only allow 500 to 1000 octets in a packet we have too much stuff.	
400	version-number (integer (0:32767))	
401	status-code (integer (0:32767)) - with the value: basic-job-event(600)	
402	request-id (integer (1:MAX)) - used as notification delivery sequence number for each recipient	
403	attributes-charset (charset)	
404	attributes-natural-language (naturalLanguage)	from "notify-natural-languages" if supported
405	[subscription-id (integer(1:MAX))]	MUST if from a Per-Printer subscription
406	job-printer-uri (uri)	
407	printer-name (name(127))	
408	job-id (integer(1:MAX))	
409	job-name (name(MAX))	
410	job-trigger-event (type2 keyword)	
411	[job-trigger-message (text(255))]	
412	ISSUE 9 - is 255 too long for "job-trigger-message (text(255))", since it also has to be sent in the	
413	notification content?	
414	ISSUE 10 - How to include user friendly data, like job-name, job-owner, printer-name, date, time,	
415	and event name in a Job event notification?	
416	ISSUE 11 - Should we REQUIRE the "printer-trigger-message" to contain the user friendly data	
417	localized information about this event, namely: like job-name, job-owner, printer-name, date, time,	
418	and event name and REQUIRE "printer-trigger-message"?	
419	job-trigger-time (integer(1:MAX))	SNMP traps automatically includes
420	[job-trigger-date-time (dateTime)]	
421	job-state (type1 enum)	
422	previous-job-state (type1 enum)	not for 'job-completed' event
423	job-state-reasons (1setOf type2 keyword)	
424	job-state-reasons-added (1setOf type2 keyword)	not for 'job-completed' event
425	job-state-reasons-deleted (1setOf type2 keyword)	not for 'job-completed' event
426	ISSUE 12 - in a Job notification why not just "previous-job-state-reasons" instead of "job-state-	
427	reasons-added" and "job-state-reasons-deleted"?	
428	[job-impressions-completed (integer(0:MAX))]	only for 'job-completed' event
429	[job-media-sheets-completed (integer(0:MAX))]	only for 'job-completed' event
430	job-collation-type (type2 enum)	only for 'job-progress' event
431	sheet-completed-copy-number (integer(-2:MAX))	only for 'job-progress' event
432	sheet-completed-document-number(integer(-2:MAX))	only for 'job-progress' event
433	impressions-interpreted (integer(-2:MAX))	only for 'job-progress' event
434	impressions-completed-current-copy (integer(-2:MAX))	only for 'job-progress' event

6.2 Printer Event Notification content:

The following notification content is sent for both Per-Job and Per-Printer Subscriptions when a Job event occurs (see section 5.2 for Printer events):

438   ISSUE 10 - Most transports only allow 500 to 1000 octets in a packet so we have too much stuff
439   here.
440   version-number (integer (0:32767))
441   status-code (integer (0:32767)) - with the value: basic-job-event(601)
442   request-id (integer (1:MAX)) - used as notification delivery sequence number for each recipient
443   attributes-charset (charset)
444   attributes-natural-language (naturalLanguage)   from "notify-natural-languages" if supported
445   [subscription-id (integer(1:MAX))]   MUST if from a Per-Printer subscription
446   printer-uri (uri)
447   printer-name (name(127))
448   [job-id (integer(1:MAX))]   MUST if from a Per-Job subscription
449   [job-name (name(MAX))]   MUST if from a Per-Job subscription
450   printer-trigger-event (type2 keyword)
451   [printer-trigger-message (text(255))]
452   ISSUE 14 - is 255 too long for "printer-trigger-message (text(255))", since it also has to be sent in
453   the notification content?
454   ISSUE 15 - How to include user friendly data, like printer-name, date, time, and event name in a
455   Printer event notification?
456   ISSUE 16 - Should we REQUIRE the "printer-trigger-message" to contain the user friendly data
457   localized information about this event, namely: printer-name, date, time, and event name and
458   REQUIRE "printer-trigger-message"?
459   printer-trigger-time (integer(1:MAX))   SNMP traps automatically includes
460   [printer-trigger-date-time (dateTime)]
461   printer-state (type1 enum)
462   previous-printer-state (type1 enum)
463   printer-state-reasons (1setOf type2 keyword)
464   printer-is-accepting-jobs (boolean)
465   printer-state-reasons-added (1setOf type2 keyword)
466   printer-state-reasons-deleted (1setOf type2 keyword)
467   ISSUE 17 - in a Printer notification why not just "previous-printer-state-reasons" instead of "printer-
468   state-reasons-added" and "printer-state-reasons-deleted"?
469

470   7   REQUIRED Per-Job Subscription mechanism

471 This section lists the REQUIRED operation requests and response that have some relation to notification,
472 i.e., that are related to Per-Job subscription.   Section 8 lists the OPTIONAL operation requests and
473 responses associated with the OPTIONAL Per-Printer subscription and its Subscription object.

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

As in the current notification spec (July 22, 1999), the usual method for a client to associate one subscription with a Job is to specify the subscription when the job is created. For a Per-Job Subscription, the client supplies the following notification operation attributes with any of the job creation operations (Create-Job, Print-Job, Print-URI), plus Validate-Job (which doesn't create a job or subscription):

notify-recipients (1setOf uri)	REQUIRED for Printer to support
[notify-events (1setOf type2 keyword)]	REQUIRED for Printer to support
[notify-user-data (octetString(255))]	REQUIRED for Printer to support
[notify-exclude-event-mask (octetString(8))]	~~OPTIONAL~~ REQUIRED for Printer to support
[notify-natural-languages" (1setOf naturalLanguage)]	OPTIONAL for Printer to support

See section 4.1.1 for a definition of these operation attributes, since they are copied to the Job object as Job Description attributes.

7.2 Get-Printer-Attributes operation

The Get-Printer-Attributes can be used to request the additional Printer Description attributes that indicate the notification supported. See section 4.1.2.

7.3 Get-Job-Attributes operation

The Get-Job-Attributes can be used to request the additional Job Description attributes that relate to notification. See section 4.1.1.

7.4 Set-Job-Subscription operation

The Set-Job-Subscription operation allows a client to change the Job attributes of the Job's Per-Job subscription, including to add them in the first place and to remove them from the Job object. This operation replaces all the values of the supplied notification attributes: "notify-recipients", "notify-events", "notify-user-data", and "notify-exclude-event-mask" Job attributes. Any notification attributes not supplied are not affected. In order to cancel a Per-Job subscription, the client supplies the "notify-events" with a 'none' value. ISSUE 11 - or should canceling a Per-Job subscription be done by supplying the out-of-band 'no-value' value for "notify-recipients" in a Set-Job-Subscription operation, instead of 'none' to "notify-events", since "notify-recipients" is the attribute that indicates that there is notification in a job creation operation? In order to remove the In order to remove the user-data (octetString(63)), the client supplies an octetString with zero length. In order to remove the "notify-exclude-event-mask (octetString(8))", the client supplies a octetString with zero length.

Request:

Group 1: Operation Attributes	Printer support
"attributes-charset"	R
"attributes-natural-language"	R
"printer-uri" (uri)	R
"requesting-user-name" (name(MAX))	RECOMMENDED
["notify-recipients" (1setOf uri)]	R
["notify-events" (1setOf type2 keyword)]	R
["notify-user-data" (octetString(255))]	R
["notify-exclude-event-mask" (1setOf octetString(8))]	R
["notify-natural-languages" (1setOf naturalLanguage)]	O

Response:

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

See section 4.1.1 for a definition of these operation attributes, since they are copied to the Job object as Job Description attributes.

8 OPTIONAL Per-Printer Subscriptions and the Subscription object

This section lists the operation requests and responses that MUST be supported if the OPTIONAL Per-Printer Subscriptions are supported which REQUIRES that the Subscription object be supported and vice versa.

8.1 Create-Printer-Subscription operation

The Create-Printer-Subscription operation creates a Per-Printer subscription by creating a Per-Printer Subscription object which is added to the Printer object.

516 `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
"notify-recipients" (1setOf uri)	R
["notify-events" (1setOf type2 keyword)]	R
["notify-user-data" (octetString(255))]	R
["notify-exclude-event-mask" (1setOf octetString(8))]	R~~O~~
["notify-natural-languages" (1setOf naturalLanguage)]	O
["notify-lease-time-requested" (integer(0:MAX))]	R
["notify-persistence" (boolean)]	O

517 `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-time" (integer(0:MAX))	R
"notify-lease-time-~~remaining~~granted" (integer(0:MAX))	R
Group 2: Unsupported Attributes	R

518 Attribute Notes:

519 "notify-lease-time-requested" (integer(0:MAX) - the number of seconds requested for the Per-Printer
520   subscription lease. A value of 0 indicates a request that the Per-Printer Subscription lease never
521   expire. Supplying a 0 value MAY require authentication in order to be used, if 0 is supported at all.

522 "notify-persistence" (boolean) - whether or not the Per-Printer Subscription is to be persistent, i.e., saved
523   across power cycles. Persistent trap registrations is a client option in SNMPv3 [RFC2573].
524   ISSUE 12 - Ok to have added the OPTIONAL "notify-persistence" (boolean) operation attribute to
525   the Create-Printer-Subscription operation?

526 "notify-lease-time" (integer(0:MAX)) - the time in the future at which the subscription lease will expire,
527   i.e., the "printer-up-time" value (in time ticks) at which the lease will expire. A value of 0 indicates
528   that the lease subscription will never expire.

529 ~~A "notify-lease-time" and "notify-lease-time-remaining" of 0 means infinite, i.e., the Per-Printer~~
530   ~~Subscription has no lease. Supplying a 0 value MAY require authentication in order to be used~~
531   ~~when supported at all..~~

532 ~~The Printer object MUST return the~~ "notify-lease-time-granted" - number of seconds of time actually
533   granted. The value of "notify-lease-time-granted" ~~which~~ may be less than the requester requested in
534   the "notify-lease-time-requested" if it was greater than the MAX supported or more than the

535   requester requested if it was less than the MIN supported, as indicated in the Printer's "lease-time-
536   supported" (rangeOfInteger(0:MAX)) attribute.

537 ~~The Printer object MUST return the "notify-lease-time-remaining" which is the number of seconds~~
538   ~~remaining in the lease.~~

8.2   Get-Printer-Subscription-Attributes operation

540 The Get-Printer-Subscription-Attributes returns the requested attributes of the identified Per-Printer
541 Subscription object. See section 4.1.3.

542 Request:

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

543 Response:

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

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

547 ### *8.3 Get-Printer-Subscriptions operation*

548 The Get-Printer-Subscriptions operation returns Per-Printer subscriptions, i.e., Subscription objects.

549 `Request:`

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

550 `Response:`

Group 1: Operation Attributes	Printer support
"status-code" (type2 enum)	R
"attributes-charset"	R
"attributes-natural-language"	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

551

552 Notes: This operation is similar to the Get-Jobs operation. If the client wants any attributes returned,
553 including the "subscription-id", it must include the attribute keyword name in the "requested-attributes"
554 operation attribute. If the "requested-attributes operation attribute is omitted, the Printer MUST respond as
555 if the client supplied the value: 'subscription-id'.

8.4 Renew-Printer-Subscription operation

556

557 The Renew-Printer-Subscription operation permits a client to request the IPP Printer to extend the lease on
558 a Per-Printer Subscription object instance.

559 Request:

Group 1: Operation Attributes	Printer support
"attributes-charset"	R
"attributes-natural-language"	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

560 Response:

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

561 Notes: The Printer object MUST return the "notify-lease-time-granted" which is the number of seconds
562 actually granted. This value which MAY be less than the requester requested if it was greater than the
563 MAX supported or more than the requester requested if it was less than the MIN supported, as indicated in
564 the Printer's "lease-time-supported" (rangeOfInteger(0:MAX)) attribute.

565 The Printer object MUST return the "notify-lease-time-remaininggranted" which is the number of seconds
566 remaining in the lease and the "notify-lease-time" which is the time in the future, i.e., the value of the
567 Printer's "printer-up-time" (time-ticks) that the notification will be canceled by the Printer.

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

573 ISSUE 13 - What happens if the client supplies a 0 for the "notify-lease-time-requested" operation
574 attribute? Does that result in a canceling of the Subscription? Is that ok? Ok to keep Cancel-Printer-
575 Subscription operation too?

576 ISSUE 14 - Ok that Renew-Printer-Subscription doesn't allow any modification of the Subscription object
577 attributes, except the "notify-lease-time" attribute? If we need a way to change other attributes would we
578 add it to this same operation and rename it to Set-Printer-Subscription? Having a Set-Printer-Subscription
579 operation would allow a client/server to set the "notify-events" to 'none' and thereby reserve a slot using the

580 <mark>lease time, but not have any Notifications until a subsequence Set-Printer-Subscription operation was done</mark>
581 <mark>that set some events.</mark>

582 Note: There is no need to renew a Per-Job Subscription, since it is effectively the time that the Job has
583 documents.

8.5   Cancel-Printer-Subscription operation

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

589 Request:

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

590 Response:

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

591

9   References

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

596 [ipp-prog]
597   Hastings, T., Bergman, R., Lewis, H., "Proposed Job Progress Attributes for IPP", <ipp-job-prog-
598   attr-990518.txt> work in progress, May 18, 1999.

599 [ipp-req]
600   Hastings, T., Manros, C., "Internet Printing Protocol/1.0 & 1.1: Requirements for IPP Notification",
601   <draft-ietf-ipp-not-032.txt> work in progress, June 24Aug 11, 1999.

602 [RFC2573]
603   D. Levi, P. Meyer, B. Stewart, "SNMP Applications", RFC 2573, April 1999.

10 Change History

The following changes have been made to this Notification Model document, in reverse chronological order:

10.1 Changes made from 99/08/07 version to make the 99/08/11 version

The following changes were made from the 99/08/07 version to make the 99/08/11 version as result of the IPP telecon, 9908/11:

1. Changed the name of the Replace-Job-Subscription operation to Set-Job-Subscription, since it can add, remove or change Job notification attributes, not just replace them.

2. Added the configuration pictures.

3. Reduced the size of "notify-user-data" from 255 octets to 63 octets, since it is also send in the Notification content which has a limit of 480 or so octets on some transports.

4. Changed the conformance requirements for the "notify-exclude-event-masks (1setOf octetString(8))" operation attributes in Job creation and Create-Printer-Subscription operations from OPTIONAL to REQUIRED, since their implementation means that clients can count on it. Also the minimum number of Per-Printer Subscriptions can be less, since a client won't be forced to use multiple subscriptions when recipients have different events. Finally, SNMPv3 has the exclude mask mechanism.

5. Added the OPTIONAL "notify-natural-languages" (1setOf naturalLanguage) Job Description attribute so that each Notification Recipient could get a requested natural language, instead of the one in the Job creation request.

6. Deleted the "job-trigger-message" Job Description attribute from Job object attribute. The client can localize the remaining Job attributes on its slow scan in case Notifications are dropped.

7. Added the "notify-request-ids (1setOf integer(0:MAX))" Job Description attribute to the Job object to indicate (and remember) the most recent request-id delivered for each Notification Recipient.

8. Deleted "previous-job-state (type1 enum)" and "job-state-reasons-added (1setOf type2 keyword)" and "job-state-reasons-deleted (1setOf type2 keyword)" from the Job object. The "job-state-reasons" value has enough information about why the Job is in the current state, so that we don't need the previous state and state reasons for the slow scan polling Notification Recipient that is attempting to overcome possible dropped Notifications.

9. Added REQUIRED "max-recipients-supported (integer(0:MAX))" Printer Description attribute to the Printer object to indicate the max number of Per-Job and Per-Printer Notification Recipients supported in each subscription. REQUIRE a minimum of 3 recipients.

10. Added the requirement that at least 8 Per-Printer Subscription object instances MUST be supported, if Per-Printer subscriptions are supported. With the exclude mask, such a low number as 8 isn't too restrictive on clients and isn't too expensive for low end devices.

11. Added the OPTIONAL "persistent-subscriptions-supported" (boolean) Printer Description attribute to the Printer object to indicate whether or not the Printer supports persistent Per-Printer Subscriptions.

12. Changed the Printer support requirements for the "notify-exclude-event-mask" (1setOf octetString(8)) from OPTIONAL to REQUIRED, so that clients can count on it and to reduce the number of Subscription objects that a Printer MUST support.

13. Added the OPTIONAL "notify-natural-languages" (1setOf naturalLanguage) Subscription object attribute so that each Notification Recipient could get a requested natural language, instead of the one in the Create-Printer-Subscription request.

14. Added the "subscription-id" attribute to the Subscription object so that each object instance is identified in a manner analogous to Job objects.

15. Changed the name of "most-recent-request-id" to "notify-request-ids" for consistency in naming and to be the same as the corresponding Job Description attribute ("notify-request-ids").

16. Changed the name of the "notify-lease-time-remaining" Subscription object attribute to "notify-lease-time". Changed the semantics from the lease time interval requested to the "printer-up-time" (time ticks) that the lease expires. The client can subtract the "printer-up-time" from the "notify-lease-time" to find out how much time is left on the lease. This change makes the attribute value be a constant, instead of having to be recomputed on each Get-Printer-Subscription-Attribute request.

17. Changed the name of the "notify-lease-time (integer(0:MAX))" Subscription object attribute to "notify-lease-time-granted (integer(0:MAX))" and indicated that it is set by the Printer not the client.

18. Added 'printer-announce' event, so that the Subscriber can have the Notification Recipient be warned that other Printer events are forth coming.

19. Added 'printer-queue-changed' - so that an application that is monitoring the queue can re-fetch the queue with Get-Jobs.

20. Added "attributes-charset" and "attributes-natural-language" to the Job and Printer Notification content, since the content is an 'application/ipp' MIME type for an IPP response.

21. Added "printer-name (name(127))" and "job-name (name(MAX))" to Job and Printer Notification Content. These are helpful to end users to identify from whence came a notification.

22. Deleted the "job-trigger-message (text(255))" from the Job Notification content since the data is already available in other attributes that the recipient should localize to present to a human. Also the Human Consumable form will have many of the attributes when it is used.

23. Deleted "previous-job-state (type1 enum)" and "job-state-reasons-added (1setOf type2 keyword)" and "job-state-reasons-deleted (1setOf type2 keyword)" from the Job Notification content. The current "job-state-reasons" value has enough information about why the Job is in the current state, so that we don't need the previous state and state reasons.

24. Deleted the "printer-trigger-message (text(255))" from the Printer Notification content, since the data is already available in other attributes that the recipient should localize to present to a human. Also the Human Consumable form will have many of the attributes when it is used.

25. Added [job-id (integer(1:MAX))] and [job-name (name(MAX))] to the Printer Notification content for when the event is from a Per-Job subscription.

26. Deleted "previous-printer-state (type1 enum)" and "printer-state-reasons-added (1setOf type2 keyword)" and "printer-state-reasons-deleted (1setOf type2 keyword)" from the Printer Notification content. The current "printer-state-reasons" value has enough information about why the Printer is in the current state, so that we don't need the previous state and state reasons.

27. Added "printer-is-accepting-jobs (boolean)" attribute to the Printer Notification content, since it is a state variable and its change causes the 'printer-state-change' event.

28. Added the definition of the Set-Job-Subscription operation.

29. Changed the name of "notify-lease-time" operation attribute in the Create-Printer-Subscription operation to "notify-lease-time-requested" to distinguish it from the "notify-lease-time" (uptime ticks) and "notify-lease-time-granted" returned in the response.

30. Added the "notify-persistence (boolean)" operation attribute to the Create-Printer-Subscriptions to indicate whether the notification content delivery is to make the Per-Printer Subscription be permanent.