

1 **File: Toward-resolving-Apples-JPS2-comments-4-thru-7.doc**

2
3 **From:** ipp-bounces@pwg.org on behalf of Michael Sweet [msweet@apple.com]

4 **Sent:** Monday, June 21, 2010 12:58

5 **To:** ipp@pwg.org

6 **Subject:** [IPP] Apple has reviewed the JPS2 specification and has comments

7
8 **Attachments:** ATT02705.txt

9 Comments:

10
11 1. The document date is wrong in the headers.

12 [TH> Updated to today's date.](#)

13
14 2. The "font-size-requested" attribute uses points as the units, however it should probably
15 use a higher-precision unit (at least decipoints, or 1/720th of an inch) to avoid issues with
16 getting common fixed-width font spacing right (e.g. 17 characters per inch with a
17 standard Courier font would need a point size of 5.293...) [And apologies for the late
18 notice on this - I haven't read through the older material in a while...]

19 [TH> See separate email thread and attached .pdf more readable version for this issue and](#)
20 [possible alternatives for resolution.](#)

21
22 3. Question: do we need the PWG process reference at the beginning of section 3? Do we
23 need to explain why we organize our standards documents this way?

24 [TH> Need IPP WG input here.](#)

25
26 4. On line 372 of the -rev document you say "these features are useful in the "light
27 production printing" environments, but we don't define what "light production printing"
28 is, just "production printing".

29 [TH> Just remove the clause entirely, OK?](#)

30
31 5. Line 427 - I think Close-Job is Owner + Operator, per RFC 2911's description of Send-
32 Document and the Close-Job references in JPS2:

33
34 Access Rights: The authenticated user (see section 8.3) performing
35 this operation must either be the job owner (as determined in the
36 Create-Job operation) or an operator or administrator of the Printer
37 object (see Sections 1 and 8.5). Otherwise, the IPP object MUST
38 reject the operation and return: 'client-error-forbidden', 'client-
39 error-not-authenticated', or 'client-error-not-authorized' as
40 appropriate.

41 [TH> Added:](#)

42 [Access Rights: The authenticated user \(see \[RFC2911\] section 8.3\) performing this operation](#)
43 [must either be the job owner \(as determined in the Create-Job operation\) or an operator or](#)
44 [administrator of the Printer object \(see \[RFC2911\] Sections 1 and 8.5\). Otherwise, the IPP object](#)
45 [MUST reject the operation and return: 'client-error-forbidden', 'client-error-not-authenticated', or](#)
46 ['client-error-not-authorized' as appropriate.](#)

47
48 6. Line 567-568: The definition says up to 255 characters for job-password and job-
49 password-supported, but here we say 127 octets.

50 | TH> Fixed

51

52 | 7. a: Section 10.5 (job-spooling-supported) - shouldn't this be a 1setOf?

53

54 | b: Also, lines 580-581 basically says that the values are implementation-dependent
55 | regardless of whether the attribute is supported or not. :)

56

57 | c: Based on the current description I think we need to specify when the value is 1setOf
58 | and when it is not (i.e. a Get-Printer-Attributes request with a document-format would
59 | return the value for that format, otherwise you'd get the value for the default format or a
60 | 1setOf if the answer can't be made definitively without a format...)

61

62 | d: This will also affect the registration info on line 1073.

63

64 | TH> Apple's 4 suggestions (which I've labeled a thru d) are that a Printer implementation
65 | MAY vary the spooling strategy based on the document format. For example, a
66 | document format that might have backward references would require the Printer to spool,
67 | while one that cannot, the Printer might stream. Therefore, even though there is no way
68 | for the client to dictate to the Printer which strategy to use, since there isn't a Job
69 | Template attribute to recommend/require the spooling strategy (unless we want to add
70 | one), the Printer could support more than one value of "job-spooling-supported",
71 | depending on the document format(s) of the document(s) in the submitted job.

72

73 | I've taken a stab at clarifying that "job-spooling-supported" can depend on the document
74 | format supplied in the job, as suggested by Apple comment #7:

75

76 | **10.5 job-spooling-supported (1setOf type2 keyword)**

77 | This attribute indicates whether or not jobs are spooled before the document data is
78 | interpreted (RIPped). In other words, this attribute indicates when jobs are processed by
79 | the Printer with respect to when the Printer receives and returns responses to Job
80 | Creation requests (i.e., Print-Job, Print-URI), receives and returns responses to
81 | Document Creation requests (i.e., Send-Document and Send-Uri requests) and
82 | "receives" or "fetches" such document data.

83 | The value of this attribute returned in a Get-Printer-Attributes response MAY
84 | depend on the "document-format" attribute supplied in the Get-Printer-Attributes
85 | request (see [RFC2911] Section 3.2.5.1 and 6.2).

86 | If the Printer supports this attribute, the value supported depend on implementation. If
87 | the Printer does not support this attribute, then the spooling behavior is implementation
88 | dependent.

89 | The Get-Printer-Supported-Values operation (see description in [RFC3380]) returns a
90 | '1setOf type2 keyword' so that all possible values that the implementation is capable of
91 | supporting are indicated.

92 | The standard keyword values are:

Keyword	Description
---------	-------------

Comment [th1]: **ISSUE:** OK to add "1setOf" to "job-spooling-supported" as suggested by Apple's IPP WG Last Call Comment #7a?

Comment [th2]: **ISSUE:** Comment made during the June 21 telecon:

Add: "job-spooling-configured" (type2 keyword).

But there is no mention in the minutes.

Also with the Apple's Comment #7 suggestion that the spooling strategy could depend on the document-format in the submitted job, configuring for a single strategy wouldn't work.

OK NOT to add "job-spooling-configured"?

Comment [th3]: **ISSUE:** OK to add that the spooling used MAY depend on the document format as suggested by Apple's Comment #7 and allowed by [RFC2911] Section 3.2.5.1 as long as explicitly called out in the definition as indicated in [RFC2911] Section 6.2?

Comment [th4]: **ISSUE:** OK to add that the spooling used MAY depend on the document format as suggested by Apple's Comment #7c, but refer to RFC 2911 Section 3.2.5.1 for the details about what is returned in Get-Printer-Attributes response depending on whether or not the client supplies "document-format: in the request?"

Comment [th5]: **ISSUE:** OK to remove this sentence as suggested by Apple's Comment #7b?

Comment [th6]: **ISSUE:** OK to remove this paragraph as suggested by Apple's Comment 7c?

'spool'	The Printer starts processing a job until the Printer has (1) accepted and responded to the Job Creation request and all Document Creation requests (for a multi-document job) and (2) has "received" or "fetched" all document data for the job, i.e., spool rather than stream.
'stream'	The Printer starts processing a job (1) before the Printer has accepted all Document Creation requests and (2) before the Printer has "received" or "fetched" all document data, i.e., stream rather than spool
'automatic'	The Printer chooses whether to process a job before or after the Printer has accepted all Document Creation requests and has "received" or "fetched" all document data, i.e., the Printer MAY spool and/or stream depending on policy and other factors, such as the size of the job, and the document format, including a combination of spooling and streaming.

Comment [th7]: ISSUE: OK to add that "document-format" could be an example that causes the Printer to vary the spooling strategy as suggested by Apple's Comment #7c?

93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126

TH> However, I don't think we need to get into the details about Get-Printer-Attributes vs. Get-Printer-Attributes-Supported, since it is now clear that multiple values MAY be returned in either or both operations, depending on implementation. [RFC2911] Section 3.2.5.1 Get-Printer-Attributes goes into gory detail about attributes whose values are affected by "document-format" supplied in the Get-Printer-Attributes requests:

"document-format" (mimeType):

The client OPTIONALLY supplies this attribute. The Printer object MUST support this attribute. This attribute is useful for a Printer object to determine the set of supported attribute values that relate to the requested document format. The Printer object MUST return the attributes and values that it uses to validate a job on a create or Validate-Job operation in which this document format is supplied. The Printer object SHOULD return only (1) those attributes that are supported for the specified format and (2) the attribute values that are supported for the specified document format. By specifying the document format, the client can get the Printer object to eliminate the attributes and values that are not supported for a specific document format. For example, a Printer object might have multiple interpreters to support both 'application/postscript' (for PostScript) and 'text/plain' (for text) documents. However, for only one of those interpreters might the Printer object be able to support "number-up" with values of '1', '2', and '4'. For the other interpreter it might be able to only support "number-up" with a value of '1'. Thus a client can use the Get-Printer-Attributes operation to obtain the attributes and values that will be used to accept/reject a create job operation.

If the Printer object does not distinguish between different sets of supported values for each different document format when validating jobs in the create and Validate-Job operations, it MUST NOT distinguish between different document formats in the Get-Printer-Attributes operation. If the Printer object does distinguish between different sets of supported values for each different document format specified by the client, this specialization applies only to the following Printer object attributes:

- 127 - Printer attributes that are Job Template attributes ("xxx-default" "xxx-
128 supported", and "xxx-ready" in the Table in Section 4.2),
129 - "pdl-override-supported",
130 - "compression-supported",
131 - "job-k-octets-supported",
132 - "job-impressions-supported",
133 - "job-media-sheets-supported"
134 - "printer-driver-installer",
135 - "color-supported", and
136 - "reference-uri-schemes-supported"
137

138 The values of all other Printer object attributes (including "document-format-
139 supported") remain invariant with respect to the client supplied document format
140 (except for new Printer description attribute as registered according to section
141 6.3).
142

143 If the client omits this "document-format" operation attribute, the Printer object
144 MUST respond as if the attribute had been supplied with the value of the Printer
145 object's "document-format-default" attribute. It is RECOMMENDED that the
146 client always supply a value for "document-format", since the Printer object's
147 "document-format-default" may be 'application/octet-stream', in which case the
148 returned attributes and values are for the union of the document formats that the
149 Printer can automatically sense. For more details, see the description of the
150 'mimeMediaType' attribute syntax in section 4.1.9.
151

152 If the client supplies a value for the "document-format" Operation attribute that is
153 not supported by the Printer, i.e., is not among the values of the Printer object's
154 "document-format-supported" attribute, the Printer object MUST reject the
155 operation and return the 'client-error-document-format-not-supported' status code.
156
157

158
159
160 Michael Sweet, Senior Printing System Engineer, PWG Chair
161
162

163
164
165 --
166 This message has been scanned for viruses and
167 dangerous content by [MailScanner](#), and is
168 believed to be clean.