

Resource objects ~~counter~~ summary proposal

From ~~Paul Moore~~ Ira McDonald and Tom Hastings

11/27/00

File: resource-objects-~~counter~~summary-proposal-001127-rev.doc

This document is a ~~counter~~ summary proposal ~~to of~~ the one that Ira McDonald and Tom Hastings prepared for the September 2000 PWG meeting that defined a general Resource object with a "resource-type" attribute for sub-typing and operations to query, create, and delete Resource object instances. See ftp://ftp.pwg.org/pub/pwg/ipp/new_RES/draft-ietf-ipp-get-resource-01.pdf. In order to compare the resource object approach more directly with Paul Moore's counter proposal, this document is shows revisions to Paul's proposal to make the Resource object proposal. Also non-essential differences, such as filtering have been made the same, so that only real differences remain.

Table of Contents

1	Introduction and Summary.....	1
2	Get-Resource-Attributes and Get-Resources <u>Get-Printer-Collection-Rows</u> operations <u>32</u>	
3	Operations to Add, Delete, or Modify <u>Resource object instances</u> Collection rows ..	<u>43</u>
3.1	<u>Create-Resource</u> Add Printer Collection Row	<u>43</u>
3.2	<u>Delete-Resource</u> Delete Printer Collection Row	<u>54</u>
3.3	<u>Set-Resource-Attributes</u> Modify Printer Collection Row operation	<u>54</u>
3.3	<u>Renew-Resource</u>	<u>5</u>
4	Get-Resource-Data <u>Get-Printer-Collection-Row-Data</u> , <u>Create-Resource</u> Set-Printer-Collection-Row-Data operations	<u>65</u>
5	Observations	<u>75</u>
6	Suggested <u>Resource member</u> -attributes	<u>76</u>
7	Possible <u>Resource types</u> ' !setOf collection ' Printer attributes	<u>86</u>
8	Examples	<u>87</u>
8.1	Input-trays	<u>87</u>
8.2	Media descriptions	<u>97</u>
8.3	Images	<u>97</u>

1 Introduction and Summary

Use a polymorphic generic Resource object type with sub-typing '~~!setOf collection~~' ~~attributes~~ to describe fonts, media, paper trays, downloaded JPEGs, ICC Color Profiles, macros, ... Some of these resources can be down-loaded into the Printer, some can be

36 installed by means outside the IPP protocol, and some can be properties or characteristics
37 of the Printer as it comes from the vendor or is configured by the administrator when the
38 Printer is installed. Some of these resources can have associated opaque binary data,
39 such as font data, while others consist solely of attributes.

40 These ~~collection~~ Resource object attributes are retrieved using the ~~regular~~ (new)
41 GetPrinterAttributes ~~Get-Resource-Attributes~~ and Get-Resources operations which are
42 modeled on the IPP/1.1 Get-Job-Attributes and Get-Jobs operations and the Get-
43 Subscription-Attributes and Get-Subscriptions operations. Resource objects that can be
44 loaded are defined to have Resource Template attributes (just like Job and Subscription
45 objects), so that there are "xxx" Resource attributes and "xxx-supported" Printer
46 attributes, since they are ordinary collection attribute with an attribute syntax of '1setOf
47 collection'. More than one can be asked for in a request. All of the members of all of the
48 rows are returned (as per the current collections spec). Note: according to the collection
49 spec the "xxx-supported" Job Template attributes usually have the attribute syntax:
50 '1setOf type2 keyword', rather than '1setOf collection'. The keywords indicate which
51 member attributes are supported for the collection and the corresponding "xxx-supported"
52 indicate the values supported for each "xxx" member attribute. For such Job Template
53 attributes, a new naming convention is introduced: "xxx-rows" for the Printer attributes
54 with the attribute syntax of '1setOf collection'.

55 The "xxx-rows" (1setOf collection) attributes are never returned by Get-Printer-Attributes
56 unless they are explicitly asked for (i.e., they are never included in groups or 'all', since
57 there would be too much data in the response). A single row can be queried using a new
58 Get-Printer-Collection operation described in the next section.

59 The following new operations are defined for use with '1setOf collection'
60 attributes Resource objects:

- 61 • Get-Resource-Attributes - returns the requested attributes of the identified
62 Resource object instance.
- 63 • Get-Resources ~~Get-Printer-Collection-Rows~~ - return the requested attributes of the
64 Resource object instances ~~rows of a '1setOf collection' Printer attribute~~ based on a
65 simple filter supplied by the client
- 66 • Create-Resource ~~Add-Printer-Collection-Row~~ - add a Resource object instance
67 row to a '1setOf collection' ~~Printer attribute~~
- 68 • Delete-Resource ~~Delete-Printer-Collection-Row~~ - delete a Resource object
69 instance from the ~~row of a '1setOf collection' Printer attribute~~
- 70 • (new) Set-Resource-Attributes ~~Modify-Printer-Collection-Row~~ - modify a
71 Resource object instance ~~row of a '1setOf collection' Printer attribute~~
- 72 • Get-Resource-Data ~~Get-Printer-Collection-Row-Data~~ - same as Get-Resource-
73 Attributes ~~Get-Printer-Collection-Row~~, and in addition get the row's object
74 instance's associated opaque data.
- 75 • Create-Resource ~~Set-Printer-Collection-Row-Data~~ - same as Set-Printer-
76 Collection-Row, and in addition operation ~~sets~~ the row's object instance's
77 associated opaque data.

- 78 • Renew-Resource - update the lease time for the Resource object instance for those
79 Resource types that have leases.

80 For consistency all ~~six~~ seven operations have an Operation Attributes Group and a Printer
81 Resource Attributes Group in each request and response. The response always includes
82 the requested all of the member-Resource object attributes ~~of each row returned~~. In
83 addition to the usual request operation attributes for a Printer operation, all six operations
84 MUST include:

85 "resource-type" (type2 keyword) - which indicates the type of Resource, e.g.,
86 'media', 'font', 'image', 'input-tray', 'output-bin', etc.

87 Either "resource-name" (name(127)) or "resource-id" (integer(1:MAX)) -
88 identifies the resource object instance. The Printer MUST support both.

89 "collection-attribute" (type2 keyword) - which identifies the collection attribute to
90 be affected. For example: "collection-attribute" = 'font-rows-supported'
91 or "collection-attribute" = 'tray-rows-supported'

92 2 Get-Resource-Attributes and Get-ResourcesGet- 93 Printer-Collection-Rows operations

94 The Get-Resource-Attributes and Get-Resources operations for Resource objects follow
95 the pattern established by the IPP/1.1 Get-Job-Attributes and Get-Jobs operations for Job
96 objects and the Get-Subscription-Attributes and Get-Subscriptions operations for
97 Subscription objects. The Get-Resource-Attributes operation retrieves requested
98 attributes from one Resource object instance specified by the Key Attribute supplied by
99 the client.

100 The Get-Resources~~Get-Printer-Collection-Rows~~ operation retrieves requested attributes
101 from one or more ~~values~~ Resource object instances of a ~~1setOf~~ collection attribute. Each
102 collection value is called a "row". The ~~rows~~ Resource objects instances are selected on
103 the basis of a filter specified in the operation. Only one Filter Attribute is permitted and it
104 is expressed as ~~the only attribute in the~~ Printer-Resource Attributes group. The Printer
105 matches the Filter Attribute against all the ~~member~~ attributes of all of the ~~rows~~ Resource
106 object instances ~~in the collection value~~. The attribute name, syntax, and value of the Filter
107 Attribute MUST be the same as one of the ~~member~~ attributes in the ~~1setOf~~
108 collection Resource object instance, in order to match. A value match occurs if all of the
109 values of the Filter Attribute are a subset of the ~~member-Resource object~~ attribute ~~of a~~
110 row.

111 The client MUST also supply the Filter Attribute as the only attribute in a separate Printer
112 Resource Attributes group. For example:

113 "font-point-size" = '12'

114 The Printer returns ~~all member~~ the requested attributes of all Resource object instances
115 ~~rows of the collection~~ that match the Filter Attribute. Each row is returned in a separate
116 Printer-Resource Attributes group in the response (like Get-Jobs response). If no rows
117 match then the status code 'client-error-not-found' error is returned. **ISSUE: Or should**

118 the status code be 'successful-ok' (0), with an empty Printer-Resource Attributes group
119 returned to be more like Get-Printer-Attributes?

120 When a collection-Resource object type attribute with a !setOf collection attribute syntax
121 is defined, the definition SHOULD specify an 'identifying member-attribute', called the
122 Key Attribute that uniquely identifies an object instance-row. No two Resource object
123 instances of the same type rows can have the same Key Attribute value. The role of the
124 Key Attribute is the same as a primary key in a data base. The Key Attribute facilitates
125 direct indexing into !setOf collection-attributes-Resource object instances. Possible
126 examples could be tray name, media name, font name, etc. In some cases the identifying
127 member attribute could be a printer generated unique ID.

128 If a Resource object type definition a collection-attribute has a '!setOf collection' attribute
129 syntax, but the definition of that attribute does not indicate which member attribute is the
130 Key Attribute, that collection-attribute-Resource object type MAY still be used in the Get-
131 Resources-Get-Printer-Collection-Rows operation, but there is no way for the client to
132 unambiguously request a single object instance-row.

133 **3 Operations to Add, Delete, or Modify Resource object** 134 **instances-Collection rows**

135 The operations defined in this section add, delete, or modify a Resource object instance
136 for object types row in a !setOf collection-Printer-attribute that is defined to have a Key
137 Attribute.

138 These operations do not work on all Resource types-collections— there are some
139 collections-Resource types that represent state or non-logical capabilities of the device
140 (paper loaded, input trays, etc.). In this case the se collections-Resource instances are
141 read-only (either by definition or in a particular implementation).

142 There also can exist Printer-collections-Resource object attributes that represent collections
143 objects that are software modifiable entities but that are still not updated via these
144 operations in an implementation. For example fonts could be loaded by a specific set of
145 font management operations, rather than these operations.

146 What this means is that the collection-Resource object querying can be used on all
147 entities that are represented as '!setOf collection-Resource objects' but there can be many
148 mechanisms that create those collections-instances. The definition of the collection
149 Resource object type MUST indicate how the rows are created, modified, and removed.

150 **3.1 Create-Resource-Add-Printer-Collection-Row**

151 This operation adds a collection-row to an existing !setOf collection-Printer-attribute
152 Resource object instance provided that the collection-Resource object type definition
153 defined a Key Attribute.

154 In addition to the "collection-attribute" "resource-type" and either "resource-name" or
155 "resource-id" operation attributes, the client MUST supply the Key Attribute as the first
156 attribute in the Printer-Resource Attributes group in the request. For example, "font-
157 name" = 'TimesRomanItalic'. The client supplies the remaining attributes for the row

158 object instance as the remaining attributes in the Printer-Resource Attributes group. For
159 example, "font-size" = '12', "font-style" = 'italic', etc.

160 If the row-object instance already exists, the Printer MUST reject the request and return
161 the (new) 'client-error-row-already-exists'.

162 If the "collection-attributeresource-type" does not specify a '!setOf collection-attribute
163 Resource type whose definition includes a Key Attribute or the first attribute in the
164 Printer-Resource Attributes group is not the Key Attribute defined for the
165 collectionResource type, the Printer MUST reject the request with the 'client-error-bad-
166 request'.

167 **3.2 Delete-ResourceDelete-Printer-Collection-Row**

168 This operation deletes a collection row from an existing '!setOf collection-Printer
169 attribute-Resource object instance provided that the Resource object type collection
170 definition defined a Key Attribute.

171 In addition to the "resource-type" and either "resource-name" or "resource-id"
172 "collection-attribute"-operation attributes, the client MUST supply the Key Attribute as
173 the only attribute in the Printer-Resource Attributes group in the request. For example,
174 "font-name" = 'TimesRoman'.

175 If the row-object instance does not exist, the Printer MUST reject the request and return
176 the 'client-error-not-found' error status code.

177 If the "resource-typecollection-attribute" does not specify a '!setOf collection-attribute
178 Resource type whose definition includes a Key Attribute or the only attribute in the
179 Printer-Resource Attributes groups is not the Key Attribute defined for the
180 collectionResource type, the Printer MUST reject the request with the 'client-error-bad-
181 request'.

182 **3.3 Set-Resource-AttributesModify-Printer-Collection-Row** 183 **operation**

184 This attribute-operation modifies an existing Resource object instance collection row of
185 an '!setOf collection-Printer attribute-provided that the collection-Resource object type
186 definition defined a Key Attribute.

187 In addition to the "resource-type" and either "resource-name" or "resource-id"
188 "collection-attribute"-operation attributes, the client MUST supply the Key Attribute as
189 the first attribute in the Printer-Resource Attributes group in the request. For example,
190 "font-name" = 'TimesRomanItalic'. The client supplies the remaining attributes to be
191 modified for the row as the remaining attributes in the Printer-Resource Attributes group.
192 For example, "font-size" = '12', "font-style" = 'italic', etc. Any member-Resource
193 attributes of the row-object instance that the client omits are unchanged.

194 If the row-object instance does not exist, the Printer MUST reject the request and return
195 the 'client-error-not-found' error status code.

196 If the "collection-attributeresource-type" does not specify a '!setOf collection-attribute
197 Resource type whose definition includes a Key Attribute or the first attribute in the

198 Printer-Resource Attributes groups is not the Key Attribute defined for the
199 collectionResource type, the Printer MUST reject the request with the 'client-error-bad-
200 request'.

201 **3.4 Renew-Resource**

202 This operation renews the lease for the specified Resource object instance provided that
203 the Resource object type definition defined a Key Attribute.

204 In addition to the "resource-type" and either "resource-name" or "resource-id" operation
205 attributes, the client MUST supply the Key Attribute as the first attribute in the Resource
206 Attributes group in the request. For example, "font-name" = "TimesRomanItalic". The
207 client supplies the "xxx-lease-duration" attributes as the remaining attribute in the
208 Resource Attributes group. For example, "font-lease-duration" = nnn.

209 If the object instance does not exist, the Printer MUST reject the request and return the
210 'client-error-not-found' error status code.

211 If the "resource-type" does not specify a Resource type whose definition includes a Key
212 Attribute or the first attribute in the Resource Attributes groups is not the Key Attribute
213 defined for the Resource type or the Resource type definition does not include an "xxx-
214 lease-duration" attribute, the Printer MUST reject the request with the 'client-error-bad-
215 request'.

216 **4 Get-Resource-DataGet-Printer-Collection-Row-Data,** 217 **Create-ResourceSet-Printer-Collection-Row-Data** 218 **operations**

219 Some ~~'setOf collection' attributes~~ Resource types may have data associated with each
220 instance~~their rows~~. In this case then one choice available to the designer of the collection
221 Resource type is to use the Get-Resource-Data and Create-Resource ~~Get-Printer-~~
222 ~~Collection-Row-Data and Set-Printer-Collection-Row-Data~~ operations to read and write
223 opaque blobs (as well as the Resource object attributes).

224 For the Get-Resource-Data~~Get-Printer-Collection-Row-Data~~, the client supplies the
225 "collection-attribute~~resource-type~~" name and the Key Attribute as in the other Collection
226 Row-Resource operations. The requested Resource object row's attributes are returned in
227 the Printer-Resource Attributes Group, followed by the data as a data stream in the
228 response (packaged the same way the print-job's data is following the 'end-of-attributes-
229 tag').

230 The data is sent in the same way using the Create-Resource ~~Set-Printer-Collection-Row-~~
231 ~~Data~~ operation.

232 Note that for some collection-Resource types it might be possible to read the data but not
233 write it (uploading font metrics from ROM for example). Also it might be possible to
234 write it but not read it (macros are not intended to be used outside the printer so there is
235 not point in providing read capabilities).

236 Alternative design #1. There is a member attribute for each row that specifies a URI for
237 the data. The Add-Printer-Collection-Row operation returns the URI generated by the

238 ~~Printer when the row is created. The data is got and set by HTTP GET and POST to that~~
239 ~~URI. One problem with using HTTP GET and PUT to get or set data is the security~~
240 ~~checking. For example, ordinary users may not be able to add some resources to the~~
241 ~~Printer, such as media, but may be able to add other resources, such as images, to the~~
242 ~~Printer (for a leased amount of time).~~

243 Alternative design #12. Collections-Resource types that have associated data have
244 explicit row-object creation ~~and~~ operations (Load-Font operation for example) but the
245 data is read by HTTP get or ~~an IPP Get-Printer-Collection-Row-Data~~ Get-Resources and
246 Get-Resource-Attributes operations. This alternative overcomes the non-atomic nature of
247 adding a row then uploading the data.

248 5 Observations

249 Some collections-Resource types may have read-only rows-object instances and read-
250 write rows-object instances (fonts supported may include ROM fonts and soft fonts).

251 Jobs and Subscriptions could have been done using '1setOf collection-Printer
252 attributesResource objects of type 'job' and 'subscription'', but ~~since~~ we already have
253 operations defined for Jobs and Subscription objects.

254 Driver down loading could have been done using '1setOf collection-Printer
255 attributesResource objects of type 'driver'', but we have a specification that uses Get-
256 Printer-Attributes and a new Get-Client-Print-Support-Files operation.

257 Expiration times for collection-rows-Resource types can be specified in the collection
258 Resource type definition if that is what the collection-Resource type needs. For example,
259 if users are allowed to down load images into the Printer for a period of time.

260 This mechanism is only defining a standardized ways of viewing structured data – it does
261 not imply that common mechanisms must be used by implementations.

262 6 Suggested ~~member-Resource~~ attributes

263 In order to get some consistency in definition of '1setOf collection-Printer
264 attributesResource types', the following ~~member~~-attribute names and attribute syntaxes
265 are suggested if the ~~member~~-attribute is appropriate for the resource type. However, none
266 of these attributes are REQUIRED for a definition.

267 For key attributes that a client can supply (but cannot modify):

268 xxx-name (name(127)) or xxx-key (name(127) | type3 keyword) - Key Attribute
269

270 For attributes that a client can supply (or modify):

271 xxx-info (text(127)) - general information

272 xxx-create-date-time (dateTime) - the date and time that the resource was
273 originally created, not added to the Printer.

274 xxx-lease-duration (integer(0:MAX)) - lease duration in seconds, 0 is infinite

275 xxx-data-uri (1setOf uri) - uri of the data when supplied by the client

276 xxx-data-k-octets (integer(0:MAX)) - size of the data

277 xxx-data-compression (type3 keyword) - data compression

278
 279 For READ-ONLY attributes populated by the Printer:
 280 xxx-id (integer(1:MAX)) - integer id for those resources that do not have a natural
 281 name supplied by the client.
 282 xxx-create-user-name (name(MAX)) - user name who added the resource to the
 283 Printer
 284 xxx-create-time (integer(MIN:MAX)) - the "printer-up-time" when the resource
 285 was added to the Printer. A 0 or negative value means before this Printer
 286 power-up (see RFC 2911 section 4.3.14).
 287 xxx-expiration-time (integer(0:MAX)) - the "printer-up-time" when the lease
 288 expires
 289 xxx-data-uri (1setOf uri) - uri of the data when supplied by the Printer
 290

291 **7 Possible Resource types'1setOf collection' Printer**
 292 **attributes**

Collection	Identifying member Key Attribute	Members	Data
Input trays	name	Loaded media, state, capacity, level	none
Output bins	name	State, capacity, level	none
Fonts supported	Name (face-size-style)	Size, style, format	Font metrics
Media-descriptions	name	Size, weight,	none
Macros	name	Date, format	Macro data
Images	name	Date, format, description	Image data

293

294 **8 Examples**

295 These are examples of how this proposal could be used to represent various items. The
 296 full variety of choices is used. *These are not intended as actual proposals for their*
 297 *respective collections, but rather just indicate how the mechanism proposed in this paper*
 298 *would work.*

299 **8.1 Input-trays**

300 The "input-tray-rows-supported" (1setOf collection) Printer attribute contains one row for
 301 each input tray supported by the printer.

302 The rows are identified by an "input-tray-name" (type3 keyword | name(MAX)) Key
303 Attribute whose value is either defined by the PWG or is defined by the implementation.
304 The values of the "input-tray-name" member attribute may be submitted in a Job Creation
305 operation as the value of a (new) "input-tray" Job Template attribute.

306 The member attributes include "input-tray-max-capacity", "input-tray-current-level",
307 "input-tray-status", and "input-tray-media-name" with semantics taken from the Printer
308 MIB.

309 The rows of this collection are read using the Get-Printer-Collection-Rows operation.
310 Rows are not created or deleted, though in some implementations, certain member
311 attributes, such as "input-tray-media-name" can be set.

312 **8.2 Media descriptions**

313 The "media-rows-supported" (1setOf collection) Printer attribute contains one row for
314 each supported / known media. See the PWG Production Printing Extension spec for the
315 definition of the member attributes. The "media-col" (collection) Job Template attribute
316 can be supplied by the client in Job Creation operations.

317 The rows are identified by a "media-key" (type3 keyword | name(MAX)) Key Attribute
318 whose value is either defined by the PWG or is defined by the administrator. The values
319 of the "media-key" member attribute may be submitted in a Job Creation operation as the
320 value of the IPP/1.1 "media" Job Template attribute.

321 The member attributes include "media-size" (1setOf collection) {x-dimension, y-
322 dimension", "media-weight", "media-color", etc.

323 The rows of this collection are read using Get-Printer-Collection-Rows. Rows are created
324 using the Add-Printer-Collection-Row operation and deleted using the Delete-Printer-
325 Collection-Row operation.

326 **8.3 Images**

327 The "image-rows-supported" (1setOf collection) Printer attribute contains one row for
328 each supported / known.

329 The rows are identified by an "image-name" (name(MAX)) Key Attribute whose value is
330 either defined by its creator. The values of the "image-name" member attribute may be
331 submitted in a Job Creation operation as the value of the (new) "image" Job Template
332 attribute.

333 Member attributes include "image-size", "image-format", "image-version", etc. The
334 images are created by a (new) image-specific operation: Load-Image. This operation
335 includes all the member attributes that describe the image plus the image data as an
336 attached 'print-job' data stream. Some member attributes are derived from the image
337 (size for example).

338 The Set-Printer-Collection-Row-Data operation is not used. Instead, a (new) Load-
339 Image operation is defined and the client may specify an expiration time for the image.

340 Images may be explicitly deleted using the Delete-Printer-Collection-Row operation. The
341 Add-Printer-Collection-Row and Modify-Printer-Collection-Row operations are not
342 defined for use with images.
343 The image data is not readable externally.
344