

1 INTERNET-DRAFT
2 <draft-ietf-ipp-notify-mailto-02.txt>

Robert Herriot
Xerox Corp.
Henrik Holst
i-data international a/s
Tom Hastings
Xerox Corp.
Carl-Uno Manros
Xerox Corp.
July 13, 2000

12 Internet Printing Protocol (IPP):
13 **The ‘mailto:’ Notification Delivery Method**

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

15 Status of this Memo

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

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

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

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

26 **Abstract**

27 The notification extension document [ipp-ntfy] defines operations that a client can perform in order to
28 create *Subscription Objects* in a Printer and carry out other operations on them. The Subscription Object
29 specifies that when one of the specified *Events* occurs, the Printer sends an asynchronous *Event Notification*
30 to the specified *Notification Recipient* via the specified *Delivery Method* (i.e., protocol).

31 The notification extension document [ipp-ntfy] specifies that each Delivery Method is defined in another
32 document. This document is one such document, and it specifies the ‘mailto’ delivery method.

33 For this Delivery Method, when an Event occurs, the Printer immediately sends an Event Notification via
34 an email message to the Notification Recipient specified in the Subscription Object. The message body of
35 the email consists of Human Consumable text and is not intended to be parsed by a machine.

36 The Notification Recipient receives the Event Notification in the same way as it receives any other email
37 message.

38 The full set of IPP documents includes:

- 39 Design Goals for an Internet Printing Protocol [RFC2567]
- 40 Rationale for the Structure and Model and Protocol for the Internet Printing Protocol [RFC2568]
- 41 Internet Printing Protocol/1.1: Model and Semantics [ipp-mod]
- 42 Internet Printing Protocol/1.1: Encoding and Transport [ipp-pro]
- 43 Internet Printing Protocol/1.1: Implementer's Guide [ipp-iig]
- 44 Mapping between LPD and IPP Protocols [RFC2569]
- 45 Internet Printing Protocol (IPP): IPP Event Notification Specification [ipp-ntfy]

46

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

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

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

60 The "Internet Printing Protocol/1.1: Encoding and Transport" document is a formal mapping of the abstract
61 operations and attributes defined in the model document onto HTTP/1.1 [RFC2616]. It defines the
62 encoding rules for a new Internet MIME media type called "application/ipp". This document also defines
63 the rules for transporting over HTTP a message body whose Content-Type is "application/ipp". This
64 document also defines a new scheme named 'ipp' for identifying IPP printers and jobs.

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

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

72 The "Event Notification Specification" document describes an extension to the IPP/1.0, IPP/1.1, and future
73 versions. This extension allows a client to subscribe to printing related Events. The Subscription Object
74 specifies that when one of the specified *Event* occurs, the Printer sends an asynchronous *Event Notification*
75 to the specified *Notification Recipient* via the specified *Delivery Method* (i.e., protocol). A client associates
76 Subscription Objects with a particular Job by performing the Create-Job-Subscriptions operation or by
77 submitting a Job with subscription information. A client associates Subscription Objects with the Printer

78 by performing a Create-Printer-Subscriptions operation. Four other operations are defined for Subscription
79 Objects: Get-Subscriptions-Attributes, Get-Subscriptions, Renew-Subscription, and Cancel-Subscription.

80

Table of Contents

81	1	Introduction.....	6
82	2	Terminology.....	6
83	3	Model and Operation	6
84	4	General Information.....	7
85	5	Subscription Template Attributes	8
86	5.1	Additional Subscription Template Attributes.....	8
87	5.1.1	notify-mailto-text-only (boolean).....	8
88	5.2	Additional Information about Subscription Template Attributes.....	9
89	5.2.1	notify-recipient-uri (uri)	9
90	5.2.2	notify-user-data (octetString(63)).....	10
91	6	Event Notification Content	10
92	6.1	Headers	10
93	6.1.1	'Date' header	10
94	6.1.2	'From' header	10
95	6.1.3	'Subject' header.....	11
96	6.1.4	'Sender' header.....	11
97	6.1.5	'Reply-to' header.....	12
98	6.1.6	'To' header	12
99	6.1.7	'Content-type' header.....	12
100	6.2	Message Body.....	13
101	6.2.1	Event Notification Content Common to All Events.....	13
102	6.2.2	Additional Event Notification Content for Job Events	15
103	6.2.3	Additional Event Notification Content for Printer Events	15
104	6.3	Examples	16
105	6.3.1	Job Event Example.....	16
106	6.3.2	Printer Event Example.....	17
107	6.3.3	Printer Event Example (localized to Danish).....	17
108	7	Conformance Requirements.....	18
109	8	IANA Considerations.....	19
110	9	Internationalization Considerations	19
111	10	Security Considerations	19
112	11	References	19
113	12	Author's Addresses.....	20

114 13 Full Copyright Statement.....21

115

116 **Table of Tables**

117 Table 1 – Information about the Delivery Method.....7

118 Table 2 – Printer Name in Event Notification Content14

119 Table 3 – Event Name in Event Notification Content.....14

120 Table 4 – Job Name in Event Notification Content15

121 Table 5 – Job State in Event Notification Content.....15

122 Table 6 – Printer State in Event Notification Content.....15

123

124 **1 Introduction**

125 The notification extension document [ipp-ntfy] defines operations that a client can perform in order to
126 create *Subscription Objects* in a Printer and carry out other operations on them. A Subscription Object
127 represents a Subscription abstraction. The Subscription Object specifies that when one of the specified
128 *Events* occurs, the Printer sends an asynchronous *Event Notification* to the specified *Notification Recipient*
129 via the specified *Delivery Method* (i.e., protocol).

130 The notification extension document [ipp-ntfy] specifies that each Delivery Method is defined in another
131 document. This document is one such document, and it specifies the 'mailto' delivery method.

132 For this Delivery Method, when an Event occurs, the Printer immediately sends an Event Notification via
133 an email message to the Notification Recipient specified in the Subscription Object. The message body of
134 the email consists of Human Consumable text and is not intended to be parsed by a machine. The 'mailto'
135 Delivery Method is a 'push' Delivery Method as defined in [ipp-ntfy].

136 The Notification Recipient receives the Event Notification in the same way as it receives any other email
137 message.

138 **2 Terminology**

139 This section defines the following terms that are used throughout this document:

140 Capitalized terms, such as **MUST**, **MUST NOT**, **REQUIRED**, **SHOULD**, **SHOULD NOT**, **MAY**,
141 **NEED NOT**, and **OPTIONAL**, have special meaning relating to conformance to this specification. These
142 terms are defined in [ipp-mod section 13.1 on conformance terminology, most of which is taken from RFC
143 2119 [RFC2119].

144 For capitalized terms that appear in this document, see [ipp-ntfy].

145 **3 Model and Operation**

146 In a Subscription Creation Operation, when the value of the "notify-recipient-uri" attribute contains the
147 scheme "mailto", the client is requesting that the Printer use the 'mailto' Delivery Method for Event
148 Notifications generated from the new Subscription Object.

149 For this Delivery Method, the "notify-recipient-uri" attribute value **MUST** consist of a "mailto" scheme
150 followed by a colon, and then followed by an address part (e.g. 'mailto:smith@abc.com'). See section 5.2.1
151 for the syntax of the "notify-recipient-uri" attribute value for this Delivery Method.

152 A Printer **MUST** support SMTP [RFC821], and it **MAY** support other email protocols. A Printer **MAY** use
153 additional services, such as SMTP delivery status notification [RFC1891] or S/MIME encryption
154 [RFC2633].

155 If the client wants the Printer to send Event Notifications via the 'mailto' Delivery Method, the client
 156 MUST choose a value for "notify-recipient-uri" attribute which conforms to the rules of section 5.2.1. To
 157 avoid denial-of-service attacks, a client SHOULD NOT use distribution lists as the Notification Recipient.

158 When an Event occurs, the Printer MUST immediately:

- 159 1. Find all pertinent Subscription Objects P according to the rules of section 9 of [ipp-ntfy], AND
- 160 2. Find the subset M of these Subscription Objects P whose "notify-recipient-uri" attribute has a
 161 scheme value of 'mailto', AND
- 162 3. For each Subscription Object in M, the Printer MUST
 - 163 a) generate an email message as specified in section 5.2.2 AND
 - 164 b) send the email message to the Notification Recipient specified by the address part of the "notify-
 165 recipient-uri" attribute value (see section 5.2.1).

166 If the Printer supports only SMTP, it MUST send the email message via SMTP. If the Printer supports
 167 additional email protocols, it MUST determine the protocol from the address part of the "notify-recipient-
 168 uri" attribute value and then send the email message via the appropriate email protocol.

169 4 General Information

170 If a Printer supports this Delivery Method, the following are its characteristics.

171 **Table 1 – Information about the Delivery Method**

Document Method Conformance Requirement	Delivery Method Realization
1. What is the URL scheme name for the Delivery Method?	mailto
2. Is the Delivery Method REQUIRED, RECOMMEND, or OPTIONAL for an IPP Printer to support?	RECOMMENDED
3. What transport and delivery protocols does the Printer use to deliver the Event Notification Content, i.e., what is the entire network stack?	A Printer MUST support SMTP. It MAY support other email protocols.
4. Can several Event Notifications be combined into a Compound Event Notification?	A Printer implementation MAY combine several Event Notifications into a single email message.
5. Is the Delivery Method initiated by the Notification Recipient (pull), or by the Printer (push)?	This Delivery Method is a push.
6. Is the Event Notification content Machine	Human Consumable

Consumable or Human Consumable?	
7. What section in this document answers the following question? For a Machine Consumable Event Notification, what is the representation and encoding of values defined in section 9.1 of [ipp-ntfy] and the conformance requirements thereof? For a Human Consumable Event Notification, what is the representation and encoding of pieces of information defined in section 9.2 of [ipp-ntfy] and the conformance requirements thereof?	Section 6
8. What are the latency and reliability of the transport and delivery protocol?	Same as the underlying SMTP (or other optional) email transport
9. What are the security aspects of the transport and delivery protocol, e.g., how it is handled in firewalls?	Same as the underlying SMTP (or other optional) email transport
10. What are the content length restrictions?	None
11. What are the additional values or pieces of information that a Printer sends in an Event Notification content and the conformance requirements thereof?	None
12. What are the additional Subscription Template and/or Subscription Description attributes and the conformance requirements thereof?	See section 5.1.1 on "notify-mailto-text-only"
13. What are the additional Printer Description attributes and the conformance requirements thereof?	None

172 5 Subscription Template Attributes

173 5.1 Additional Subscription Template Attributes

174 This Delivery Method introduces one additional Subscription Template Attribute.

175 5.1.1 notify-mailto-text-only (boolean)

176 When the Printer generates an Event Notification from a Subscription Object, this attribute specifies
 177 whether the Printer generates the Event Notification with only plain text (i.e. 'text/plain') or with Content-
 178 Types that the Printer chooses.

179 The Printer MUST support this attribute if it supports the 'mailto' Delivery Method.

180 A client MAY supply this attribute. If a client does not supply this attribute, the Printer MUST populate this
181 attribute with the value of 'false' on the Subscription Object. There is no "notify-mailto-text-only-default"
182 attribute.

183 If the value of this attribute is 'true' in a Subscription Object, the message body of each Event Notification
184 that the Printer generates from the Subscription Object MUST contain plain text only (i.e. 'text/plain' with
185 the charset specified by the "notify-charset" Subscription Object attribute).

186 If the value of this attribute is 'false' in a Subscription Object, the message body of each Event Notification
187 that the Printer generates from the Subscription Object MUST contain a 'multipart/alternative'. One
188 message body of the 'multipart/alternative' MUST be the same as the 'text/plain' message body when this
189 attribute has the value of 'true'. Each of the other message bodies of the 'multipart/alternative' MAY be any
190 Content-Type (e.g. 'text/html', 'image/gif', 'audio/basic', etc.).

191 A Printer MUST support both values ('true' and 'false') of this attribute. There is no "notify-mailto-text-
192 only-supported" attribute.

193 **5.2 Additional Information about Subscription Template Attributes**

194 This section describes additional values for attributes defined in [ipp-ntfy].

195 **5.2.1 notify-recipient-uri (uri)**

196 This section describes the syntax of the value of this attribute for the 'mailto' Delivery Method. The syntax
197 for values of this attribute for other Delivery Method is defined in other Delivery Method Documents.

198 In order to support the 'mailto' Delivery Method, the Printer MUST support the following syntax for the
199 'mailto' Delivery Method when the Printer uses SMTP. The line below use RFC 822 syntax rules and
200 terms.

201 "mailto:" mailbox

202 Note: the above syntax allows 1 occurrence of 'mailbox'. The occurrence of 'mailbox' represents an email
203 address of a Notification Recipient.

204 For SMTP, the phrase 'address part' of the "notify-recipient-uri" attribute value refers to the 'mailbox' part
205 of the value.

206 The Printer MAY support other syntax for the 'address part' if it supports email protocols in addition to
207 SMTP.

208 **5.2.2 notify-user-data (octetString(63))**

209 This attribute has a special use for the 'mailto' Delivery Method. It specifies the email address of the
210 Subscribing Client. It is primarily useful when the Notification Recipient is some person other than the
211 Subscribing Client. Then the Notification Recipient has a way to reply to the Subscribing Client.

212 If a client specifies this Delivery Method in a Subscription Creation Operation, and the specified
213 Notification Recipient is not associated with the same person as the client, the client SHOULD supply its
214 email address as the value of the "notify-user-data" attribute. If the client does not supply this attribute, the
215 Printer MUST NOT populate the Subscription Object with this attribute.

216 **6 Event Notification Content**

217 This section describes the content of an Event Notification sent via the 'mailto' Delivery Method using the
218 SMTP protocol. This document does not describe the content for other email protocols, but an
219 implementation should use this section as a model.

220 When a Printer sends an email message via SMTP, the content MUST conform to RFC 822. The following
221 sections define the content that a Printer MUST send. A Printer MAY send additional content as long as the
222 resulting content conforms to RFC 822.

223 Each subsection below specifies the syntax that pertains to the subsection. The syntax rules and syntactic
224 terms (e.g. 'date-time') in each subsection come from RFC 822, except for the section on "Content-Type"
225 which comes from RFC 1521.

226 The Event Notification content has two parts, the headers and the message body. The headers precede the
227 message body and are separated by a blank line (see [RFC 822]).

228 **6.1 Headers**

229 When a Printer sends an Event Notification via SMTP, it MUST include the following headers. RFC 822
230 RECOMMENDS that the headers be in the order that they appear below.

231 **6.1.1 'Date' header**

232 **Syntax:** "Date" ":" date-time

233 This header contains the date and time that the Event occurred.

234 The Printer MUST include a "Date" header if and only if it supports the "printer-current-time" Printer
235 attribute.

236 **6.1.2 'From' header**

237 **Syntax:** "From" ":" mailbox

238 where

239 mailbox = addr-spec / phrase route-addr

240 This header causes a typical email reader to show the email as coming from the Printer that is sending the
241 Event Notification.

242 The Printer MUST include a "From" header whose syntax is specified above.

243 The Printer MUST use the second alternative of the syntax for 'mailbox' defined above (i.e. 'phrase route-
244 addr'). The 'phrase' is the Printer's display name and it MUST be the value of the "printer-name" Printer
245 attribute. The 'route-addr' MUST contain an email address (inside angle brackets) belonging to either an
246 administrator or the output-device. This email address NEED NOT be capable of receiving mail. There is
247 no Printer attribute to hold this email address, so that it cannot be configured using the IPP protocol without
248 an implementation-defined attribute extension.

249 **6.1.3 'Subject' header**

250 **Syntax:** "Subject" ":" *text

251 This header specifies the subject of the message and contains a short summary of the Event Notification.

252 The Printer MUST include a "Subject" header whose syntax is specified above.

253 The Printer MUST localize the '*text' using the values of the "notify-charset" and "notify-natural-
254 language" Subscription Object attributes.

255 For Printer Events, the '*text' SHOULD start with the localized word "printer:", followed by the Printer
256 name, and then followed by the localized Event name, e.g., in English: "printer: 'tiger' stopped" or in
257 French: 'imprimeur: 'tigre' arrêté'.

258 For Job Events, the '*text' SHOULD start with the localized phrase "print job:", followed by the Job name,
259 and then followed by the localized Event name, e.g., in English: "print job: 'financials' completed".

260 The wording is implementation dependent. A Notification Recipient MUST NOT expect to be able to
261 parse this text. But an email filter might look for "printer" or "print job".

262 **6.1.4 'Sender' header**

263 **Syntax:** "Sender" ":" mailbox

264 This header causes a typical email reader to show the email as coming on behalf of the person associated
265 with the Subscribing Client.

266 If the Subscription Object contains the "notify-user-data" attribute, and if its value satisfies the RFC 822
267 syntax rules for 'mailbox', the Printer MUST include a "Sender" header whose syntax is specified above.
268 Otherwise, the Printer MUST NOT include a "Sender" header.

269 For the "Sender" header, the 'mailbox' MUST be the value of the "notify-user-data" Subscription Object
270 attribute. See section 5.2.2 for details about the "notify-user-data" attribute.

271 **6.1.5 'Reply-to' header**

272 **Syntax:** "Reply-to" ":" mailbox

273 If the Notification Recipient replies to Event Notification email, this header causes a typical email reader to
274 send email to the person acting as the Subscribing Client. The rules are identical to the "Sender" header.

275 If the Subscription Object contains the "notify-user-data" attribute, and if its value satisfies the RFC 822
276 syntax rules for "mailbox", the Printer **MUST** include a "Reply-to" header whose syntax is specified above.
277 Otherwise, the Printer **MUST NOT** include a "Reply-to" header.

278 For the "Reply-to" header, the "mailbox" **MUST** be the value of the "notify-user-data" Subscription Object
279 attribute. See section 5.2.2 for details about the "notify-user-data" attribute.

280 **6.1.6 'To' header**

281 **Syntax:** "To" ":" 1#mailbox

282 See [RFC 1521] for the syntax.

283 This header specifies the Notification Recipient(s).

284 The Printer **MUST** include a "To" header whose syntax is specified above.

285 The '1#mailbox' **MUST** be the '1#mailbox' part of the value of the "notify-recipient-uri" Subscription
286 attribute, i.e. the part after the "mailto:".

287 **6.1.7 'Content-type' header**

288 **Syntax:** "Content-Type" ":" type "/" subtype *(";"parameter)

289 See [RFC 1521] for the syntactic terms (e.g. 'type').

290 This header specifies the format of the message body.

291 The Printer **MUST** include the "Content-Type" header.

292 If the value of the "notify-mailto-text-only" Subscription Object attribute is 'true', the 'type' **MUST** be
293 "plain", the 'subtype' **MUST** be "text" and the 'parameter' **MUST** be "charset=" XXX" where XXX is the
294 value of the "notify-charset" Subscription Object attribute, e.g. 'text/plain; charset=UTF-8'.

295 If the value of the "notify-mailto-text-only" Subscription Object attribute is 'false', the 'type' **MUST** be
296 "multipart", the 'subtype' **MUST** be "alternative" and the 'parameter' **MUST** include the boundary string.
297 Each header of a body part of a multipart entity also has a Content-Type and its value of 'type', 'subtype'
298 and 'parameter' **MUST** be values allowed by RFC 1521 or some registered MIME type. That is, a Printer
299 **MAY** send any format it wishes in each body part of a multipart entity, e.g. 'text/html', 'image/gif', or
300 'audio/basic'.

301 **6.2 Message Body**

302 This document describes a message body that is plain text. The content of all other Content-Types is
303 implementation dependent. A Printer **MUST** include a plain text message even when it sends other
304 Content-Types in a 'multipart/alternative'.

305 When a Printer sends a plain text message, it **MUST** localize the text using the values of the "notify-
306 charset" and "notify-natural-language" Subscription Object attributes.

307 Section 9.2 in [ipp-ntfy] specifies the information that a Delivery Method **MUST** specify and a Printer
308 **SHOULD** send. This section contains the information from section 9.2 in [ipp-ntfy] and changes "Printer
309 **SHOULD** send" to "Printer **MUST** send".

310 A Printer **MUST** send the following localized information in the message body. The specific wording of
311 this information and its layout are implementation dependent.

- 312 a) the Printer name (see Table 2)
- 313 b) omitted (see below).
- 314 c) for Printer Events only:
 - 315 i) the Event (see Table 3) and/or Printer state information (see Table 6)
- 316 d) for Job Events only:
 - 317 i) the job identity (see Table 4)
 - 318 ii) the Event (see Table 3) and/or Job state information (see Table 5)

319 Item b) in the above list is omitted because the Printer sends the time of the Event as an email header (see
320 section 6.1.1 on the 'Date' header).

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

322 The Printer **MAY** send additional information, depending on implementation.

323 Notification Recipients **MUST NOT** expect to be able to parse the message.

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

- 325 a) for all Events
- 326 b) for Job Events only
- 327 c) for Printer Events only

328 **6.2.1 Event Notification Content Common to All Events**

329 The Printer **MUST** send the following information.

330 There is a separate table for each piece of information. Each row in the table represents a source value for
331 the information and the values are listed in order of preference, with the first one being the preferred one.
332 An implementation **SHOULD** use the source value from the earliest row in each table. It **MAY** use the

333 source value from another row instead, or it MAY combine the source values from several rows. An
 334 implementation is free to determine the best way to present this information.

335 The tables in this section and following contain the following columns for each piece of information:

- 336 a) **Source of Value:** the name of the attribute that supplies the value for the Event Notification
- 337 b) **Sends:** if the Printer supports the value (column 1) on the Source Object (column 3) the
 338 Delivery Method MUST specify
- 339 **MUST:** that the Printer MUST send the value.
- 340 **SHOULD:** either that the Printer MUST send the value or that the value is incompatible
 341 with the Delivery Method.
- 342 **MAY:** that the Printer MUST, SHOULD, MAY, MUST NOT, SHOULD NOT, or NEED
 343 NOT send the value. The Delivery Method specifies the level of conformance for the Printer.
- 344 c) **Source Object:** the object from which the source value comes.

345 In all tables of this section, all rows contain a "MAY" in order to state that the Delivery Method specifies
 346 the conformance.

347 Table 2 lists the source of the information for the Printer Name. The "printer-name" is more user-friendly
 348 unless the Notification Recipient is in a place where the Printer name is not meaningful. For example, an
 349 implementation could have the intelligence to send the value of the "printer-name" attribute to a
 350 Notification Recipient that can access the Printer via value of the "printer-name" attribute and otherwise
 351 send the value of the "notify-printer-uri" attribute.

352 **Table 2 – Printer Name in Event Notification Content**

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

353

354 Table 3 lists the source of the information for the Event name. A Printer MAY combine this information
 355 with state information described for Jobs in Table 5 or for Printers in Table 6.

356 **Table 3 – Event Name in Event Notification Content**

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

357

358 6.2.2 Additional Event Notification Content for Job Events

359 This section lists the source of the additional information that a Printer MUST send for Job Events.

360 Table 4 lists the source of the information for the job name. The “job-name” is likely more meaningful to a
361 user than “job-id”.

362 **Table 4 – Job Name in Event Notification Content**

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

363

364 Table 5 lists the source of the information for the job-state. If a Printer supports the “job-state-message” and
365 “job-detailed-state-message” attributes, it SHOULD use those attributes for the job state information,
366 otherwise, it should fabricate such information from the “job-state” and “job-state-reasons”. For some
367 Events, a Printer MAY combine this information with Event information.

368 **Table 5 – Job State in Event Notification Content**

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

369 6.2.3 Additional Event Notification Content for Printer Events

370 This section lists the source of the additional information that a Printer MUST send for Printer Events.

371 Table 6 lists the source of the information for the printer-state. If a Printer supports the “printer-state-
372 message”, it SHOULD use that attribute for the job state information, otherwise it SHOULD fabricate such
373 information from the “printer-state” and “printer-state-reasons”. For some Events, a Printer MAY combine
374 this information with Event information.

375 **Table 6 – Printer State in Event Notification Content**

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

376 6.3 Examples

377 This section contains three examples. One is a Job Event and the other two are Printer Events, the latter in
378 Danish.

379 A Printer implementation NEED NOT generate Event Notification content that is identical or even similar
380 to these examples. In fact it would be unfortunate if every implementation copied these example as is.
381 These examples merely show some possibilities and are not necessarily the best way to convey information
382 about an Event.

383 6.3.1 Job Event Example

384 This section contains an example of an Event Notification of a Job Event.

385 A Subscribing Client Mike Jones (who works for xyz Corp.) performs a Subscription Creation Operation as
386 part of the Print-Job operation on Printer "ipp://tiger@abc.com". Mike Jones specifies that the "job-name"
387 is "financials". Mike is printing the Job for Bill Smith at abc Corp. The Subscription Object then has the
388 following attributes:

Attribute Name	Attribute Value
notify-recipient-uri	mailto:bsmith@abc.com
notify-events	job-completed
notify-user-data	mjones@xyz.com
notify-mailto-text-only	true
notify-charset	us-ascii
notify-natural-language	en-us
notify-subscription-id	35692
notify-sequence-number	0
notify-printer-up-time	34593
notify-printer-uri	ipp://tiger@abc.com
notify-job-id	345
notify-subscriber-user-name	mjones

389 When the Job completes, the Printer generates and sends the following email message:

```
390 Date: 17 Jul 00 1632 PDT
391 From: tiger <printAdmin@abc.com>
392 Subject: print job: 'financials' completed
393 Sender: mjones@xyz.com
394 Reply-to: mjones@xyz.com
395 To: bsmith@abc.com
396 Content-type: text/plain
397
398 printer: tiger
399 job: financials
```


400 job-state: completed

401 The reader should note that the phrases are not identical to IPP keywords. They have been localized to
402 English.

403 **6.3.2 Printer Event Example**

404 This section contains an example of an Event Notification of a Printer Event.

405 A Subscribing Client Peter Williams, a Printer admin, performs a Create-Printer-Subscriptions operation on
406 Printer "ipp://tiger@abc.com". The Subscription Object then has the following attributes:

Attribute Name	Attribute Value
notify-recipient-uri	mailto:pwilliams@abc.com
notify-events	printer-state-changed
notify-mailto-text-only	true
notify-charset	us-ascii
notify-natural-language	en-us
notify-subscription-id	4623
notify-sequence-number	0
notify-printer-uptime	23002
notify-printer-uri	ipp://tiger@abc.com
notify-lease-expiration-time	0
notify-subscriber-user-name	pwilliams

407 When the Printer jams, the Printer generates and sends the following email message:

```
408       Date: 29 Aug 00 0832 PDT
409       From: tiger <printAdmin@abc.com>
410       Subject: printer: 'tiger' has stopped
411       To: pwilliams@abc.com
412       Content-type: text/plain
```

```
413       
414       Printer tiger has stopped with a paper jam.
415       
```

416 The reader should note that the phrases are not identical to IPP keywords. They have been localized to
417 English.

418 **6.3.3 Printer Event Example (localized to Danish)**

419 This section contains an example of an Event Notification of a Printer Event localized to Danish.

420 A Subscribing Client Per Jensen, a Printer admin, performs a a Create-Printer-Subscriptions operation on
421 Printer "ipp://tiger@def.dk". The Subscription Object then has the following attributes:

Attribute Name	Attribute Value
notify-recipient-uri	mailto:pjensen@def.dk
notify-events	printer-state-changed
notify-mailto-text-only	true
notify-charset	utf-8
notify-natural-language	da
notify-subscription-id	50225
notify-sequence-number	0
notify-printer-uptime	53217
notify-printer-uri	ipp://tiger@def.dk
notify-lease-expiration-time	0
notify-subscriber-user-name	pjensen

422 When the Printer jams, the Printer generates and sends the following email message:

```
423 Date: 29 Jan 00 0832 CET
424 From: tiger <admin@def.dk>
425 Subject: Printeren 'tiger' er standset
426 To: pjensen@def.dk
427 Content-type: text/plain; charset=utf-8
428
429 Printerens navn er 'tiger'.
430 Printeren er standset.
431 Aarsagen er papir stop.
```

432 7 Conformance Requirements

433 The 'mailto' Delivery Method is RECOMMENDED for a Printer to support.

434 If the Printer supports the 'mailto' Delivery Method, the Printer MUST:

- 435 1. meet the conformance requirements defined in [ipp-ntfy].
- 436 2. support the "notify-mailto-text-only" Subscription Object attribute defined in section 5.1.1.
- 437 3. support the syntax for the "notify-recipient-uri" Subscription Object attribute defined in section 5.2.1
- 438 4. support the use for the "notify-user-data" Subscription Object attribute defined in section 5.2.2
- 439 5. support SMTP for sending Event Notifications.
- 440 6. support the 'text/plain' Content-Type for the message body.
- 441 7. support sending Event Notification via email with the content specified in section 5.2.

442 **8 IANA Considerations**

443 Because the 'mailto' URL scheme is already defined in a standards track document [RFC 2368] and
444 registered with IANA, this document does not require anything further of IANA.

445 **9 Internationalization Considerations**

446 This Delivery Method presents no internationalization considerations beyond those covered in the [ipp-
447 ntfy] document, and sections 6.1.3 and 6.2 of this document.

448 The Notification Recipient is expected to present the email as received because the Printer does all
449 necessary localization to the Event Notification contents.

450 **10 Security Considerations**

451 The biggest security concern is that a Subscribing Client will cause unsolicited Event Notifications to be
452 sent to third parties, potentially creating denial-of-service problems (i.e., spam). The problem is even worse
453 if the third parties are distribution lists.

454 There exist scenarios where third party notification is required (see Scenario #2 and #3 in [ipp-not-req]).
455 The fully secure solution would require active agreement of all persons before they can become Notification
456 Recipients. However, requirement #9 in [ipp-req] ("There is no requirement for IPP Printer receiving the
457 print request to validate the identity of an event recipient") argues against this. To minimize the risk, a
458 Printer could disallow third party Notification Recipients (a traditional facsimile model).

459 The Delivery Method recommends that the Subscribing Client supply his or her email address as the value
460 of the "notify-user-data" attribute in the Subscription Creation Operation when the Notification Recipient is
461 a third party. To reduce the chance of spamming or identify the spammer, a Printer could disallow third
462 party Notification Recipients if the Subscribing Client doesn't supply the "notify-user-data" attribute with a
463 valid email address.

464 Some firewall administrators prevent mail attachments from being accepted into their organizations because
465 of the problem of the attachments containing computer viruses. The 'mailto' Delivery Method allows the
466 Subscribing Client to request that the Content-Type of a message body be 'text/plain'.

467 **11 References**

468 [ipp-iig]

469 Hastings, T., Manros, C., Kugler, K, Holst H., Zehler, P., "Internet Printing Protocol/1.1: draft-ietf-
470 ipp-implementers-guide-v11-01.txt, work in progress, May 9, 2000

471 [ipp-mod]

472 R. deBry, T. Hastings, R. Herriot, S. Isaacson, P. Powell, "Internet Printing Protocol/1.0: Model and
473 Semantics", <draft-ietf-ipp-model-v11-06.txt>, March 1, 2000.

- 474 [ipp-ntfy]
475 Isaacson, S., Martin, J., deBry, R., Hastings, T., Shepherd, M., Bergman, R., "Internet Printing
476 Protocol/1.1: IPP Event Notification Specification", <draft-ietf-ipp-not-spec-04.txt>, July 13, 2000.
- 477 [ipp-pro]
478 Herriot, R., Butler, S., Moore, P., Tuner, R., "Internet Printing Protocol/1.1: Encoding and
479 Transport", draft-ietf-ipp-protocol-v11-05.txt, March 1, 2000.
- 480 [RFC821]
481 Jonathan B. Postel, "Simple Mail Transfer Protocol", RFC 821, August, 1982.
- 482 [RFC822]
483 David H. Crocker, "Standard For The Format Of ARPA Internet Text Messages", RFC 822, August
484 13, 1982.
- 485 [RFC1341]
486 N. Borenstein, N. Freed, "MIME (Multipurpose Internet Mail Extensions): Mechanisms for
487 Specifying and Describing the Format of Internet Message Bodies", RFC 1341, June, 1992.
- 488 [RFC1521]
489 N. Borenstein, N. Freed, "MIME (Multipurpose Internet Mail Extensions) Part One: Mechanisms
490 for Specifying and Describing the Format of Internet Message Bodies", RFC 1521, September 1993.
- 491 [RFC1891]
492 K. Moore, "SMTP Service Extension for Delivery Status Notifications", RFC 1891, January 1996
- 493 [RFC2026]
494 S. Bradner, "The Internet Standards Process -- Revision 3", RFC 2026, October 1996.
- 495 [RFC2046]
496 R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, "Hypertext
497 Transfer Protocol - HTTP/1.1", RFC 2616, June 1999.
- 498 [RFC2368]
499 P. Hoffman, L. Masinter, J. Zawinski, "The mailto URL scheme", RFC 2616, July 1998.
- 500 [RFC2616]
501 R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, "Hypertext
502 Transfer Protocol - HTTP/1.1", RFC 2616, June 1999.
- 503 [RFC2633]
504 B. Ramsdell, "S/MIME Version 3 Message Specification", RFC 2633, June 1999.

505 **12 Author's Addresses**

506

507 Robert Herriot
508 Xerox Corporation
509 3400 Hillview Ave., Bldg #1
510 Palo Alto, CA 94304
511
512 Phone: 650-813-7696
513 Fax: 650-813-6860
514 Email: robert.herriot@pahv.xerox.com

515
516 Henrik Holst
517 i-data international a/s
518 Vadstrupvej 35-43
519 2880 Bagsvaerd, Denmark
520
521 Phone: +45 4436-6000
522 Fax: +45 4436-6111
523 e-mail: hh@i-data.com

524
525 Tom Hastings
526 Xerox Corporation
527 737 Hawaii St. ESAE 231
528 El Segundo, CA 90245
529
530 Phone: 310-333-6413
531 Fax: 310-333-5514
532 e-mail: hastings@cp10.es.xerox.com

533
534 Carl-Uno Manros
535 Xerox Corporation
536 737 Hawaii St. ESAE 231
537 El Segundo, CA 90245
538
539 Phone: 310-333-8273
540 Fax: 310-333-5514
541 e-mail: manros@cp10.es.xerox.com

542 **13 Full Copyright Statement**

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

544 This document and translations of it may be copied and furnished to others, and derivative works that
545 comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and
546 distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and
547 this paragraph are included on all such copies and derivative works. However, this document itself may not
548 be modified in any way, such as by removing the copyright notice or references to the Internet Society or
549 other Internet organizations, except as needed for the purpose of developing Internet standards in which

550 case the procedures for copyrights defined in the Internet Standards process must be followed, or as
551 required to translate it into languages other than English.

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

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