

3
4
5
6
7
8 **Internet Printing Protocol/1.1:**
9 **attribute syntax for The 'cCollection' attribute syntax**

10 Status of this Memo:

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

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

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

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

20 **Abstract**

21 This document specifies an OPTIONAL attribute syntax called 'collection' for use with the
22 Internet Printing Protocol/1.0 (IPP) [RFC2565, RFC2566] and IPP/1.1 [ipp-mod, ipp-pro]. A
23 'collection' is a container holding one or more named values, which are called "member"
24 attributes. A collection allows data to be grouped like a C struct.

25 **Table of Contents**

26 1 Problem Statement..... 2
27 2 Solution..... 2
28 3 Definition of a collection type..... 2
29 4 Unsupported Values 3
30 5 Encoding..... 3
31 6 Legacy issues..... 4
32 7 IANA Considerations 5
33 8 Internationalization Considerations..... 5
34 9 Security Considerations..... 5
35 10 References 5
36 11 Author's Addresses 6
37 12 APPENDIX A: Example of collection usage..... 6
38 12.1 "job-notify" Operation attribute 6
39 13 Appendix A: Full Copyright Statement..... 7

40

41 1 Problem Statement

42 IPP supports most of the common data structures that are available in programming languages. It lacks a
43 mechanism for grouping several values of different types. The C language uses the struct to solve this
44 problem.

45 2 Solution

46 The IPP 'collection' is a container holding one or more named values (i.e. attributes), which are called
47 member attributes. A collection also has a type name, which identifies the allowed member attributes, as
48 does the name of a C struct or Java class. A collection value is similar to a group, such as an operation
49 group. They both consist of a series of attributes.

50 The name of each member attribute **MUST** be unique within a collection, but **MAY** be the same as the
51 name of a member attribute in another collection type. In order to support legacy IPP implementations, the
52 name of a member attribute **MUST** be different from any attribute in an operation or object unless its
53 semantics are identical to those in the operation or object.

54 Each member attribute can have any syntax type, including collection, and can be either single-valued or
55 multi-valued. The length of a collection value is not limited. However, the length of each member attribute
56 **MUST NOT** exceed the limit of its attribute syntax.

57 Note: if a collection contains two or more member attributes with the same attribute name, the collection is
58 not well formed. The receiver of such a collection can either treat the collection as a bad value or ignore all
59 but one of the identically named members.

60 3 Definition of a collection type

61 When a specification defines an attribute whose syntax type is 'collection' or '1setOf collection', it must
62 define following aspects of the collection.

- 63 1. the name of the collection type, whose characters are the same as those for a keyword.
- 64 2. the following information about each member attribute:
 - 65 a) its name, which is a keyword like all attributes. It must be unique within the collection type. It must
66 also be unique with respect to operation and object attributes unless its semantics are identical to
67 those in the operation or object.
 - 68 b) its syntax type, which may be any IPP syntax type, include collection. If the syntax type starts with
69 "1setOf", the member attribute is multi-valued.
 - 70 c) its allowed values, either enumerated explicitly or specified by the values of a referenced attribute.
 - 71 d) whether it **MUST** be or **MAY** be supplied by a client.

- 72 e) its default value if a client MAY supply it. The default value can be stated explicitly or can come
73 from a specified attribute.
- 74 f) whether it MUST be or MAY be supported by the printer.
- 75 g) its semantics

76 4 Unsupported Values

77 The rules for returning an unsupported collection attribute are an extension to the current rules.

- 78 1. If a collection contains unrecognized, unsupported member attributes and/or conflicting value,
79 the attribute returned in the Unsupported Group is a collection containing the unrecognized,
80 unsupported member attributes, and/or conflicting values. The unrecognized member attributes
81 have an out-of-band value of unsupported. The unsupported member attributes and conflicting
82 values have their unsupported values.

83 5 Encoding

84 This section defines the encoding of a collection syntax type. A collection is encoded by using three new
85 tags:

| Tag name | Tag value | Meaning |
|-----------------|-----------|--|
| beginCollection | 0x34 | Begin the named collection. |
| endCollection | 0x37 | End the named collection. |
| sepCollection | 0x38 | Separate two collections of a multi-valued attribute |

86 A collection value is encoded as a sequence of attribute values preceded by a beginCollection value and
87 followed by an endCollection value. The value field of a beginCollection and an endCollection both
88 contain the name of the collection type, which is a string of ASCII characters. These values allow a
89 receiver to optionally match an endCollection value with a beginCollection. A 1setOf collection is encoded
90 using the rules for 1setOf and collection, except that adjacent endCollection and beginCollection values
91 MUST be combined into a single sepCollection value. Its value field contains the collection type. In a
92 1setOf collection, the endCollection value marks the end of last collection in the 1setOf collection. For
93 legacy reasons, the name field for the endCollection and sepCollection must be non-empty. The name is
94 arbitrarily assigned to be "c".

95 The following example is written in the style of the IPP/1.1 "Encoding and Transport" document [ipp-pro].
96 The following example is for a job-notify attribute containing a set of 2 collections.

| Octets | Symbolic Value | Protocol field | comments |
|--------|-----------------|----------------|-----------------------------|
| 0x34 | beginCollection | value-tag | Beginning of the collection |
| 0x000a | | name-length | |

| Octets | Symbolic Value | Protocol field | comments |
|----------------------|------------------|----------------|-------------------------------------|
| job-notify | job-notify | Name | |
| 0x000f | | Value-length | |
| job-notify-coll | job-notify-coll | Value | Collection type |
| 0x45 | uri type | value-tag | "notify-recipients" attribute |
| 0x0010 | | name-length | |
| notify-recipient | notify-recipient | Name | |
| 0x0013 | | value-length | |
| ipp-notify:port=700 | | Value | |
| 0x44 | keyword type | value-tag | "notify-event-groups" attribute |
| 0x000d | | name-length | |
| notify-events | | Name | |
| 0x0d | | value-length | |
| job-completed | | Value | |
| 0x44 | keyword type | value-tag | 2nd "notify-event-groups" attribute |
| 0x0000 | | name-length | 0 length means next multiple value |
| 0x0011 | | value-length | |
| job-state-changed | job-completion | Value | |
| 0x38 | sepCollection | value-tag | Separator between collection values |
| 0x0001 | | name-length | |
| c | | Name | Non-empty for legacy |
| 0x000f | | value-length | |
| job-notify-coll | | Value | Matches value of beginCollection |
| 0x45 | uri type | value-tag | "notify-recipients" attribute |
| 0x0010 | | name-length | |
| notify-recipient | | Name | |
| 0x0014 | | value-length | |
| mailto:smith@foo.com | | Value | |
| 0x44 | keyword type | value-tag | "notify-event-groups" attribute |
| 0x000d | | name-length | |
| notify-events | | Name | |
| 0x0d | | value-length | |
| job-completed | | Value | |
| 0x37 | endCollection | value-tag | End of last collection |
| 0x0001 | | name-length | |
| c | | Name | Non-empty for legacy |
| 0x000f | | value-length | |
| job-notify-coll | | Value | Matches value of beginCollection |

97 6 Legacy issues

98 The encoding has been designed to work with IPP/1.0 and IPP/1.1 implementations. An IPP/1.0 or IPP/1.1
99 receiver will treat the three new syntax types, beginCollection, endCollection and sepCollection as
100 unrecognized syntax types. A legacy implementation is expected to behave as follows.

101 A beginCollection value appears to be an attribute with an unsupported value.

102 The member attributes that follow the beginCollection appear to be normal attributes within their group
103 (e.g. normal for the operation attributes group). If an attribute has the same name as an attribute allowed in
104 the group, it as a recognized member of the group (e.g. as a normal operation attribute).

105 An endCollection value appears to be an attribute with an unsupported value and unrecognized name "c".
106 The same is true for a sepCollection value.

107 7 IANA Considerations

108 This attribute syntax will be registered with IANA after the WG approves its specification according to the
109 procedures for extension of the IPP/1.1 Model and Semantics [ipp-mod] and after IPP becomes a proposed
110 IETF standard.

111 8 Internationalization Considerations

112 This attribute syntax by itself has no impact on internationalization. However, the member attributes that
113 are subsequently defined for use in a collection may have internationalization considerations, as may any
114 attribute.

115 9 Security Considerations

116 This attribute syntax causes no more security concerns than any attribute syntax. It is only the attributes
117 that are subsequently defined to use this or any other attribute syntax that may have security concerns,
118 depending on the semantics of the attribute.

119 10 References

120 [ipp-mod]

121 Isaacson, S., deBry, R., Hastings, T., Herriot, R., Powell, P., "Internet Printing Protocol/1.1: Model
122 and Semantics" draft-ietf-ipp-model-v11-043.txt, June 23, 1999.

123 [ipp-not]

124 Isaacson, S., Martin, J., deBry, R., Hastings, T., [Shepherd, M., Bergman, R.](#) " Internet Printing
125 Protocol/1.0 & 1.1: IPP Event Notification Specification" draft-ietf-ipp-not-spec-010.doc, work in
126 progress, [October 10](#)~~August 25~~, 1999.

127 [ipp-pro]

128 Herriot, R., Butler, S., Moore, P., Turner, R., "Internet Printing Protocol/1.1: Encoding and
129 Transport", draft-ietf-ipp-protocol-v11-032.txt, June 23, 1999.

130 [ISO-10175]

131 ISO/IEC 10175 Document Printing Application (DPA), June 1996.

132 [\[RFC2565\]](#)
133 [Herriot, R., Butler, S., Moore, P., Tuner, R., "Internet Printing Protocol/1.0: Encoding and](#)
134 [Transport", RFC 2565, April 1999.](#)

135 [\[RFC2566\]](#)
136 [R. deBry, T. Hastings, R. Herriot, S. Isaacson, P. Powell, "Internet Printing Protocol/1.0: Model and](#)
137 [Semantics", RFC 2566, April 1999.](#)

138 **11 Author's Addresses**

139 [Tom Hastings](#)
140 [Xerox Corporation](#)
141 [737 Hawaii St. ESAE 231](#)
142 [El Segundo, CA 90245](#)
143
144 [Phone: 310-333-6413](#)
145 [Fax: 310-333-5514](#)
146 [e-mail: \[hastings@cp10.es.xerox.com\]\(mailto:hastings@cp10.es.xerox.com\)](#)
147

148 [Robert Herriot](#)
149 [Xerox Corp.](#)
150 [3400 Hill View Ave, Building 1](#)
151 [Palo Alto, CA 94304](#)
152
153 [Phone: 650-813-7696](#)
154 [Fax: 650-813-6860](#)
155 [e-mail: \[robert.herriot@pahv.xerox.com\]\(mailto:robert.herriot@pahv.xerox.com\)](#)
156

157 [Roger deBry](#)
158 [Utah Valley State College](#)
159 [Orem, UT 84058](#)
160
161 [Phone: \(801\) 222-8000](#)
162 [EMail: \[debryro@uvsc.edu\]\(mailto:debryro@uvsc.edu\)](#)

163 **12 APPENDIX A: Example of collection usage**

164 This section describes one collection Job Template example.

165 **12.1 "job-notify" Operation attribute**

166 The following example illustrates the definition of a collection attribute for the "job-notify" operation
167 attribute. Each column of the table corresponds to information that is required for member attributes. Only
168 the semantics have been omitted.

169 1. collection type: "job-notify-coll"

170 2. members of the collection

| Member name | Member type | Supported-values | Client supplied/ default | Printer support |
|--|----------------------|--|---|--------------------|
| notify-recipient | uri | notify-recipient- schemes-supported | MUST | MUST |
| notify-events | 1setOf type2 keyword | notify-events- supported | notify-events-default | MUST |
| subscriber-user-data | octetString(63) | <any octet string> | <empty octetString> | MUST |
| notify-attributes- charset | charset | charset-supported | attributes-charset in operation group | MAY |
| notify-attributes- natural-language | naturalLanguage | generated-natural- language-supported | attributes-natural- language in operation group | MAY |

171 Note: for the "client supplied/default" column, the default is specified if the client MAY supply it.

172 **13 Appendix A: Full Copyright Statement**

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

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

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

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

189