1 Subj: Strawman Requirements for encoding IPP protocol using XML
2 Date: 01/13/98
3 Files: ftp://ftp.pwg.org/pub/pwg/ipp/new_PRO/ipp-pro-requirements.pdf   .doc   .txt
4 Version: 0.1
5

6 To those developing XML encoding schemes for IPP:

7 Here are the coding requirements that our current IPP Protocol document meets. A
8 proposal for encoding IPP using XML should meet these requirements too.

9 The current IPP protocol encoding specification is available as an Internet-Draft as:

10   <draft-ietf-ipp-protocol-04.txt>.

11 The 05 version has been forwarded to the Internet-Drafts DL and will be out shortly. It is
12 currently available at:

13   ftp://ftp.pwg.org/pub/pwg/ipp/new_PRO/ipp-pro-980109.pdf

14 The current IPP Model and Semantics specification is available as an Internet-Draft as:

15   <draft-ietf-ipp-model-08.txt>

16   ftp://ftp.pwg.org/pub/pwg/ipp/new_MOD/ipp-model-971219.pdf

17 Several requirements have alternative requirements listed immediately following, with an
18 "a" appended. We need to decide which requirement should be the one we want to have.

19 The term "recipient" is used to indicate both a client and an IPP object, since an IPP
20 object receives requests from a client and a client received responses from an IPP object.

Strawman Requirements for encoding IPP protocol using XML

22 The encoding of IPP in XML shall meet the following requirements:

23 1. Do not need to change the IPP Model document at all, or only at least trivially. It has
24   all of the semantics that we want to represent.

25 2. Be registered as a MIME media type, that can be transmitted with any transport, such
26   as SMTP, rather than being restricted to HTTP 1.1, i.e., be registered with Intended
27   usage: COMMON.

28 3. Be able to represent all of the IPP operations: Print-Job, Print-URI, Validate-Job,
29   Create-Job, Get-Printer-Attributes, Get-Jobs, Send-Document, Cancel-Job, and Get-
30   Job-Attributes. See section 4.2 to 4.4 of the Model document. Each operation has a
31   defined request format and a defined response format, consisting of groups of
32   attributes.

33 4. In addition, the Print-Job and Send-Document requests have appended document data
34   which must be easily separable by the recipient.

35 5. The appended document must be able to be any text or binary document format,
36   including an XML document itself.

37 6. Also the Send-Document request must be able to be sent with no document (if this is
38   a problem for XML, we could add a new operation, say, Close-Job, to close a multi-
39   document job without sending a document).

40 7. It must be possible for any version of a recipient to detect that a request or response is
41   a later or earlier version for all future versions.

42 8. Need to be able to represent all of the attribute syntaxes (data types) that are in
43   Section 4.1 of the Model document: 'text', 'textWithLanguage', 'name',
44   'nameWithLanguage', 'keyword', 'enum', 'uri', 'uriScheme', 'charset', 'naturalLanguage',
45   'mimeMediaType', 'octetString', 'boolean', 'integer', 'rangeOfInteger', 'dateTime',
46   'resolution', and '1setOf X' (multi-valued attributes). A 'dictionary' value is reserved
47   for the future.

48

49   82a. A different, more natural to XML, method for overriding the natural language for
50   'text' and 'name' attributes than using the 'textWithLanguage' and
51   'nameWithLanguage' attribute syntaxes would be acceptable.

52   8b. A different, more natural to XML, method for grouping a bunch of attributes as
53   the value of an attribute would be acceptable to having a 'dictionary' attribute syntax.

54 9. Keywords are used to identify attributes and attribute values and are specified in
55   lower case US-ASCII and U.S. English and allow hyphen (-), dot (.) and underscore
56   (_).

57

58   93a. Using keywords to identify attribute syntaxes would be desirable, though not
59   currently done in our current IPP protocol.

60 10. Implementers must be able to add private keywords using private keyword syntax for
61   which clash with other implementers' private keywords is not possible. Implication:
62   a recipient must be able ignore any attributes (and values) not recognized and/or
63   supported and continue parsing a request or response.

64 11. The PWG must be able to add additional attribute syntaxes following a registration
65   procedure that includes PWG review. Implication: a recipient must be able ignore
66   any attributes (and values) not recognized and/or supported and continue parsing a
67   request or response.

68 12. Implementers must be able to include private attribute syntaxes in conforming
69   interchange. Implication: a recipient must be able ignore any attribute syntaxes not
70   supported and continue parsing a request or response.

71   106a. It would be desirable if clashes with other implement private attribute syntaxes
72   could be guaranteed to be avoided - something that our current IPP encoding does not
73   solve, since attribute syntaxes are encoded as integers without a registration scheme.

74 Using keywords for attribute syntaxes would be a way to achieve such name clash
75 avoidance.

76 13. Implementers must be able to add private enum values.

77 14. Attributes must be identified by keywords in US-ASCII and US-English.

78 15. Some attribute syntaxes must be able to be defined to be a value that contains several
79 distinct specific data types in a specific order, like a C struct. Currently, each such
80 value must be present, there is no need for values to be optional in such a structure.

81 16. Attributes must be able to be multi-valued, i.e., to have multiple instances of the
82 value.

83 17. The syntax specification must be able to indicate which attribute are permitted to be
84 multi-values, and which must be single valued.

85 18. Any attribute must be able to have "out-of-band" values, i.e., values that can be used
86 with any attribute and attribute syntax. Currently, we have the following "out-of-
87 band" values: 'unsupported', 'unknown', 'no-value' with 'default' reserved for the
88 future.

89 19. Some attributes must be able to have more than one attribute syntax defined for them,
90 such as 'keyword' and 'name' (or 'text' and 'textWithLanguage' if that mechanism is
91 used for overriding natural language on an attribute value by attribute value basis).

92 20. An instance of a multi-valued attribute must be able to employ any of the attribute
93 syntaxes defined for it. For example be able to mix 'keyword' and 'name' values.

94 21. Each value of an attribute needs to identify which attribute syntax it is using.

95 22. The 'text' and 'name' attribute syntaxes must be able to represent any ISO 10646
96 character, probably using utf-8 encoding.

97 23. The 'text' and 'name' attribute values must be representable in other charsets than
98 utf-8. Possibly, some simple coding transformation may be required by the client for
99 some charsets. There is no need for this to be done on an attribute by attribute basis.
100 Only a single request or a single response needs to be in a charset specified in the
101 interchange.

102 24. It must be possible to group attributes in an interchange into several groups.

103 For requests: Operation attributes and Job Template attributes.

104 For responses: Operation attributes, Job Attributes, Printer Attributes, Job
105 Description Attributes, Printer Descriptions Attribute, Unsupported Attributes.

106 25. Attribute groups must be identified as to which group.

107 26. The syntax definition should restrict the order of occurrence of attribute groups.

108 27. The recipient should be able to ignore entire attribute groups that are not recognized
109 and/or supported.

110   28. No syntactic distinction is needed between multi-valued attributes that are ordered,
111   and ones that are not. The semantics specify which attributes are ordered and which
112   are not (most are not).

113   29. Similarly, no syntactic distinction need be made between attribute groups that require
114   certain attributes to be first and ones that do not. The semantics specify which
115   attributes must come first in which groups. Most attribute can occur anywhere in the
116   groups in which they are permitted.