

IEEE-ISTO

Industry Standards and Technology Organization
affiliated with the IEEE and the IEEE Standards Association

[Template for:]

The Printer Working Group (PWG) Working Draft of Proposed Standard for The Internet Printing Protocol (IPP): Rest of your title

[This PWG template for creating IEEE-ISTO specifications includes instructions for you to use. These instructions are highlighted in gray and bracketed like this. Text highlighted in yellow like this is to be filled in by you. A few IPP-specific instructions and suggested text are highlighted in green like this to provide specific examples. Always include a title, URL, date, version number on the first page by updating the fields that are highlighted like this in the template. Leave these fields highlighted, so you won't miss them when editing the next version. For IPP, add a separate line after the second line: "The Internet Printing Protocol (IPP):"]

Abstract: This specification defines a template for PWG Working Drafts for IEEE-ISTO Proposed Standards.

[Write a short abstract that doesn't overflow the page and will help the reader figure out whether or not to read your specification. The Abstract section SHOULD provide a concise and comprehensive overview of the purpose and contents of the entire specification, to give a technically knowledgeable reader a general overview of the function of the specification. Start with "This specification defines ..."
For IPP extensions: "This specification extends the Internet Printing Protocol (IPP/1.0, IPP/1.1 and future versions). This extension provides ..."
Expand any acronyms in their first occurrence in the Abstract. Don't just repeat the first few paragraphs of your Introduction; be briefer in the Abstract.]



This version of this specification is available electronically at:
<ftp://ftp.pwg.org/pub/pwg/general/templates/-v2-030127.pdf>, .doc

Editor: Tom Hastings
Version 0.2, 27 January 2003

Don't use an automatically generated current date, so that it is clear when the specification was last modified. Use fractional version numbers until after a Last Call has passed. Always publish both a .doc and a .pdf file. You SHOULD also publish a -rev.doc and a -rev.pdf version that shows the revisions from the previous version as tracked changes so your reviewers can see the changes.



IEEE

Industry Standards and Technology Organization (IEEE-ISTO)

445 Hoes Lane • P.O. Box 1331 • Piscataway, NJ 08855-1331, USA
Phone +1.732.981.3434 • Fax +1.732.562.1571 • <http://www.ieee-isto.org>

33 *This Template contains each of the sections that your PWG Proposed IEEE-ISTO standard MUST have.*
34 *The Template also contains the instructions at the beginning of each section (10% gray highlighted, bold*
35 *italics, set off with square brackets). You MAY want to retain these instructions in the first few versions*
36 *of your specification if your reviewers are new to this type of specification, so that early reviewers can*
37 *compare what you wrote with these instructions. On the other hand, reviewers who are familiar with*
38 *this type of specification will find the instructions annoying, so delete them in the first version. Or for*
39 *early working drafts, you MAY leave some of the sections to be done later, so you SHOULD leave these*
40 *unfilled sections with the instructions included, rather than deleting the sections altogether. To delete*
41 *all of these highlighted instructions, Replace All style Body Text Instr with nothing (assuming that you*
42 *haven't used this style by mistake in your writing). A companion specification contains some Tips for*
43 *Good Technical Writing that you SHOULD read first (see*
44 *<ftp://ftp.pwg.org/pub/pwg/general/templates/good-writing-tips-latest.pdf>).*

45

46

47

48 *[You MUST name your file following one of the following formats to be chosen by your PWG WG:*

49 *<ftp://ftp.pwg.org/pub/pwg/topic/pwg-topic-aaa-bbb-vn-yymmdd.pdf>, .doc*
50 *<ftp://ftp.pwg.org/pub/pwg/topic/subtopic/pwg-topic-aaa-bbb-vn-yymmdd.pdf>, .doc*

51 *where topic is your WG acronym if this is not an individual contribution, subtopic is a sub-directory, and*
52 *aaa-bbb is a short series of words, acronyms, or abbreviations. Your WG SHOULD create sub-*
53 *directories for separate sub-topics, if you are producing more than one standard. The file name and*
54 *sub-directory MUST contain only hyphen (-), period (.), and all lower-case letters (helps with*
55 *interoperability across different file systems). The file name MUST NOT have underlines (hypertext*
56 *links cover up underlines, so they are indistinguishable from spaces), MUST NOT have spaces (not*
57 *supported in most file systems and most protocols) and MUST NOT have digits (think of more*
58 *meaningful names). Do not include digits, but name the contents meaningfully instead. The filename*
59 *extension, e.g., .pdf, .txt, MUST be present and MUST correctly identify the format.*

60 *The total length of the URL SHOULD be less than 72 characters, else email will line wrap the URL and*
61 *some users will not be able to figure out how to get the file. Don't assign an IEEE-ISTO number until*
62 *after the first PWG approval Last Call to progress the specification to a PWG Proposed Standard. If you*
63 *need more than 9 Working Drafts, number the Working Drafts following 0.9, as: 0.10, 0.11, etc., rather*
64 *than being strictly numeric fractions.*

65

65 *[PWG Working Drafts, Proposed Standards, Draft Standards and Standards MUST include this*
66 *copyright notice as the second page of the document:]*

67 **Copyright (C) 2002, IEEE ISTO. All rights reserved.**

68 This specification may be copied and furnished to others, and derivative works that comment on,
69 or otherwise explain it or assist in its implementation may be prepared, copied, published and
70 distributed, in whole or in part, without restriction of any kind, provided that the above copyright
71 notice, this paragraph and the title of the specification as referenced below are included on all
72 such copies and derivative works. However, this specification itself may not be modified in any
73 way, such as by removing the copyright notice or references to the IEEE-ISTO and the Printer
74 Working Group, a program of the IEEE-ISTO.

75 The IEEE-ISTO and the Printer Working Group DISCLAIM ANY AND ALL WARRANTIES,
76 WHETHER EXPRESS OR IMPLIED INCLUDING (WITHOUT LIMITATION) ANY IMPLIED
77 WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

78 The Printer Working Group, a program of the IEEE-ISTO, reserves the right to make changes to
79 the specification without further notice. The specification may be updated, replaced or made
80 obsolete by other specifications at any time.

81 The IEEE-ISTO takes no position regarding the validity or scope of any intellectual property or
82 other rights that might be claimed to pertain to the implementation or use of the technology
83 described in this specification or the extent to which any license under such rights might or might
84 not be available; neither does it represent that it has made any effort to identify any such rights.

85 The IEEE-ISTO invites any interested party to bring to its attention any copyrights, patents, or
86 patent applications, or other proprietary rights which may cover technology that may be required
87 to implement the contents of this specification. The IEEE-ISTO and its programs shall not be
88 responsible for identifying patents for which a license may be required by a specification and/or
89 IEEE-ISTO Industry Group Standard or for conducting inquiries into the legal validity or scope of
90 those patents that are brought to its attention. Inquiries may be submitted to the IEEE-ISTO by e-
91 mail at:

92 ieee-isto@ieee.org.

93 The Printer Working Group acknowledges that the IEEE-ISTO (acting itself or through its
94 designees) is, and shall at all times, be the sole entity that may authorize the use of certification
95 marks, trademarks, or other special designations to indicate compliance with these materials.

96 Use of this specification is wholly voluntary. The existence of this specification does not imply
97 that there are no other ways to produce, test, measure, purchase, market, or provide other goods
98 and services related to its scope.

99

99 About the IEEE-ISTO

100

101 The IEEE-ISTO is a not-for-profit corporation offering industry groups an innovative and flexible
102 operational forum and support services. The IEEE-ISTO provides a forum not only to develop
103 standards, but also to facilitate activities that support the implementation and acceptance of
104 standards in the marketplace. The organization is affiliated with the IEEE (<http://www.ieee.org/>)
105 and the IEEE Standards Association (<http://standards.ieee.org/>).

106

107 For additional information regarding the IEEE-ISTO and its industry programs visit
108 <http://www.ieee-isto.org>.

109

110

111 About the IEEE-ISTO PWG

112 The Printer Working Group (or PWG) is a Program of the IEEE Industry Standards and
113 Technology Organization (ISTO) with member organizations including printer manufacturers, print
114 server developers, operating system providers, network operating systems providers, network
115 connectivity vendors, and print management application developers. The group is chartered to
116 make printers and the applications and operating systems supporting them work together better.
117 All references to the PWG in this specification implicitly mean “The Printer Working Group, a
118 Program of the IEEE ISTO.” In order to meet this objective, the PWG will publish the results of
119 their work as open standards that define print related protocols, interfaces, procedures and
120 conventions. Printer manufacturers and vendors of printer related software will benefit from the
121 interoperability provided by voluntary conformance to these standards.

122 In general, a PWG standard is a specification that is stable, well understood, and is technically
123 competent, has multiple, independent and interoperable implementations with substantial
124 operational experience, and enjoys significant public support.

125 For additional information regarding the Printer Working Group and joining current mailing lists,
126 visit:

127

<http://www.pwg.org>

128

129

129 **Table of Contents**

130 1 Introduction 7

131 2 Terminology 7

132 2.1 Conformance Terminology 7

133 2.2 Other Terminology..... 8

134 **3 <body of standard. Many sections at header1 level>** 11

135 3.1 *Conformance Level(s), Modules, and/or Profiles* 14

136 4 Conformance Requirements for **<fill in part of your title>** 15

137 **4.1 Printer** conformance requirements 15

138 **4.2 Client** conformance requirements 16

139 5 References..... 16

140 5.1 Normative References 16

141 5.2 Informative References 17

142 6 IANA Considerations..... 18

143 **6.1 Attribute Registrations** 18

144 **6.2 Keyword attribute value registrations** 19

145 **6.3 Enum Attribute Value Registrations** 19

146 **6.4 Attribute syntax Registrations** 20

147 **6.5 Operation Registrations** 20

148 **6.6 Attribute Group tag Registrations** 20

149 **6.7 Status code Registrations** 20

150 **6.8 Registration of Events** 21

151 7 Internationalization Considerations..... 21

152 8 Security Considerations 21

153 9 Contributors 22

154 10 Acknowledgments..... 22

155 11 Editor Contact Information 22

156 Appendix A <fill-in a title for Annex> (Normative) 23

157 Appendix B <fill-in a title for Annex> (Informative)..... 23

158 Appendix C Differences between this Template and the IEEE-ISTO standard style 23

159 Appendix D Change Log (Informative)..... 24

160 D.1 Changes to make version 0.1, November 14, 2002..... 24

161 D.2 Changes to make version 0.2, January 27, 2003 24

162

163 **Table of Tables**

164 Error! No table of figures entries found.

165 **Table of Figures**

166 Figure 1 - Logical Requester - Provider Diagram **for IPP** 12

167 Figure 2 - Leaf IPP Printer objects 13

168 Figure 3 - Non-Leaf Printer objects and their Subordinate Printer objects 14

169

169

170 1 Introduction

171 *[You MUST write a short introduction which gives an overview of the problem being addressed and a*
172 *high level description of the solution. Your Introduction MUST be self-contained as if there were no*
173 *Abstract section.]*

174 *In order to write a specification that is intended to facilitate interoperation between implementations*
175 *from different vendors, you MUST define an Interface that separates the implementations of two*
176 *vendors. Typically, there will be a Requester implementation and a Provider implementation of your*
177 *Interface. The purpose of your specification is to define the semantics of your Interface and usually the*
178 *syntax. In order to achieve interoperability between vendors' implementations, your specification MUST*
179 *define conformance requirements for the Requester and for the Provider such that two conforming*
180 *implementations will interoperate. [You SHOULD include the scope that is covered in the specification,*
181 *that is, summarize what is covered. Sometimes it is helpful to include what is not covered as well, in case*
182 *some readers might make the wrong assumptions and be disappointed.]*

183 **This specification extends the Internet Printing Protocol IPP/1.0 [RFC2566, RFC2565], IPP/1.1**
184 **[RFC2911, RFC 2910], and all future versions by defining ...**

185 2 Terminology

186 *[You MUST include a terminology section that has both Conformance terms and other terminology.]*

187 *You MUST NOT repeat definitions that are defined in referenced specifications. Otherwise, the*
188 *definitions may get out of synch and/or the reader may think you are redefining the terms. If you use*
189 *the terms defined in other specifications, just list them here and state that your specification uses these*
190 *same terms with the same meanings.]*

191 This section defines terminology used throughout this specification.

192 2.1 Conformance Terminology

193 *[You MUST understand and use correctly the conformance terminology as defined in [RFC2119].*

194 *You MUST capitalize the entire conformance term so that it is easy for readers to find them and you will*
195 *think more carefully whether to use MUST, SHOULD, or MAY when you write each statement.*

196 *Each conformance statement throughout your specification MUST be clear as to whether it applies to*
197 *the Requester and/or the Provider of your Interface. So avoid statements like: "The xxx attribute is*
198 *REQUIRED". Instead, use the verb form with a subject such as: "A Requester MUST supply the xxx*
199 *attribute or the Provider MUST support the xxx attribute."]*

200 Capitalized terms: **MUST, MUST NOT, REQUIRED, RECOMMENDED, SHOULD, SHOULD**
201 **NOT, MAY, and OPTIONAL**, have special meaning relating to conformance as defined in
202 [RFC2119]. In order to claim conformance to this specification, an implementation MUST
203 observe the semantics of all capitalized conformance terms in this specification as defined in
204 [RFC2119]. These capitalized terms apply to conformance to this specification only and do not
205 apply to conformance to other specifications.

206 2.2 Other Terminology

207 *[You MUST define any additional terms that your specification uses.*

208 *You MUST define the term that you will use for your Interface. For IPP specifications, the Interface is*
209 *usually the IPP protocol. However, if you are defining a profile of IPP for a specific purpose, such as a*
210 *Printer to Notification Server, then you MUST define a term for that protocol.*

211 *You MUST define the terms that you will use for the Requester and Provider of your Interface.*
212 *Normally, for IPP the terms “client” and “Printer” defined in [RFC2911] will suffice. However, if you*
213 *are defining a protocol between, say, a client and a Notification Server or between a Printer and a*
214 *Notification Server, then you need to define these terms in this section and identify which is the*
215 *Requester and which is the Provider of your Interface.*

216 *On the other hand, if your specification covers two or more Interfaces, then you MUST define separate*
217 *terms for each Requester and each Provider of your Interface, even if a single implementation can be a*
218 *Requester and/or a Provider of more than one of your Interfaces. Then each conformance statement is*
219 *clear as to which Interface and to which side of that Interface that conformance statement applies. For*
220 *example, see Figure 1 which has both a Protocol Interface and a UI Interface.*

221 *Your terms SHOULD be the ones that are commonly used in the industry, rather than being different*
222 *words. Some authors misguidedly use different words in order to avoid confusion with the common*
223 *words. However, that makes their documents much harder to understand. Instead, use the commonly*
224 *used words and define them in this terminology section with the specialized meaning that you want. For*
225 *example, the work “attribute” is commonly used to indicate a named feature of an object with a data type*
226 *and one or more values. You SHOULD define Attribute in your specification and define exactly what it*
227 *means in your specification, possibly saying how it differs from that term used in other contexts. If*
228 *necessary, you can put an adjective as part of your term or in front of the other usage, if you have to use*
229 *it in your specification. For example, XML uses the term attribute in a specialized way as part of an*
230 *XML tag with the syntax: attribute-name = value1 value2. If you need to refer to the XML attribute and*
231 *attributes in your specification, define XML Attribute and Attribute as two distinct terms in your*
232 *specification, rather than defining some other term for the Attribute introduce in your specification.*

233 *Your definition SHOULD identify the class of things to which the term belongs in the beginning of the*
234 *first phrase followed by “that ...” that explains the specialization of the defined term within the broader*
235 *class of things. See the definitions below for examples of definition writing.*

236 *You MUST capitalize the first letter of each word in your terms throughout the specification, so that the*
237 *reader knows that you are referring to the specially defined term.*

238 *You MUST bold the terms in the Terminology section, so that they are easier to spot for the reader. You*
239 *SHOULD NOT bold the terms in the rest of the specification.*

240 *You SHOULD list the terms in alphabetic order for ease of reference, unless the number of terms is very*
241 *small.*

242 *You SHOULD use the phrase “this specification” to refer to your document, rather than “this*
243 *document” since IPP specifications also talk about “documents” that are being submitted for printing by*
244 *an IPP Printer. Also do not refer to your specification as “this standard”, since as a PWG Working*
245 *Draft, or a PWG Proposed Standard, it isn’t yet a standard.*

246 *Here are some example terms that you SHOULD use unless you have good reason to use others. Delete*
247 *any of these terms that you do not use]*

248 This specification uses the same terminology as [RFC2911], such as “client”, “Printer”,
249 “attribute”, “Operation attribute”, “Job Template attribute”, “attribute value”, “keyword”,
250 “operation”, “request”, “response”, and “support” document with the same meaning. Note:
251 [RFC2911] doesn't completely conform to the capitalization of terms. In addition, the following
252 terms are defined for use in this specification:

253 **Application Program Interface (API)** - an **Interface** consisting of an application program calling
254 a **Function** or **Procedure** using a well-defined **Function** or **Procedure** calling sequence with
255 well-defined input parameters, output parameters, possibly, a **Function** value, and possibly
256 thrown exception conditions.

257 **Conformance Level (Level for short)** - a set of **REQUIRED** and **OPTIONAL Features** of an
258 **Interface** for a **Provider** to support that provide a certain level of service. If an **Interface** has
259 multiple **Conformance Levels**, each higher **Conformance Level** is a superset of all the lower
260 levels.

261 **Conformance Module (Module for short)** - a set of **REQUIRED** and **OPTIONAL Features** of an
262 **Interface** for a **Provider** to support that provide a certain grouping of service.

263 **Conformance Profile (Profile for short)** - a set of **Conformance Modules** that together are
264 **REQUIRED** for conformance. A specification **MAY** define multiple **Conformance Profiles**.
265 Such Conformance Profiles do not need to have a sub-set/super-set relationship. If they do, then
266 they are called Conformance Levels instead. In order to claim conformance for an
267 implementation, such a claim **MUST** indicate which Conformance Profile or Profiles are
268 supported.

269 **Customer** - a user who purchases a **Provider** product or a **Requester** product from a **Vendor**. It
270 is the Customer who is concerned about Interoperability between **Provider** and **Requester**
271 products of a particular **Interface**. Some conformance requirements of a **Requester** or **Provider**
272 include interactions between the **Requester** and the **Customer** or between the **Provider** and the
273 **Customer**. For example, in order for a **Vendor** to claim conformance to a specification for a
274 **Provider**, a **Customer** **MUST** be able to configure a default natural language for the **Provider** to
275 use.

276 **Feature** - the parts of an **Interface** that a **Provider** supports and that a **Requester** invokes in
277 order to achieve some well-defined behavior. The semantics of the **Feature** is defined in the
278 Interface specification. Examples of **Features** of an (1) API, (2) a file format, or (3) a network
279 protocol **Interface** are: (1) functions in an Application Programming Interface (API), (2) an
280 element in an XML file, and (3) an operation or attribute in a network protocol such as JMF,
281 respectively.

282 **Function** - a **Procedure** programming language construct in which the result consists of an
283 output value which is not passed by reference, and **MAY** also return additional output parameters
284 by reference.

285 **Interface** - a well-defined boundary between a **Requester** and a **Provider** which communicate
286 with each other with one-directional or bi-directional interaction. The specification of the
287 **Interface** is such that a conforming **Requester** and a conforming **Provider** can be implemented
288 independently by different **Vendors** and be able to interoperate. The **Interface** specification
289 defines the syntax and semantics of **Features**. The **Interface** specification also defines which
290 **Features** a **Provider** **MUST**, **SHOULD**, and **MAY** support and which **Features** a **Requester**
291 **MUST**, **SHOULD**, and **MAY** supply. Examples of an **Interface** include a **Application**
292 **Programming Interface (API)** such as a POSIX C library, a file format such as PDF/is, or a
293 **Network Protocol**, such as IPP.

- 294 **Interface Specification** - a specification that defines the syntax and semantics of an **Interface**
295 between a **Requester** and a **Provider**, including the conformance requirements for the
296 **Requester** and **Provider** such that conforming implementations will interoperate.
- 297 **Interoperability** - the condition when a **Requester** is able to invoke **Features** successfully of an
298 **Interface** supported by a **Provider** made by different **Vendors** without requiring prior agreements
299 to be reached between the vendors. An **Interface** specification MAY REQUIRE the **Provider** to
300 query the capabilities of the **Requester** in order to be able to interoperate.
- 301 **Job Creation operation** - operations that create Job objects, specifically: Print-Job, Print-URI,
302 and Create-Job as defined in [RFC2911].
- 303 **Job Attribute** - an **Operation attribute** or a **Job Template attribute** that applies to the Job
304 (object) as a whole. For example, "job-name" is an **Operation attribute** and "job-priority" is a
305 **Job Template attribute**.
- 306 **Network Protocol** (or more simply **Protocol**) - an **Interface** over a distributed medium, typically
307 a network, in which the initiator sends an operation request and the responder returns an
308 operation response.
- 309 **Procedure** - a subroutine programming language construct in which a calling program passes
310 input parameters by value or reference and output parameters by reference to a named entry
311 point which assigns values to the output parameters based on the input parameters and,
312 possibly, some retained state from previous calls and/or the environment.
- 313 **Provider** - the entity that provides service by accepting invocations of **Features** from a
314 **Requester** across the defined **Interface**. Examples of a **Provider** are an IPP Printer, a spooler,
315 or a directory service. Other examples of a **Provider** for (1) an API, (2) a file format, and (3) a
316 network protocol **Interface**, are (1) a function implementation, (2) a PSI XML parser, or (3) a
317 server, respectively.
- 318 **Requester** - an entity that requests service by invoking **Features** of a **Provider** across the
319 defined **Interface**. Examples of a **Requester** are an **IPP Client**, an application, or a print driver
320 that request some kind of action. Other examples of a **Requester** of (1) an API, (2) a file format,
321 and (3) a network protocol **Interface**, are (1) an application program (or calling program), (2) an
322 Emitter, or (3) a **client**, respectively.
- 323 **Requesting User** - a person that directs the actions of a **Requester** to invoke **Features** of a
324 **Provider** across the **Interface**. Some conformance requirements of a **Requester** include
325 interactions between the **Requesting User** and the **Requester**. For example, the **Requester**
326 MUST localize a certain **Feature** according to the natural language of the **Requesting User**
327 when received from the **Provider** across the **Interface**.
- 328 **Support** - a **Provider** implementation **supports** a **Feature**, if the **Provider** implementation
329 responds to a request from a **Requester** for that **Feature** across the **Interface**. Such a response
330 MUST be according to the semantics defined in the Interface Specifications. Note: a **Customer**
331 MAY choose to disable a **Feature** in the **Provider**, so that the **Provider** implementation no longer
332 **supports** the **Feature**, but that is a **Customer** choice, not a **Vendor** choice.
- 333 **Vendor** - a company or organization that produces a product that conforms to the **Interface**
334 defined in an **Interface Specification** as a **Requester** or as a **Provider** of that **Interface**. Some
335 conformance requirements of a **Requester** or **Provider** include interactions between the **Vendor**
336 and the **Customer**. For example, in order for a **Vendor** to claim conformance to this ICS for a
337 **Provider**, a **Vendor** MUST state the **Conformance Levels** and **Conformance Modules**
338 supported by the **Provider** implementation in any claim of conformance to this ICS.

339 3 <body of standard. Many sections at header1 level>

340 *[You MUST specify what Requesters and Providers of your Interface or Interfaces MUST, SHOULD,*
341 *MAY, SHOULD NOT and MUST NOT do.*

342 *[You MUST include examples that explain general statements. Writing out examples gives you an*
343 *opportunity to test your own understanding of the general statements while writing them and helps a*
344 *reader to test his understanding of the general statements while reading them. If you can't generate an*
345 *example from your statements, that is a red flag to you that you need to work on your statements.*

346 *You SHOULD consult the document entitled, "Tips for Good Technical Writing"(see*
347 *ftp://ftp.pwg.org/pub/pwg/general/good-writing-tips-latest.pdf) before you write.]*

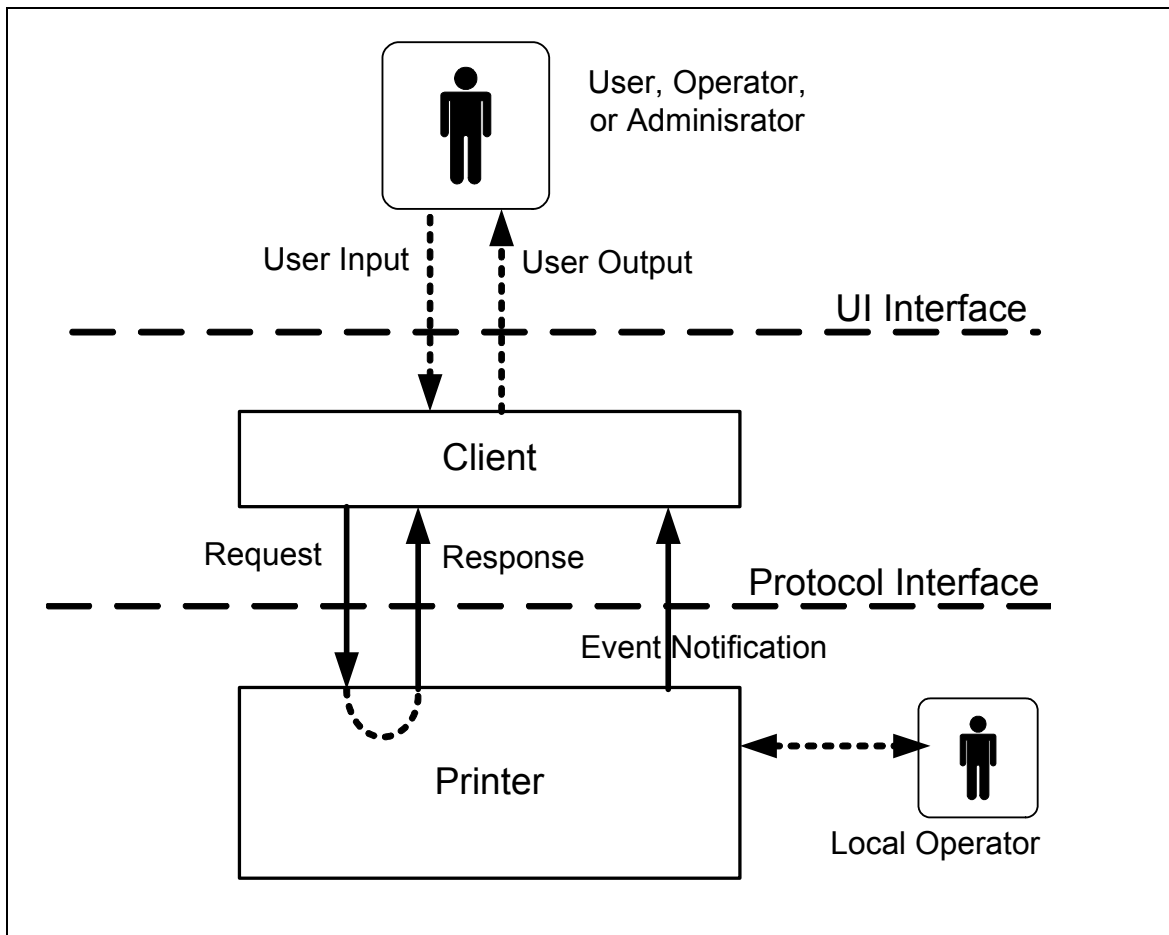
348 *You MUST include some Figures to explain your Interface and its Producers and Consumers. You*
349 *MUST identify each symbol in the Figure. For example, don't assume that an arrow is self-explanatory.*
350 *Don't assume that an arrow is self-evident. A single headed arrow could represent data flow, control*
351 *flow, a request, or a request and response. You MUST include both a Logical Diagram and a*
352 *Configuration Diagram, unless your specification is an extension specification to a base specification*
353 *that already includes a Logical Diagram and a Configuration Diagram:*

354 *A Logical Diagram shows your Interface with the minimum number of Providers and*
355 *Requesters that your specification applies to, typically a single Provider and a single Requester*
356 *across a single Interface (see Figure 1 for an example). Show what flows across your Interface,*
357 *including direction. Indicate all of the Actors in your system, including human users.*

358 *A Configuration Diagram shows a number of Providers and Requesters interacting across one*
359 *or more instances of your Interface(s) and how they interact in a typical deployment (see Figure*
360 *2 and Figure 3 for three IPP examples each). For example, if a Provider can also be a*
361 *Requester and forward requests to a second Provider, one of your Configuration diagrams*
362 *needs to show the two Interfaces with the Provider/Requester in the middle and a Requester*
363 *invoking the first Interface and a Provider supporting the second Interface. You MUST also*
364 *state the slightly different conformance requirements for Non-Leaf, Leaf, and Subordinate*
365 *Providers, if any. For example, a Subordinate Provider MUST be able to accept a "job-id"*
366 *attribute supplied by a trusted Non-Leaf Provider which is acting like a Requester, rather than*
367 *the Subordinate Provider assigning its own "job-id" Job Description attribute values when*
368 *creating jobs. Any claims to conformance to your specification MUST indicate which set(s) of*
369 *conformance requirements are met.*

370 *For IPP specifications, the Requester is called a Client and the Provider is called the IPP Printer or*
371 *Printer for short.*

372 *Figure 1 shows a Logical Diagram for an IPP client and Printer. The downward arrow shows the*
373 *Client sending an IPP Protocol Request to the Printer. The connecting upward arrow shows the*
374 *Printer returning a Response to the Client. And the other vertical arrow shows the Printer*
375 *sending an Event Notification to the Client. The dashed curved line suggests that the Response*
376 *follows immediately after the Request in time. In IPP the UI Interface is not specified very much*
377 *at all, except that the Client SHOULD localize User Output and display GUI information for User*
378 *Input. The dashed bi-directional arrow between the Printer and the Local Operator indicates*
379 *information flow in both directions using an unspecified mechanism.]*



380
381

Figure 1 - Logical Requester - Provider Diagram for IPP

382 Figure 2 and Figure 3 are Configuration Diagrams. The single headed solid arrow labeled JMP
 383 represents a JMF request and response. The arrow labeled “?” represent any sort of
 384 communication which may be uni-directional or bi-directional. The Subordinate Printer (see
 385 Figure 3) can be a Non-Leaf Printer as in Configurations D-F or a Leaf Printer as in
 386 Configurations A-C.
 387

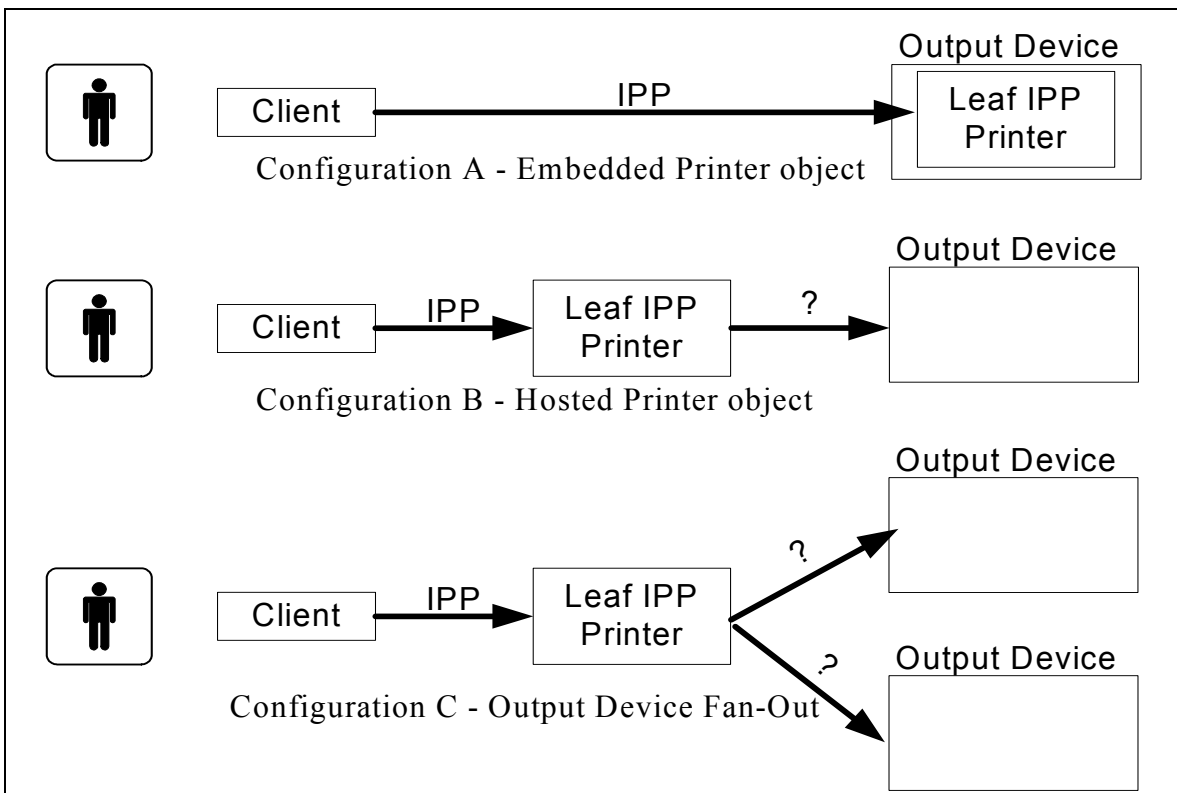
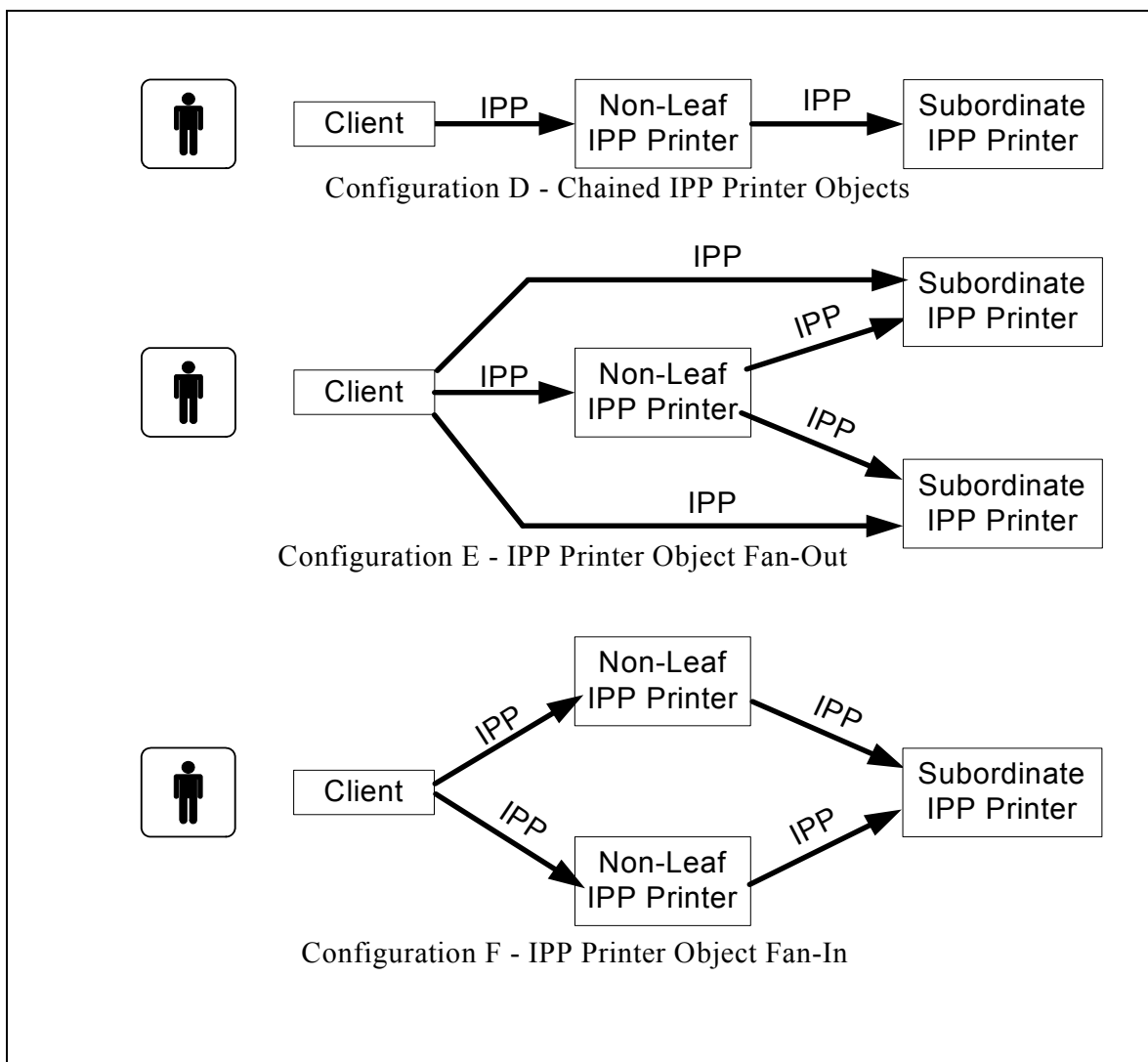


Figure 2 - Leaf IPP Printer objects

388
389

390



391
392
393

Figure 3 - Non-Leaf Printer objects and their Subordinate Printer objects

394 **3.1 Conformance Level(s), Modules, and/or Profiles**

395 *Your specification MUST specify a minimum Conformance Level for the chosen Interface, that is, a*
 396 *minimum number of Features that a Provider implementation MUST support in order to claim*
 397 *conformance to the specification. Your specification MUST also specify a set of Features that the*
 398 *Requester implementation MUST be able to invoke when the Requesting User wishes to invoke the*
 399 *Feature. In other words, if one vendor sells a Requester that conforms to your specification and another*
 400 *vendor sells a Provider for your specification, a Customer buying the two products can expect them to*
 401 *interoperate, at least at the minimum Conformance Level. Furthermore, the interoperability MUST*
 402 *require no prior agreements between the vendors, except to conform to your specification.*

403 *Sometimes writing such an Interoperability specification is not so simple. Sometimes the number of*
 404 *OPTIONAL Features that a Requester or Provider do not need to support is so large that interoperability*
 405 *is affected. Some Customers of a conforming Requester might expect that a conforming Provider would*
 406 *support some of these OPTIONAL Features. Such a specification covering such an Interface SHOULD*
 407 *define additional Conformance Levels, instead of just a minimum Conformance Level, where each Level*
 408 *requires all of the REQUIRED Features of the lower Levels and adds some additional REQUIRED*

409 *Features. Any claim of conformance to an Interface in your specification MUST include which Level or*
410 *Levels are supported. Then for interoperability, the Requester and Provider MUST support the same*
411 *Level in order to give the Customer access to all of the REQUIRED Features at that level. However,*
412 *even if the Levels supported between Requester and Provider are different, there will be interoperability*
413 *at the highest Level that they both support, as well as all lower Levels. In these cases, the prior*
414 *agreement that is necessary is the maximum conformance level that both Requester and Provider*
415 *support of the Interface in question.*

416 *Sometimes the Feature sets do not form subset relationships. For example, whether a Printer supports*
417 *Color Features is fairly independent of whether it also supports Finishing Features. Therefore, it would*
418 *not be possible to have a Color Features Conformance Level and a Finishing Conformance Level above*
419 *or below such a Color Conformance Level. In this case, you can divide the Features into disjoint sets of*
420 *Features called Conformance Modules. For example, one Module could contain Color Features and*
421 *another Finishing Features. Each Module will have REQUIRED Features that a Provider MUST*
422 *support and additional OPTIONAL Features that a Provider MAY support. Any claim of conformance*
423 *to an Interface in your specification MUST include which Module or Modules are supported.*
424 *Requesters and Providers which support the same Modules, will be able to interoperate using the*
425 *Modules that they have in common.*

426 *Sometimes, there are Features that need to belong to multiple Modules. In this case, you SHOULD*
427 *define your Modules with disjoint Feature sets as before. But in addition, define Profiles in which each*
428 *Profile has a list of REQUIRED Modules. Separate Profiles MAY include the same Modules. With*
429 *Profiles, there is no requirement for a super-set, sub-set relationship. Any claim of conformance to an*
430 *Interface defined by your specification MUST include which Profile or Profiles are supported.*
431 *Requesters and Providers which support the same Profiles, will be able to interoperate using the Profiles*
432 *that they have in common.*

433 **4 Conformance Requirements for <fill in part of your title>**

434 *[You MUST include a conformance section, which is a summary list of the important conformance*
435 *requirements for Requesters and Providers of your Interface.*

436 *You SHOULD reference each major section in your specification that contains conformance statements.*
437 *Such a listing is a useful checklist for the implementer of the conformance requirements they need to*
438 *follow without repeating the conformance statements themselves (which could get out of synch).*

439 *If your Interface has a large number of items that Requesters or Providers MAY support, you SHOULD*
440 *attempt to group them into one or more Conformance Levels, Modules, and/or Profiles (see section 3.1).*

441 *The Conformance section MUST make it clear which statements apply to Requesters and which apply to*
442 *Providers and SHOULD list them in two separate sub-sections. Usually, such clarity is easiest by having*
443 *separate sub-sections, one for Requesters and one for Providers. Such separation also simplifies an*
444 *implementer job of going over the list of conformance requirements that apply to the implementation,*
445 *since most implementers are implementing either a Requester or a Provider, but not both.]*

446 This section specifies the conformance requirements for Printers and clients.

447 **4.1 Printer conformance requirements**

448 *[You MUST list the conformance requirements of a Provider of your Interface either as a single*
449 *minimum set or as levels of conformance.]*

450 **In order to conform to this specification, a Printer:**

- 451 1. **MUST** meet the conformance requirements for Printers specified in [RFC2911].
- 452 2. **MUST** support the xxx operations that section nn **REQUIRES** a Printer to support.
- 453 3. **MUST** support the attributes that section nn **REQUIRES** a Printer to support.
- 454 4. etc.

455 4.2 **Client** conformance requirements

456 *[You **MUST** list the conformance requirements for the Requester of your Interface either as a single*
457 *minimum set or as levels of conformance.]*

458 **In order to conform to this specification, a client:**

- 459 1. **MUST** meet the conformance requirements for clients specified in [RFC2911].
- 460 2. **MAY** support any of the xxx operations defined in section nn.
- 461 3. etc.

462 5 References

463 *[You **MUST** list the references that are required in order to conform to your specification.*

464 *You **MUST** include references inside square brackets ([]), so it is clear that you are making a reference*
465 *and that the reader can look up the reference in one of the two References sections. It is acceptable to*
466 *talk about a referenced document using the [] notation directly. For example, “The Create-Job*
467 *operation is defined in [RFC2911] section 3.2.4.”*

468 *Once a standard has been assigned a number, you **MUST** refer to that specification in references using*
469 *some short prefix and the number. The prefix indicates the source of the numbering. For example,*
470 *“rfc” indicates an IETF specification, pwg” indicates an ISTO-IEEE specification, “iso” indicates and*
471 *ISO standard, “ieee” indicates an IEEE standard, etc. This is an example IEEE-ISTO reference:*
472 *[pwg5100.3]. Before a specification is assigned a number, use a short mnemonic reference that the*
473 *knowledgeable reader can surmise. For example, [doc-obj] is a reference to the PWG Working Draft*
474 *“IPP Document Object” specification*

475 *Note: The IETF recommended practice [IETF-TEM] is for two top level sections. However, combining*
476 *them into one with two sub-sections is also acceptable, which we have done here, so that information*
477 *that is common to both sub-sections can be stated just one in the top level section, such as the URL for*
478 *the location of the document repository.]*

479 RFCs are available at: <http://www.rfc-editor.org/rfcsearch.html>. Internet-Drafts are available at:
480 <http://www.ietf.org/ID.html>. IEEE-ISTO PWG standards are located at: <http://www.pwg.org> or
481 more specifically at: <http://www.pwg.org/ipp/index.html>.

482 5.1 Normative References

483 *[You **MUST** list references that contain conformance requirements that also apply to a Requester and/or*
484 *Provider of your Interface whether they be **REQUIRED**, **RECOMMENDED**, or **OPTIONAL**. You*

485 *MUST include the Editors' names, but only their family name (usually last names). Using family names*
486 *only avoids mistaken initials for nick names, improves readability, and avoids putting family names in*
487 *the wrong order in cultures where the family name is normally written first. Normative References*
488 *MUST be at the same level of approval as your specification.]*

489 [RFC2119]
490 Bradner, "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119, March
491 1997

492 [RFC2911]
493 Hastings, Herriot, deBry, Isaacson, and Powell, "Internet Printing Protocol/1.1: Model and
494 Semantics", RFC 2911, September 2000.

495 [IPP-PROD]
496 Ocke, Hastings, IEEE-ISTO Std. 5100.3-2001, IPP Production Printing Attributes – Set
497 1, February 2001. Available at: <ftp://ftp.pwg.org/pub/pwg/standards/pwg5100.3.pdf>, .doc,
498 .rtf

499 5.2 Informative References

500 *[You MUST list additional references that will help the reader understand the subject or a work from*
501 *which the current work was derived, but that does not contain REQUIRED, RECOMMENDED, or*
502 *OPTIONAL conformance requirements for Requester or Providers of the Interface in your specification.*
503 *You MUST include the Editors' names, but only their family name (usually last names). Using family*
504 *names only avoids mistaken initials for nick names, improves readability, and avoids putting family*
505 *names in the wrong order in cultures where the family name is normally written first. Informative*
506 *References MAY be still works in progress (that is, not passed a Last Call), in which case they MUST be*
507 *labeled "work in progress".]*

508 [IANA-CON]
509 Narte and Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs",
510 BCP 26, RFC 2434, October 1998.

511 [RFC1759]
512 Smith, Wright, Hastings, Zilles, and Gyllenskog, "Printer MIB", RFC 1759, March 1995.

513 [RFC2565]
514 Herriot, Butler, Moore, and Turner, "Internet Printing Protocol/1.0: Encoding and
515 Transport", RFC 2565, April 1999.

516 [RFC2566]
517 deBry, Hastings, Herriot, Isaacson, and Powell, "Internet Printing Protocol/1.0: Model and
518 Semantics", RFC 2566, April 1999.

519 [RFC2567]
520 Wright, "Design Goals for an Internet Printing Protocol", RFC 2567, April 1999.

521 [RFC2568]
522 Zilles, "Rationale for the Structure and Model and Protocol for the Internet Printing
523 Protocol", RFC 2568, April 1999.

- 524 [RFC2569]
- 525 Herriot, Hastings, Jacobs, and Martin, "Mapping between LPD and IPP Protocols", RFC
- 526 2569, April 1999.

- 527 [RFC2616]
- 528 Fielding, Gettys, Mogul, Frystyk, Masinter, Leach, and Berners-Lee, "Hypertext Transfer
- 529 Protocol - HTTP/1.1", RFC 2616, June 1999.

- 530 [RFC3196]
- 531 Hastings Manros, Zehler, Kugler, and Holst, "Internet Printing Protocol/1.1: Implementer's
- 532 Guide", RFC3196, November 2001.

- 533 [RFC3380]
- 534 Hastings, Herriot, Kugler, and Lewis, "Internet Printing Protocol (IPP): Job and Printer Set
- 535 Operations", RFC 3380, September 2002.

- 536 [RFC3381]
- 537 Hastings, Lewis, and Bergman, "Internet Printing Protocol (IPP): Job Progress
- 538 Attributes", RFC 3381, September 2002.

- 539 [RFC3382]
- 540 deBry, Hastings, Herriot, Ocke, and Zehler, "Internet Printing Protocol (IPP): The
- 541 'collection' attribute syntax", RFC 3382, September 2002.

542 **6 IANA Considerations**

543 *[Often extensible Interfaces include a means for an implementer to register the extensions with a*
 544 *registration authority who then publishes the extensions in a public place. This section MUST include*
 545 *any registration procedures or indicate that there are none.*

546 *Note: This section 6 is written for IPP standards, you MUST change it for a non-IPP standard, for*
 547 *example, to mention a standard port for a registered URL scheme.*

548 This section contains the registration information for IANA to add to the various IPP Registries
 549 according to the procedures defined in [RFC2911] section 6. The resulting registrations will be
 550 published in the <http://www.iana.org/assignments/ipp-registrations> registry.

551 *Note to the RFC Editor: Replace the [RFCnnnn] References below with the RFC number*
 552 *for the appropriate specification, so that it accurately reflects the content of the information*
 553 *for the IANA Registry.*

554 **6.1 Attribute Registrations**

555 The following table lists all the attributes defined in this specification. These are to be registered
 556 according to the procedures in [RFC2911] section 6.2.

Name	Reference	Section
Operation attributes:		
document-message (text(MAX))	[this spec]	3.8.1.1

```

561
562 Job Template attributes:
563 job-copies (integer(1:MAX)) [this spec] 7.1.1
564 job-cover-back (collection) [this spec] 7.1.2
565 <member attributes are the same as "cover-back">
566 job-cover-front (collection) [this spec] 7.1.3
567 <member attributes are the same as "cover-front">
568 job-finishings (1setOf type2 enum) [this spec] 7.1.4
569 job-finishings-col (1setOf collection) [this spec] 7.1.5
570 <member attributes are the same as "finishings-col">
571
572 Job Description attributes:
573 output-device-assigned (name(127)) [RFCnnnn] 11.8.1
574
575 Printer Description Attributes:
576 document-creation-attributes-supported (1setOf type2 keyword)
577 [this spec] 9.1
578
579 Subscription Template attributes:
580
581 Subscription Description Attributes:
582
583 Document Templates attributes:
584 copies (integer(1:MAX)) [RFC2911] 4.2.5
585

```

586 6.2 Keyword attribute value registrations

587 This section lists all the keyword attribute value registrations defined in this specification. These
588 are to be registered according to the procedures in [RFC2911] section 6.1.

589 Attribute	Value	Reference	Section
590	-----	-----	-----
591	printer-security-supported (1setOf type2 keyword)	[RFC2911]	4.4.3
592	none	[RFC2911]	4.4.3
593	ssl3	[RFC2911]	4.4.3
594	tls	[RFC2911]	4.4.3
595			
596			

597 6.3 Enum Attribute Value Registrations

598 The following table lists all the enum attribute values defined in this specification. These are to be
599 registered according to the procedures in [RFC2911] section 6.1.

600 Value	Name	Reference	Section
601	-----	-----	-----
602	document-state (type1 enum)	[this spec]	8.2.16
603	3 pending	[this spec]	8.2.16
604	5 processing	[this spec]	8.2.16
605	7 canceled	[this spec]	8.2.16
606	8 aborted	[this spec]	8.2.16
607	9 completed	[this spec]	8.2.16
608			
609	operations-supported (1setOf type2 enum)	[RFC2911]	4.4.15
610	0x0033 Cancel-Document	[this spec]	3.8

611	0x0034	Get-Document-Attributes	[this spec]	3.5
612	0x0035	Get-Documents	[this spec]	3.6
613	0x0036	Delete-Document	[this spec]	3.9
614	0x0037	Set-Document-Attributes	[this spec]	3.7
615	0x0039	Create-Document	[this spec]	3.1
616	0x003A	Send-Data	[this spec]	3.2

618 **6.4 Attribute syntax Registrations**

619 The following table list all the attribute syntaxes defined in this specification. These are to be
 620 registered according to the procedures in [RFC2911] section 6.3.

621	Value	Name	Reference	Section
622	-----			
623	collection:		[RFC3382]	3
624	0x34	begCollection	[RFC3382]	7.1
625	0x37	endCollection	[RFC3382]	7.1
626	0x4A	memberAttrName	[RFC3382]	7.1

628 **6.5 Operation Registrations**

629 The following table lists all of the operations defined in this specification. These are to be
 630 registered according to the procedures in [RFC2911] section 6.4.

631	Name	Reference	Section
632	-----		
633	Cancel-Document	[this spec]	3.8
634	Create-Document	[this spec]	3.1
635	Delete-Document	[this spec]	3.9
636	Get-Document-Attributes	[this spec]	3.5
637	Get-Documents	[this spec]	3.6
638	Send-Data	[this spec]	3.2
639	Set-Document-Attributes	[this spec]	3.1

641 **6.6 Attribute Group tag Registrations**

642 The following table lists all the attribute group tags defined in this specification. These are to be
 643 registered according to the procedures in [RFC2911] section 6.5.

644	Value	Name	Reference	Section
645	-----			
646	0x09	document-attributes-tag	[this spec]	14

648 **6.7 Status code Registrations**

649 The following table lists all the status codes defined in this specification. These are to be
 650 registered according to the procedures in [RFC2911] section 6.6.

651	Value	Name	Reference	Section
652	-----			
653	0x0000:0x00FF	- "successful"		
654	none	at this time		

656 0x0100:0x01FF - "informational"
 657 none at this time
 658
 659 0x0300:0x03FF - "redirection" - -- See RFC 2911 Errata
 660 none at this time
 661
 662 0x0400:0x04FF - "client-error"
 663 none at this time
 664
 665 0x0500:0x05FF - "server-error"
 666 0x050A server-error-printer-is-deactivated [RFCnnnn] 5.1
 667
 668

669 **6.8 Registration of Events**

670 The following table lists all the events defined in this specification. These are to be registered
 671 according to the procedures in [RFC2911] section 6.1 as additional type2 keywords to be used
 672 with the "notify-events" Subscription Template attribute (see [ipp-ntfy] section 5.3.3)), i.e., the
 673 "notify-events", "notify-events-default", and "notify-events-supported" attributes.

Attribute	Value	Ref.	Section:
notify-events	(1setOf type2 keyword)		
notify-events-default	(1setOf type2 keyword)		
notify-events-supported	(1setOf type2 keyword)		
<event keyword name>		RFC xxxx	m.n

681 **7 Internationalization Considerations**

682 *[You MUST indicate whether your Interface caters to internationalization, that is, regional character*
 683 *sets and/or natural languages.]*

684 All standardized textual strings must be represented as US-ASCII character codes and local
 685 translations must never be performed. Custom sizes, if limited to local use, may be represented
 686 using any desired character set.

687 **8 Security Considerations**

688 *[You MUST include considerations that an implementer of a Requester or Provider needs to take into*
 689 *account concerning security. Security includes authentication, authorization, and privacy.*
 690 *Authentication is when a Provider is able to verify that a Requester is who he says he is. Authorization*
 691 *is comparing an authenticated identity with a list of allowed Providers. Privacy is ensuring that the*
 692 *exchange has not been modified unbeknownst to the Requester and Provider. For example, include*
 693 *Denial of Service attacks, Man in the Middle attacks, and Trojan Horse scenarios that might occur if the*
 694 *implementer doesn't consider them.]*

695 This specification will have no impact on the security burden of or potential threats to the
 696 importing system.

697 **9 Contributors**

698 *[You MUST include this section, which lists contributors in alphabetical order who deserve significant*
699 *credit for the specification, including their affiliation. Then the list of Editors in section 11 can be just*
700 *the person or people who did the actual writing. Note: existing IPP specifications have listed all the*
701 *people who attended one or more meetings, whether they contributed or not. This template is suggesting*
702 *that the only those who really participated with comments, suggestions, goals, and/or direction in the*
703 *meetings and/or on the mailing list SHOULD be included.]*

704 Ron Bergman, Hitachi Printing Solutions
705 Dennis Carney, IBM Corporation
706 Harry Lewis, IBM
707 Ira McDonald, High North Inc.

708 **10 Acknowledgments**

709 *[You MAY include this section, which contains acknowledgement of Contributors listed in section 9,*
710 *including their affiliation in parenthesis, the Contributors section. Acknowledgements SHOULD*
711 *include an explanation of the contribution of each individual.]*

712 Special thanks for Ron Bergman (Hitachi Printing Solutions) for beginning this template process
713 working with the IEEE-ISTO to produce the IEEE-ISTO 5101.1 Media Name Standard.

714 Dennis Carney debugged earlier versions of the template and used it for his "IPP Actuals"
715 specification.

716 **11 Editor Contact Information**

717 *[You MUST include the list of editors, including address, phone number, and email. Only include*
718 *people who performed significant writing of the specification. Include reviewers and significant*
719 *contributors in the Contributors section (and mention them in the Acknowledgments section, if you like).*
720 *Only the Editors' names appear in the References section for you document. Long lists of Editor's*
721 *names is being discouraged by the IETF and others. For example:]*

722 Tom Hastings
723 Xerox Corporation
724 737 Hawaii St.
725 El Segundo, CA 90245
726
727 Phone: 310 333-6413
728 Fax: 310 333-5514
729 e-mail: hastings@cp10.es.xerox.com

730 **Appendix A <fill-in a title for Annex> (Normative)**

731 *[You MUST indicate for each Appendix whether it is Normative or Informative, by including either*
732 *“(Normative)” or “(Informative)” to the heading line.]*

733 **Appendix B <fill-in a title for Annex> (Informative)**

734 *[You MUST indicate for each Annex whether it is Normative or Informative, by including either*
735 *“(Normative)” or “(Informative)” to the heading line.]*

736 **Appendix C Differences between this Template and the** 737 **IEEE-ISTO standard style**

738 This Appendix lists the differences between this Template and the IEEE-ISTO standard style and
739 explains why. The IEEE-ISTO standard style is represented in the IEEE-ISTO 5101.1 Media
740 Name Standard. These differences have evolved after experience using the IEEE-ISTO standard
741 style for online viewing and printing with Adobe Acrobat. The differences are (working top to
742 bottom):

- 743 1. Merged the redundant second page (which had the Abstract) onto the first page and got
744 rid of the second page. Our understanding is that having two title pages comes from
745 publishing the document in printed form, where the first page is like a cover. Most
746 standards bodies have eliminated the cover so that the Abstract appears on the first
747 page.
- 748 2. Added “(PWG)” after “The Printer Working Group” in the title.
- 749 3. Removed the redundant indications of whether this version is a Working Draft, a
750 Proposed Standard, a Draft Standard, or a Standard from the Header. Only the title
751 indicates the status. A Working Draft has: “Working Draft for a PWG Proposed
752 Standard”. A Proposed Standard after passing WG Last Call will have: “PWG Proposed
753 Standard”. A Draft Standard after passing WG Last Call will have: “PWG Draft
754 Standard”. A Standard after passing PWG Last Call will have: “PWG Standard”
- 755 4. Added the URL for the document on the first page, so that it is easy to find.
- 756 5. Added Editor’s name(s) to the first page, so that proper credit is given as is common in
757 some standards bodies.
- 758 6. Simplified the headers and footers, so that the headers are all the same except for the
759 first page. Keeping the headers and footers simple will reduce errors in producing drafts
760 and allow editor’s to focus on content more and format less.
- 761 7. The Left side of the Header is blank for Working Drafts, then gets IEEE-ISTO 51nn.n
762 after the Working Draft passes WG Last Call. Thus the editor only changes the Header
763 once, when the specification passes Last Call.
- 764 8. In the Header, the Title is centered with most of the fixed parts removed.

- 765 9. In the Header, the page number is always on the right. The IEEE-ISTO format did not
766 have the page number on the top at all, which has proved a problem when viewing the
767 specification with Adobe Acrobat, when the user can only see part of the page at one
768 time. Also scrolling upwards and downwards needs the page numbers to be at the top
769 and bottom.
- 770 10. The Footer follows the IEEE-ISTO style, except the page number is always centered,
771 instead of alternating left and right. This makes it easier to spot the page number when
772 viewing it on the screen.
- 773 11. Page numbers in the Header and Footer, start at 1 with the first page. No Roman
774 numerals are used. This makes it easier to relate pages as numbered by Adobe Acrobat
775 and those printing on each page and in the Table of Contents. Also when printing out
776 selected pages, the page number used by the Printer driver agree with the printed page
777 numbers.
- 778 12. Defined the Body Text style to use ragged right, rather than justified, in order to make the
779 document more readable by people when printed or displayed though it may not look as
780 pretty.
- 781 13. Added editor names for PWG standards in the References section, as is practice in some
782 standards bodies and most technical publications.
- 783 14. Removed the first initials from references and used only family names as is becoming
784 practice to avoid the cultural confusion about whether family names come first or last..

785 **Appendix D Change Log (Informative)**

786 *[You MUST maintain a Change Log as the last ANNEX that summarizes the changes from one minor*
787 *version to the next.*

788 *You SHOULD put the changes in forward chronological order so that is easiest to find the most recent*
789 *changes by turning to the very end of the specification. Also chronological order is more natural if a*
790 *reader wants to read all of the history.*

791 *You SHOULD remove the Change Log when the specification is published as an integer version.]*

792 The following changes have been made to versions of this specification, in forward chronological
793 order:

794 **D.1 Changes to make version 0.1, November 14, 2002**

795 The following changes were made to create version 0.1, November 14, 2002:

796 Initial version.

797 **D.2 Changes to make version 0.2, January 27, 2003**

798 The following changes were made to create version 0.2, January 27, 2003 after careful review by
799 Dennis Carney and Ira McDonald:

- 800 1. Generalized the template so that it can be for any PWG standard, not just IPP.
- 801 2. Simplified the template so that as few fields as possible have to be updated as the
802 specification progresses through the process - otherwise everyone does it differently.
- 803 3. Follow the IEEE-ISTO style with the exceptions listed in Appendix C derived from
804 experience viewing and printing on-line versions using Adobe Acrobat.
- 805 4. Explained the file naming scheme for PWG WGs.
- 806 5. Added the requirements for a Logical Diagram and a Configuration Diagrams and
807 showed some IPP examples.
- 808 6. Highlighted IPP specific instructions and examples in green, like this, in order to give real
809 examples, but make the Template work for all PWG Working Groups
- 810 7. Added RFC 3380, 3381, 3382.
- 811 8. Added a number of terms that may be useful to PWG projects other than IPP.
- 812 9. Moved the Appendix "Tips for Good Technical Writing" to a separate document available
813 at: <ftp://ftp.pwg.org/pub/pwg/general/templates/good-writing-tips-latest.pdf>