The Printer Working Group

October 2, 2013
Working Draft

IPP Implementor's Guide v2.0

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

Status: Interim

Abstract: This document updates and extends RFC 3196 for all IPP protocol versions..

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0.

This document is a PWG Working Draft. For a definition of a "PWG Working Draft", see: ftp://ftp.pwg.org/pub/pwg/general/pwg-process30.pdf

This document is available electronically at:

ftp://ftp.pwg.org/pub/pwg/ipp/wd/wd-ippig20-20131002.pdf

Smith Kennedy 10/2/13 11:34 PM
Deleted: 20130722

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

Title: *IPP Implementor's Guide v2.0*

Smith Kennedy 10/2/13 11:34 PM
Deleted: Title: *IPP/2.0 Implementer's Guide*

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

About the IEEE-ISTO

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

Smith Kennedy 10/2/13 11:34 PM
Deleted: http://www.ieee.org/

Smith Kennedy 10/2/13 11:34 PM
Deleted: http://standards.ieee.org/)

For additional information regarding the IEEE-ISTO and its industry programs visit:

http://www.ieee-isto.org

Smith Kennedy 10/2/13 11:34 PM
Deleted: http://www.ieee-isto.org

About the IEEE-ISTO PWG

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

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

For additional information regarding the Printer Working Group visit:

http://www.pwg.org

Contact information:

The Printer Working Group
c/o The IEEE Industry Standards and Technology Organization
445 Hoes Lane
Piscataway, NJ 08854
USA

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

About the Internet Printing Protocol Work Group

The Internet Printing Protocol (IPP) working group has developed a modern, full-featured network printing protocol, which is now the industry standard. IPP allows a print client to query a printer for its supported capabilities, features, and parameters to allow the selection of an appropriate printer for each print job. IPP also provides job information prior to, during, and at the end of job processing.

For additional information regarding IPP visit:

http://www.pwg.org/ipp/

Implementers of this specification are encouraged to join the IPP mailing list in order to participate in any discussions of the specification. Suggested additions, changes, or clarification to this specification, should be sent to the IPP mailing list for consideration.

Table of Contents

1. Introduction 7
2. Terminology 7
2.1 Conformance Terminology 7
2.2 Other Terminology 7
2.3 Acronyms and Organizations 7
3. Requirements 8
3.1 Rationale 8
3.2 Use Cases 8
3.2.1 Developer Implementing IPP Client Support 8
3.2.2 Developer Implementing IPP Support in a New Printer 8
3.3 Out of Scope 8
3.4 Design Requirements 8
4. Tasks and Implementation Alternatives 9
4.1 Create A Relationship With A Printer 9
4.1.1 Discover And Select A Printer Via A Discovery Protocol 10
4.1.2 Select A Printer Via Static DNS Hostname, IPv6 Address Or JPv4 Address 10
4.2 Validate User Access to Printer 12
4.3 Get Printer Information and Print Job Options 12
4.4 Check constraints between presented options 13
4.5 Submitting a Print Job 14
4.5.1 Submitting a print job with document data 14
4.5.2 Submitting a print job with document references 16
4.6 Monitoring print job status 19
4.7 Canceling a Print Job 20
4.8 Getting printer supplies status 21
5. Using Attributes with IPP Operations 22
5.1 General Principles 22
5.2 Explicit "document-format" Selection 22
5.3 Prefer "media-col" Attribute To "media" Attribute 22
5.4 Specifying Finishings With "finishings-col", "finishings" and Others 23
5.5 Controlling Intended Output Using "ipp-attribute-fidelity", "job-mandatory-attributes", and "pdl-override-supported" 23
5.6 Prefer "multiple-document-handling" For Copies Collation 24
5.7 Prefer Supplying Explicit "orientation-requested" Attribute 25
5.8 Evaluating Printer Capability Attributes 25
5.9 Interpreting Object Status Attributes 25
6. IPP Server Best Practices 26
6.1 Resource URIs 26
7. HTTP Protocol Use 26
7.1 HTTP/1.1 Expect Header 26
7.2 "Host" Header Field 27
7.3 If-Modified-Since, Last-Modified, and 304 Not Modified 27
7.4 Cache-Control 27
8. Security Considerations 28
8.1 Client Security Considerations 28

Deleted: IPP/2.0 Implementer's Guide
Smith Kennedy 10/2/13 11:34 PM
Deleted: 7
Smith Kennedy 10/2/13 11:34 PM
Deleted: 7
Smith Kennedy 10/2/13 11:34 PM
Deleted: 7
Smith Kennedy 10/2/13 11:34 PM
Deleted: 7
Smith Kennedy 10/2/13 11:34 PM
Deleted: 7
Smith Kennedy 10/2/13 11:34 PM
Deleted: 8
Smith Kennedy 10/2/13 11:34 PM
Deleted: 8
Smith Kennedy 10/2/13 11:34 PM
Deleted: 8
Smith Kennedy 10/2/13 11:34 PM
Deleted: 8
Smith Kennedy 10/2/13 11:34 PM
Deleted: 8
Smith Kennedy 10/2/13 11:34 PM
Deleted: 8
Smith Kennedy 10/2/13 11:34 PM
Deleted: 8
Smith Kennedy 10/2/13 11:34 PM
Deleted: 8
Smith Kennedy 10/2/13 11:34 PM
Deleted: 4. Client Operations Best ... [1]
Smith Kennedy 10/2/13 11:34 PM
Deleted: 9
Smith Kennedy 10/2/13 11:34 PM
Deleted: 10
Smith Kennedy 10/2/13 11:34 PM
Deleted: IPv6…Pv4 Address 10 ... [2]
Smith Kennedy 10/2/13 11:34 PM
Deleted: 12
Smith Kennedy 10/2/13 11:34 PM
Deleted: 12
Smith Kennedy 10/2/13 11:34 PM
Deleted: 13
Smith Kennedy 10/2/13 11:34 PM
Deleted: 14
Smith Kennedy 10/2/13 11:34 PM
Deleted: 14
Smith Kennedy 10/2/13 11:34 PM
Deleted: 16
Unknown
Field Code Changed ... [3]
Smith Kennedy 10/2/13 11:34 PM
Deleted: 18
Smith Kennedy 10/2/13 11:34 PM
Deleted: 20
Unknown
Field Code Changed ... [4]
Smith Kennedy 10/2/13 11:34 PM
Deleted: 20
Unknown
Field Code Changed ... [5]
Smith Kennedy 10/2/13 11:34 PM
Deleted: 21
Smith Kennedy 10/2/13 11:34 PM
Deleted: 21
Unknown
Field Code Changed ... [6]
Unknown
Field Code Changed ... [7]
Smith Kennedy 10/2/13 11:34 PM
Unknown
Field Code Changed ... [8]
Smith Kennedy 10/2/13 11:34 PM
... [9]
Unknown
Field Code Changed ... [10]
Smith Kennedy 10/2/13 11:34 PM

8.1.1 Using the "ipps:" URI Scheme 28
8.1.2 Document Encryption 28
8.2 Server Security Considerations 28
8.2.1 Support For The "ipps" URI Scheme 28
8.2.2 Document Encryption 28
9. Important Implementation Options 28
9.1 Using "xxx-actual" attribute 28
9.2 Using "preferred-attributes" attribute 28
10. Internationalization Considerations 28
10.1 Client Considerations 28
10.2 Server Considerations 28
11. References 29
11.1 Informative References 29
12. Authors' Addresses 30
13. Change History 30
13.1 February 5, 2013 30
13.2 March 20, 2013 30
13.3 May 13, 2013 31
13.4 July 19, 2013 31
13.5 September 22, 2013 31
13.6 October 2, 2013 32

Deleted: IPP/2.0 Implementer's Guide
Unknown Field Code Changed ... [23]
Smith Kennedy 10/2/13 11:34 PM Deleted: 25
Unknown Field Code Changed ... [24]
Smith Kennedy 10/2/13 11:34 PM Deleted: 25
Unknown Field Code Changed ... [25]
Smith Kennedy 10/2/13 11:34 PM Deleted: 25
Unknown Field Code Changed ... [26]
Smith Kennedy 10/2/13 11:34 PM Deleted: 25
Unknown Field Code Changed ... [27]
Smith Kennedy 10/2/13 11:34 PM Deleted: 25
Unknown Field Code Changed ... [28]
Smith Kennedy 10/2/13 11:34 PM Deleted: 26
Unknown Field Code Changed ... [29]
Smith Kennedy 10/2/13 11:34 PM Deleted: 26
Unknown Field Code Changed ... [30]
Smith Kennedy 10/2/13 11:34 PM Deleted: 26
Unknown Field Code Changed ... [31]
Smith Kennedy 10/2/13 11:34 PM Deleted: 26
Unknown Field Code Changed ... [32]
Smith Kennedy 10/2/13 11:34 PM Deleted: 26
Unknown Field Code Changed ... [33]
Smith Kennedy 10/2/13 11:34 PM Deleted: 26
Unknown Field Code Changed ... [34]
Smith Kennedy 10/2/13 11:34 PM Deleted: 26
Unknown Field Code Changed ... [35]
Smith Kennedy 10/2/13 11:34 PM Deleted: 26
Unknown Field Code Changed ... [36]
Smith Kennedy 10/2/13 11:34 PM Deleted: 27
Unknown Field Code Changed ... [37]
Smith Kennedy 10/2/13 11:34 PM Deleted: 27
Unknown Field Code Changed ... [38]
Smith Kennedy 10/2/13 11:34 PM
Unknown Field Code Changed ... [39]
Smith Kennedy 10/2/13 11:34 PM
Unknown Field Code Changed ... [40]
Smith Kennedy 10/2/13 11:34 PM
Unknown Field Code Changed ... [41]

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

1. Introduction

The use case descriptions below represent stages or sub-tasks that users perform in the process of using a printer. Each of these below include a textual description as well as a series of workflow options for how it might be implemented using IPP. Each workflow option will be informally labeled according to its perceived quality, using the set of labels {"BAD", "POOR", "GOOD", "BETTER", "BEST"}, that are ordered from least desirable to most desirable.

2. Terminology

2.1 Conformance Terminology

Capitalized terms, such as MUST, MUST NOT, RECOMMENDED, REQUIRED, SHOULD, SHOULD NOT, MAY, and OPTIONAL, have special meaning relating to conformance as defined in Key words for use in RFCs to Indicate Requirement Levels [RFC2119]. The term CONDITIONALLY REQUIRED is additionally defined for a conformance requirement that applies to a particular capability or feature.

2.2 Other Terminology

Capitalized Term In Italics: definition of the term with any references as appropriate.

2.3 Acronyms and Organizations

IANA: Internet Assigned Numbers Authority, http://www.iana.org/

IETF: Internet Engineering Task Force, http://www.ietf.org/

ISO: International Organization for Standardization, http://www.iso.org/

PWG: Printer Working Group, http://www.pwg.org/

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

3. Requirements

3.1 Rationale

The Internet Printing Protocol/1.1: Implementor's Guide [RFC3196] was ratified in November 2001. Since that time many extensions to IPP have been ratified, and the scope of use of IPP has grown considerably. Given all these extensions to IPP, implementers would benefit from an updated best practices document that covers the use of these extensions, as well as the core of IPP that has remained unchanged, to assist implementers in their efforts to deliver a quality client experience.

3.2 Use Cases

3.2.1 Developer Implementing IPP Client Support

Garrett is a developer working on a new client platform that is adding system-level printing support. Many printers support IPP Everywhere [PWG5100.14], so he plans to implement IPP Everywhere printing support in his client platform. But IPP Everywhere and its related standards don't describe how to best use IPP for the various tasks his software must perform, in order to deliver a quality client user experience. He finds RFC 3196, but finds its recommendations are insufficient. Using the IPP Use Best Practices document, he is able to avoid some common design pitfalls and quickly deliver a quality IPP client experience.

3.2.2 Developer Implementing IPP Support in a New Printer

Duncan is a firmware developer at a printer vendor creating a new printer, and that printer includes support for IPP Everywhere. In reading the IPP Implementor's Guide v2.0, he can more accurately anticipate how some segment of clients implemented according to these practices are likely to behave, and more rapidly understand how the various operations can be used with one another to achieve certain tasks.

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementors Guide,

3.3 Out of Scope

The following are considered out of scope for this specification:

1. Specifications to extend or replace portions of the Internet Printing Protocol itself
2. Normative requirements regarding user experience

3.4 Design Requirements

The design requirements for this specification are:

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

1. Explore tasks performed by client implementations
2. Enumerate a series of alternatives
3. Rank those options according to a non-numeric qualitative grading scheme

Smith Kennedy 10/2/13 11:34 PM
Deleted: <#>Client Operations Best Practices

4. Tasks and Implementation Alternatives

For a number of tasks, the set of IPP operations provides a rich enough set of semantics that it is possible to perform those tasks in a few different ways. In this section a number of common tasks will be enumerated, and some alternatives for how those tasks might be performed using IPP operations will be discussed.

Each of the descriptions below will in most cases list IPP operations to be performed by a client, in the order that they are listed. This sequence of steps can be considered a form of pseudo-code. Some tasks may require iteration over a set of steps, or parallel execution of 2 or more steps. For the description to be complete yet concise, this document defines several pseudo-code operators. The "loop()" operator will be used to indicate iterative execution. The "forall()" operator will be used to indicate parallel execution. The operations to be performed within the context of each of these will be listed within the parentheses following the pseudo-code statement. For example, " loop(IPP Get-Job-Attributes; IPP Get-Printer-Attributes)" would be used to describe repeatedly performing an IPP Get-Job-Attributes operation followed by an IPP Get-Printer-Attributes operation.

4.1 Create A Relationship With A Printer

You can't print to a printer if you cannot establish a connection to it. Historically, connecting to a printer to establish a "relationship" with it meant identifying a printer and then creating a persistent local records and resources for that printer relationship with your system's print spooler. This was called a "print queue", and it involved binding drivers to create the relationships needed to communicate at the different levels, and then keeping record of that set of relationships so that it could be re-used at a later time. The set of printers or other devices the user's system might encounter was relatively small and fairly static.

More recent re-thinking of this relationship between client and printer has resulted in more "dynamic" relationship creation, where universal drivers can interrogate a device hosting a print service using a standardized protocol solution stack, and using that dynamically ascertain and update print service attributes. In this paradigm, a "persistent" print service record is more like a Web browser bookmark.

Both paradigms still require a method of identifying the target devices. That can be done using dynamic service discovery protocols where the services respond to discovery requests, or explicitly by name (host name or raw IPv4/IPv6 address).

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

4.1.1 Discover And Select A Printer Via A Discovery Protocol

Discovery protocols are used to identify instances of print services or printers by searching the network for service types or device types. This helps the user by making it so that they don't need to do a physical survey of devices' addresses.

Regardless of the actual discovery protocol used, the APIs driving the protocols generally can be used in either a synchronous or asynchronous fashion. Unfortunately, many legacy software systems (as well as developers) are accustomed to the synchronous model, which is easily identified by the presence of a "refresh button". The synchronous model is not as user friendly as the asynchronous model, but it is somewhat easier to write programs in a synchronous way than an asynchronous way.

Options

- POOR:
  - Perform network discovery with a synchronous API
    - Show progress bar
    - Discovery.Start()
    - sleep(X) where X is a few seconds
    - Discovery.Stop()
  - Present the results of the discovery process
  - "Refresh" button restarts the process
    - *List contents can be stale*
    - *Results are not "live"*
    - *Depends on user action / interaction for fresh results by clicking on the "Refresh" button*
  - User selects a printer and presses "Continue" or equivalent
- BETTER:
  - Same as "POOR" above but user interface self-refreshes every few seconds in a worker thread that refreshes the UI owned by the main thread (or equivalent).
- BEST:
  - Perform network discovery with an asynchronous API
    - Show List UI widget
    - Discovery.Start() with a callback
    - Callback is called when discovery responses (add or remove) are received
    - UI list updated as per discovery response (row added or removed)
  - User selects a printer and presses "Continue" or equivalent
    - Discovery.Stop()

4.1.2 Select A Printer Via Static DNS Hostname, IPv6 Address Or IPv4 Address

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPv6

In some cases, discovery protocols are not available, not adequate or unnecessary, and the user will resort to acquiring the printer's name or address and attempt to contact it

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

directly rather than looking it up via a service discovery protocol. Examples of this include: situations where the target printer does not reside within the scope that the discovery protocol services; or when a URI or bare host name or address are provided directly to the user.

For IPP this can be a source of problems because the IPP Printer Object, which is specified by the resource path, may not be known or discoverable if the user only has a simple DNS host name, IPv6 address or IPv4 address. If they have been given a URI, then the resource path hopefully is included in that URI so that it specifies the Printer Object. But that may not always be the case, and historically URIs have not been as commonly provided to users as simple DNS host names or IPv4 addresses.

Options

- BAD:
  - Let each printer model make up its own default resource path, and depend on some other protocol to get the resource path
    - *IPP has no defined standard mechanism to enumerate the Printer objects' resource paths*
- POOR:
  - IPP Get-Printer-Attributes with printer-uri set to a URI that was manually entered by the user
    - *The "ipp" URI scheme could be used to encode the hostname and the resource path*
    - *Having the user enter the URI exposes too many details to the user, including the detail about the fact that IPP is actually being used. Users need not be aware of which print protocol is being used.*
- GOOD:
  - IPP Get-Printer-Attributes with printer-uri set to a well-known Printer resource path ("/ipp/print")
    - *Fixed well-known path eliminates ambiguity to default Printer Object*
    - *Does also allow indirect discovery of other Printer Objects*
    - *Not universally implemented*
- BETTER:
  - IPP Get-Printer-Attributes with printer-uri set to "/"
  - Examine the "printer-uri-supported" attribute; use the first URI in the list
  - IPP Get-Printer-Attributes with printer-uri set to first URI
    - Allows variance and different Printer Objects to be found
    - Not widely implemented
- BEST:
  - IPP Get-Services
    - *Coming with System Control Service*

Smith Kennedy 9/23/13 9:40 AM
Comment [1]: Is this really going to be better? Yes, expected to have metadata associated with each URI specifying the class of service.

How to express this "comment" inline with text? Perhaps as a footnote.

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

4.2 Validate User Access to Printer

Selecting a printer is misleading to the user if the user isn't allowed to use the selected printer. Therefore, access restrictions should be validated before selection confirmation (queue creation, etc.) is done on the client system.

Options

- BAD:
  - Do Nothing
    - *The user may choose a printer but not be able to use it due to not having access credentials (username or password or whatever) to use that printer*
- GOOD:
  - Only validate reachability: ICMP echo, SNMP GET, HTTP HEAD on "/"
    - *Validates reachability*
    - *No validation that IPP service is available on target host*
    - *No validation that the user has access rights to the IPP Printer Object*
- BETTER:
  - IPP Get-Printer-Attributes
    - Validates reachability
    - Validates IPP service is available on target host
    - *No validation that the user has access rights to the IPP Printer Object*
- BEST:
  - IPP Validate-Job
    - *Send the defaults, but provide the credentials to allow the user access to be determined*
    - *Validates reachability*
    - *Validates IPP service is available on target host*
    - *Validates user has access rights to the IPP Printer Object*
  - <<maintain session for later operations or make note of identity used in a successful IPP Validate-Job request / response>>
    - *Later operations will have inaccurate client authentication identity information or potentially a poor user experience*

Smith Kennedy 10/2/13 10:57 PM
Comment [2]: Not sure if this should go somewhere else

4.3 Get Printer Information and Print Job Options

Once the user has selected a printer, it is necessary for the print system to understand the capabilities that the printer device's print service provides. This includes what print job payload formats can be consumed by the print service, the available options and default choices, and so forth. It also includes other information about the device itself, such as its location. Some of this is done at relationship creation time (queue creation time), perhaps by consulting information stored statically in the printer. It may be that this information can all be retrieved from the printer itself. This is basically the print dialog's activity between the time that the user performs an action to request that the print dialog be presented, and

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

the time that the dialog is presented to the user, populated with the available option choices.

Options

- BAD:
  - Depend on a-priori knowledge about a particular model as a way of listing options for the model of device identified as the target.
    - *All "legacy style" model specific print drivers are an example of this alternative.*
- GOOD:
  - IPP Get-Printer-Attributes
    - *Request includes no printer attributes; only operation attributes*
    - *Response will contain the job template attributes for all document formats*
    - *Client guesses at what attributes may work or not work for a given document format, or uses a-priori knowledge*
- BEST:
  - IPP Get-Printer-Attributes
  - Process results
    - *Decide on a document format*
    - *Choose other filterable attributes, if desired*
  - IPP Get-Printer-Attributes
    - *Request includes all attributes reported by the get-attributes-supported attribute, with desired values chosen*
      - *document-format: value specifying the desired document format*
      - *'job-template' group value*
      - *Any other attributes that the client specifically wishes to support or detect.*
      - *others?*
    - *Response will contain the job template attributes appropriately filtered ("colored") for the attributes provided in the request, such as attributes that are required for a given document format (if any exist) or changes in supported attribute values. For a detailed discussion of key attributes an IPP Client should watch for and process in the response, see §5.8.*

Smith Kennedy 10/2/13 10:57 PM
Comment [3]: What others? Cross reference between this and 5.8

4.4 Check constraints between presented options

Printer features and options are presented typically in a print dialog. Some of these have states that have relationships with other options' states, where one cannot be in a particular state if another one is too. These are known as constraints, and they must be calculated any time the state of a control changes state. There are various ways that this can be done.

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

Options

- BAD:
  - Do nothing or use local model-specific data (i.e. PPDs)
- POOR:
  - IPP Validate-Job
    - *Every time a control is changed, the client sends IPP Validate-Job with attribute values corresponding to current state of controls*
- GOOD:
  - IPP Validate-Job
    - *When "Print" button is pressed, confirms the job creation / submission will succeed (authentication, etc.)*
    - *Client depends on this operation to perform constraints validation printer-side*
- BETTER:
  - IPP Get-Printer-Attributes
    - Printer Object implements job-constraints-supported
    - Printer Object implements job-resolvers-supported
  - Process constraints on client
- BEST:
  - IPP Get-Printer-Attributes
    - Printer Object implements job-constraints-supported
    - Printer Object implements job-resolvers-supported
  - Process constraints on client
  - IPP Validate-Job
    - *When "Print" button is pressed*
    - *Confirms the job creation / submission will succeed (authentication, etc.)*
    - *Constraints validation handled client-side*

4.5 Submitting a Print Job

Once the user has decided on options, the print job is generated and ultimately made available to the printer in some fashion. There are several different ways that this may occur.

4.5.1 Submitting a print job with document data

This is the classical way that a print job is sent from the client to the print service: first a job is created, and then the job information and payload content are sent from the client to the print service.

Options

- BAD:
  - IPP Print-Job

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

- *No pre-flight checks*
- *The printer may reject it but only after it has been transmitted.*
  - *IPP would never respond early; HTTP could respond early but that would be effectively a transport level exception outside the scope of IPP.*
- *Better to check ticket and content types first.*

- GOOD:
  - IPP Validate-Job
    - *As recommended by Best Practice 4.4*
    - *Pre-flight operation to check the job by validating the job attributes and document type, authentication and transport encryption upgrades (if needed)*
    - *Check for the "preferred-attributes" attribute in the reply to detect whether attribute value substitution will occur*
  - IPP Print-Job
    - *Creates the job and sends the payload in one operation*
    - *However, the Job object's URI isn't usually known until the job transmission is complete*
    - *Doesn't work well with flow-controlled (low-end) printers*
- BETTER:
  - IPP Create-Job
    - *Returns immediately with the job URI for monitoring and ticket processing status*
    - *If there is a problem then Create-Job may fail the same as Validate-Job would, but may not, which is why we do a Validate-Job first (so that there isn't a zombie job there)*
    - *Once the job is created, the Client MAY receive a list of the xxx-actual job processing attributes from the IPP Printer, as per [PWG5100.8].*
      - *The xxx-actual job attributes could be used to detect substitutions that would be used by the Printer Object. Observing this, the client may decide to cancel the job rather than submit the document with this job. If the original job was cancelled, the client could create another job with a new set of attributes submitted, or error out and not submit a job at all.*
      - *HOWEVER, xxx-actual attributes are not stable until the job reaches a terminating state (aborted, cancelled or completed) and processors of the job receipt should keep this in mind. Given this, it isn't clear whether there is any value in examining the xxx-actual attributes since the values may not be set until after it is too late.*
    - *Allows an opportunity to perform a Cancel-Job operation during document submission*
  - IPP Send-Document
    - *Payload transmission is de-coupled from the creation of the job*
    - *Multiple documents can be sent to build up a compound job*

Smith Kennedy 9/23/13 9:40 AM
Comment [4]: xxx-actual attributes are not stable until the job reaches a terminating state (aborted, cancelled or completed) and processors of the job receipt should keep this in mind. Update with bracketed reference to PWG 5100.8 (which needs to be added to references at the bottom). Add a new section about important implementation options

Smith Kennedy 10/2/13 10:57 PM
Comment [5]: What to recommend? Add a new section about important implementation options?

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

- *Client MUST check to see if value of "multiple-document-jobs-supported" is "true", to see if it is OK to do multiple Send-Document operations to the same Job object.*

- BEST:
  - IPP Validate-Job
    - *As recommended by Best Practice 4.4*
    - *Pre-flight operation to check the job by validating the job attributes and document type, authentication and transport encryption upgrades (if needed)*
    - *Check for the "preferred-attributes" attribute in the reply to detect whether attribute value substitution will occur*
  - IPP Create-Job
    - *Returns immediately with the job URI for monitoring and ticket processing status*
    - *If there is a problem then Create-Job may fail the same as Validate-Job would, but may not, which is why we do a Validate-Job first (so that there isn't a zombie job there)*
    - *Once the job is created, the Client MAY receive a list of the xxx-actual job processing attributes from the IPP Printer, as per [PWG5100.8].*
      - *The xxx-actual job attributes could be used to detect substitutions that would be used by the Printer Object. Observing this, the client may decide to cancel the job rather than submit the document with this job. If the original job was cancelled, the client could create another job with a new set of attributes submitted, or error out and not submit a job at all.*
      - *HOWEVER, xxx-actual attributes are not stable until the job reaches a terminating state (aborted, cancelled or completed) and processors of the job receipt should keep this in mind. Given this, it isn't clear whether there is any value in examining the xxx-actual attributes since the values may not be set until after it is too late.*
    - *Allows an opportunity to perform a Cancel-Job operation during document submission*
  - IPP Send-Document
    - *Payload transmission is de-coupled from the creation of the job*
    - *Multiple documents can be sent to build up a compound job*
    - *Client MUST check to see if value of "multiple-document-jobs-supported" is "true", to see if it is OK to do multiple Send-Document operations to the same Job object.*

Smith Kennedy 9/23/13 9:40 AM
Comment [6]: xxx-actual attributes are not stable until the job reaches a terminating state (aborted, cancelled or completed) and processors of the job receipt should keep this in mind. Update with bracketed reference to PWG 5100.8 (which needs to be added to references at the bottom). Add a new section about important implementation options

Smith Kennedy 10/2/13 10:57 PM
Comment [7]: What to recommend? Add a new section about important implementation options?

Smith Kennedy 10/2/13 10:57 PM
Comment [8]: Are there IPP printer / server implementation details that must be implemented in a certain way to ensure that things work best?

Smith Kennedy 9/23/13 9:40 AM
Comment [9]: Are there IPP printer / server implementation details that must be implemented in a certain way to ensure that things work best.

4.5.2 Submitting a print job with document references

This is a slightly different way that a print job is sent from the client to the print service: a job is created and made available for retrieval by the print service, and when the print job the job information and job payload content are sent by the client to the print service.

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

Options

- POOR:
  - IPP Print-URI
    - *No pre-flight checks*
    - *Printer may reject it but only after it has been transmitted*
    - *Better to check ticket and content types first*
- GOOD:
  - IPP Validate-Job
    - *As recommended by Best Practice 4.4*
    - *Pre-flight operation to check the job by validating the job attributes and document type, authentication and transport encryption upgrades (if needed)*
    - *Check for the "preferred-attributes" attribute in the reply to detect whether attribute value substitution will occur*
  - IPP Print-URI
    - *Creates the job and sends a URL to where the payload can be retrieved in one operation*
    - *Printer Object "pulls" the document file rather than being given it by the client*
    - *However, the Job object's URI isn't usually known until the job transmission is complete*
    - *Printer may respond with client-error-document-access-error status code, or might add document-access-error to job-state-reasons*
    - *URI may not be accessible at time of processing*
- BETTER:
  - IPP Create-Job
    - *Returns immediately with the job URI for monitoring and ticket processing status*
    - *If there is a problem then Create-Job may fail the same as Validate-Job would, but may not, which is why we do a Validate-Job first (so that there isn't a zombie job there)*
    - *Once the job is created, the Client MAY receive a list of the xxx-actual job processing attributes from the IPP Printer, as per [PWG5100.8].*
      - *The xxx-actual job attributes could be used to detect substitutions that would be used by the Printer Object. Observing this, the client may decide to cancel the job rather than submit the document with this job. If the original job was cancelled, the client could create another job with a new set of attributes submitted, or error out and not submit a job at all.*
      - *HOWEVER, xxx-actual attributes are not stable until the job reaches a terminating state (aborted, cancelled or completed) and processors of the job receipt should keep this in mind. Given this, it isn't clear whether there is any value in examining the xxx-actual attributes since the values may not be set until after it is too late.*

Smith Kennedy 9/23/13 9:40 AM
Comment [10]: xxx-actual attributes are not stable until the job reaches a terminating state (aborted, cancelled or completed) and processors of the job receipt should keep this in mind. Update with bracketed reference to PWG 5100.8 (which needs to be added to references at the bottom). Add a new section about important implementation options

Smith Kennedy 10/2/13 10:57 PM
Comment [11]: What to recommend? Add a new section about important implementation options?

- *Allows an opportunity to perform a Cancel-Job operation during document submission*
- IPP Send-URI
  - *Payload URI transmission is de-coupled from the creation of the job*
  - *Printer may respond with client-error-document-access-error status code, or might add document-access-error to job-state-reasons*
  - *URI may not be accessible at time of processing*

- BEST:
  - IPP Validate-Job
    - *As recommended by Best Practice 4.4*
    - *Pre-flight operation to check the job by validating the job attributes and document type, authentication and transport encryption upgrades (if needed)*
    - *Check for the "preferred-attributes" attribute in the reply to detect whether attribute value substitution will occur*
  - IPP Create-Job
    - *Returns immediately with the job URI for monitoring and ticket processing status*
    - *If there is a problem then Create-Job may fail the same as Validate-Job would, but may not, which is why we do a Validate-Job first (so that there isn't a zombie job there)*
    - *Once the job is created, the Client MAY receive a list of the xxx-actual job processing attributes from the IPP Printer, as per [PWG5100.8].*
      - *The xxx-actual job attributes could be used to detect substitutions that would be used by the Printer Object. Observing this, the client may decide to cancel the job rather than submit the document with this job. If the original job was cancelled, the client could create another job with a new set of attributes submitted, or error out and not submit a job at all.*
      - *HOWEVER, xxx-actual attributes are not stable until the job reaches a terminating state (aborted, cancelled or completed) and processors of the job receipt should keep this in mind. Given this, it isn't clear whether there is any value in examining the xxx-actual attributes since the values may not be set until after it is too late.*
    - *Allows an opportunity to perform a Cancel-Job operation during document submission*
  - IPP Send-URI
    - *Payload URI transmission is de-coupled from the creation of the job*
    - *Printer may respond with client-error-document-access-error status code, or might add document-access-error to job-state-reasons*
    - *URI may not be accessible at time of processing*

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

Smith Kennedy 10/2/13 10:57 PM
Comment [12]: How to handle this appropriately? What recommendations should be provided?

Smith Kennedy 10/2/13 11:34 PM
Deleted: *<#>(How to handle this appropriately? What recommendations should be provided?)*

Smith Kennedy 9/23/13 9:40 AM
Comment [13]: xxx-actual attributes are not stable until the job reaches a terminating state (aborted, cancelled or completed) and processors of the job receipt should keep this in mind. Update with bracketed reference to PWG 5100.8 (which needs to be added to references at the bottom). Add a new section about important implementation options

Smith Kennedy 10/2/13 10:57 PM
Comment [14]: What to recommend? Add a new section about important implementation options?

Smith Kennedy 10/2/13 10:57 PM
Comment [15]: How to handle this appropriately? What recommendations should be provided?

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

Smith Kennedy 10/2/13 11:34 PM
Deleted: *<#>(How to handle this appropriately? What recommendations should be provided?)*

4.6 Monitoring print job status

While the print job is being processed, users may wish to know whether it is proceeding successfully, or whether there are conditions that they need to handle that are preventing processing from proceeding, such as a media jam, open covers, marking agents depleted, and so forth.

For those options below that involve polling the Printer Object, the degree to which the option is better or worse is due in no small part to the polling frequency. The interval should be tuned so that the frequency of queries is not so great that it burdens the Printer Object or Job Object or the network, but not so small that there is an undesirable lag between when an event occurs and when the user is notified. It is always a bad practice in any case if a client is polling as fast as the network can handle traffic. When polling the Printer Object, the client should not poll as fast as the network can handle traffic but should rather tune its polling interval to achieve a suitable balance between user responsiveness and resource overuse.

Options

- BAD:
  - loop(IPP Get-Jobs; IPP Get-Printer-Attributes; sleep(interval))
    - *Polling all status information needlessly uses a large number of resources*
    - *Polling for status without the actual job ID is imprecise*
    - *Polling is generally not desirable because the imprecision of Get-Jobs coupled with tight polling will unnecessarily burden the Printer Object. See above regarding polling intervals*
- POOR:
  - IPP Get-Jobs
    - *Get the job identifier in question*
  - loop(IPP Get-Job-Attributes; sleep(interval))
    - *Monitor the job with ID acquired earlier*
    - *See above regarding polling intervals*
- GOOD:
  - loop(IPP Get-Job-Attributes / IPP Get-Printer-Attributes)
    - *Monitor the value of printer-state attribute as well as targeted monitoring of a specific job's status*
    - *Only ask for the attributes you need, to minimize the bandwidth needed for the information being used.*
- BETTER:
  - IPP Create-Printer-Subscriptions
  - IPP Get-Notifications
  - <<some amount of time passes - could be short or long>>
  - Notification Received
  - IPP Get-Job-Attributes

Smith Kennedy 9/23/13 9:40 AM
Comment [16]: Update the notation - the '/' is supposed to be there for concurrent operations but many of these are sequential and may not have happened near one another in time

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

- *Asynchronous / long running queries for notifications that don't require polling*
- *When you see that a job has completed, query the state of that job at that time*
- *Printer state changes will be provided by subscribing to the printer; subscribing to the job will provide less information and not be as useful*

- BEST:
  - IPP Create-Printer-Subscriptions
    - *Subscription created at queue creation time*
  - << Submit a print job using methods in section 4.5 >>
    - *Most optimal if IPP Create-Job / IPP Send-Document is used so that the job URI is positively returned by the Printer Object; IPP Get-Jobs could be used as well*
  - IPP Create-Job-Subscriptions
  - IPP Get-Notifications
  - <<some amount of time passes - could be short or long>>
  - Notification Received
    - *Asynchronous / long running queries for notifications that don't require polling*
    - *Follow-up operations may not be needed because the desired information is in the job notification already.*
    - *When you see that a job has completed, query the state of that job at that time*
    - *Printer state changes will be provided by subscribing to the printer; subscribing to the job will provide less information and not be as useful*

4.7 Canceling a Print Job

It may be that the user wants to terminate a job before it has been fully processed, for whatever reason. There are things that must be done to ensure that the client has decisively cleaned up the state of the Job Object if the client is responsible for canceling the job. Clients' leaving broken Job objects on the Print service is bad behavior.

There is also a dependency between the options below and how the job was submitted.

Options

- BAD:
  - IPP Print-Job with incomplete transmission (client stops sending chunks)
    - *Perceived by IPP Printer as an I/O / connection problem*
- GOOD:
  - IPP Print-Job
  - [something causes the client to initiate cancelling the job submission]

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

- Client sends a zero length chunk
- IPP Cancel-Job operation via a separate TCP connection
  - *Request for the job via a second connection, which for some printers could result in a PDL interpreter hang because the last chunk sent didn't stop on a "statement" boundary*

- BEST:
  - IPP Create-Job
  - IPP Send-Document
  - Client sends a zero length chunk
  - IPP Cancel-Job via a separate TCP connection

4.8 Getting printer supplies status

Some administrative tasks, like checking consumables levels, are presented to end users in some cases, such as during print job status or in print dialogs. This is useful to end-users and should be supported. Ideally only IPP extensions based on open standards (i.e. PWG standard) that all devices implement would be needed.

Options

- BAD:
  - Do nothing
    - *Subpar user experience*
- POOR:
  - Use some proprietary protocol or (possibly closed) platform-specific extension to IPP
    - *This falls short of the ideal that all devices implement a publicly specified and open extension to IPP.*
    - *Private extensions to IPP are less desirable than a publicly specified and open protocol other than IPP.*
- GOOD:
  - SNMP and standard printer MIBs
    - *Public standard but not IPP, so falls short of the ideal that all devices implement a publicly specified extension to IPP*
- BETTER:
  - IPP Get-Printer-Attributes
    - *Printer must implement JPS3 "printer-supply" attribute*
- BEST:
  - IPP Create-Printer-Subscription operation
  - IPP Get-Notifications operation
  - <<some amount of time passes - could be short or long>>
  - IPP Get-Printer-Attributes operation when notification is received
    - *Printer must implement JPS3 "printer-supply" attribute*

Smith Kennedy 10/2/13 10:57 PM
Comment [17]: Is it possible to create a subscription that will notify not just on any change but on whether the value is within or outside some value / value range / threshold? Would this even be useful?

Smith Kennedy 9/23/13 9:40 AM
Comment [18]: another place where the notation needs to be fixed - what is the '+' trying to communicate?

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

5. Using Attributes with IPP Operations

Some attributes that IPP has labeled as optional should always be used as a best practice. Below are some of these attributes and how they should be used in various contexts.

5.1 General Principles

It is important for an IPP Client to bear in mind that there are risks with depending on printer attribute default values. The xxx-default values may not be bound to the job at submission time; binding may occur at print time. If this occurs, and the xxx-default values have changed, then unexpected results may be produced that differ from the user's expectations.

Smith Kennedy 10/2/13 11:34 PM
Deleted: ,

The xxx-actual attributes reported by the IPP Printer in its operation response will report the values that are bound to the job.

An IPP Printer should bind default values to the job's attributes at job creation time, and to report these back to the IPP Client using the xxx-actual response attributes.

5.2 Explicit "document-format" Selection

While IPP Printer Objects provide a default document format (which is known via the document-format-default attribute), as a general principle, it is much better for an IPP Client to explicitly provide the document-format attribute with all operations relating to validating or submitting a document payload to the printer (Validate-Job, Print-Job, Send-Document).

An IPP Client should not send "application/octet-stream" as the value for the "document-format" attribute in an IPP request.

Smith Kennedy 10/2/13 10:57 PM
Comment [19]: What should be said about how an IPP Server should be using and supporting "application/octet-stream" as a document-format?

An IPP Server should return the "document-format-actual" attribute in the IPP response to report the detected MIME media type. If this response is in response to a request that does include a payload, the "document-format-actual" attribute value should be the detected format of the payload, not simply returning to the IPP Client the value of the "document-format" attribute provided in the request.

5.3 Prefer "media-col" Attribute To "media" Attribute

Given a Printer Object that supports both "media" and "media-col" attributes, an IPP Client should prefer to include the "media-col" attribute with operations that accept one of these attributes. This is true for when "media" and "media-col" are top-level attributes as well as when "media" or "media-col" may be included within other collection attributes, such as "job-sheets", "job-error-sheet", "job-accounting-sheets", and others.

The "media-col" attribute provides finer and more reliable control over media selection. As one example, the "media-col.media-size" sub-attribute is defined using integer value types rather than floating point value types, which avoids floating point calculation problems when converting between PWG size names and dimensions.

The IPP Server should allow for a limited degree of inaccuracy even with "media-col.media-size". If the dimensions are off by a small amount (1-3 units) this should not be regarded by the IPP Server as a mismatch.

5.4 Specifying Finishings With "finishings-col", "finishings" and Others

There are different contexts where an IPP Client should use "finishings" as opposed to "finishings-col". This is due to...what?

When mapping from JDF or other rich ticket formats, the "finishings-col" should be used.

If an IPP Printer supports IPP Finishings 2.0, then...

An IPP Printer SHOULD support IPP Finishings 2.0 (PWG 5100.XX).

5.5 Controlling Intended Output Using "ipp-attribute-fidelity", "job-mandatory-attributes", and "pdl-override-supported"

There are several attributes that may be used by a client to control whether a print job will print as requested or rather will print with "best effort" but will not comply with all requested job attributes. These attributes are listed below, each with a synopsis of their meaning and purpose.

ipp-attribute-fidelity - (RFC 2911) - Attribute provided by IPP Client in job submission operations. If absent the IPP Printer will assume the value is "false". If present and value is "true" then the IPP Printer must support all job attributes and attribute values included in the job submission operation or else the IPP Printer MUST reject the operation.

job-mandatory-attributes - (PWG 5100.7) - Identifies which Job Template attributes the Printer MUST support in this Job Creation request in order to accept the Job. This allows the IPP Client to specify at a finer level of granularity those attributes that must be supported. If job-mandatory-attributes is included and it lists all the job attributes, this is equivalent to including the ipp-attribute-fidelity attribute with value set to "true".

pdl-override-supported - (RFC 2911) - Specifies whether the printer is capable of overriding job attributes embedded in the job document(s) with IPP job attributes, and with what level of reliability: "attempted" (RFC 2911), "not-attempted" (RFC 2911), "guaranteed" (PWG 5100.11).

Given these descriptions, an IPP Client SHOULD comply with the following guidelines:

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

Smith Kennedy 10/2/13 10:57 PM
Comment [20]: Should this say IPP Printer or IPP Server or does this apply equally to both?

Smith Kennedy 10/2/13 11:34 PM
Deleted: Prefer

Smith Kennedy 10/2/13 11:34 PM
Deleted: " Attribute To "finishings" Attribute .
Relationship Between

Smith Kennedy 10/2/13 11:34 PM
Deleted: "finishings-col" attributes, a client should prefer to include the "finishings-col" attribute with operations that accept one of these attributes.

Smith Kennedy 10/2/13 11:34 PM
Deleted: Relationship Between

Smith Kennedy 10/2/13 10:57 PM
Comment [21]: TBD

Smith Kennedy 10/2/13 10:57 PM
Comment [22]: TBD

Smith Kennedy 10/2/13 10:57 PM
Comment [23]: Should 'job-creation-attributes-supported' be mentioned here? It doesn't seem to apply to this particular sub-section

Smith Kennedy 10/2/13 11:34 PM
Deleted: <<Needs to be discussed on the mailing list>> . ... [42]

- Do not always specify 'ipp-attribute-fidelity' = 'true' because this can have unintended side effects, such as output scaling, etc.
- If available prefer 'job-mandatory-attributes' over 'ipp-attribute-fidelity'
- Regard 'pdl-override-supported' = 'attempted' as functionally equivalent to 'pdl-override-supported' = 'not-attempted', because there is no guarantee. In either of these cases, the IPP Client SHOULD provide document data that is the correct size, etc. to avoid overrides and not depend on the IPP Printer to perform these overrides unless it can guarantee it.
- Strive to implement the best option in the following list
    - GOOD: Use ipp-attribute-fidelity=true only when the user wants specific output intent
    - BETTER: Discover whether the IPP Printer supports job-mandatory-attributes, and use it if supported, listing attributes whose values were chosen by overt user interaction (explicit choices in the print dialog, etc.), and no others

An IPP Printer SHOULD comply with the following guidelines regarding how to support 'ipp-attribute-fidelity', 'job-mandatory-attributes', and 'pdl-override-supported':

- The IPP Printer SHOULD support job-mandatory-attributes
- The IPP Printer SHOULD support 'guaranteed' for at least JPEG and PWG Raster document formats.

5.6 Prefer "multiple-document-handling" For Copies Collation

An IPP Client wishing to specify that multiple copies of a Job should be collated should avoid using "sheet-collate" and instead use "multiple-document-handling" with a value set to "separate-documents-collated-copies".

The "sheet-collate" attribute, as defined in [RFC 3381], "specifies whether or not the media sheets of each copy of each printed document in a job are to be in sequence, when multiple copies of the document are specified by the 'copies' attribute". The key point here is that the IPP Client likely doesn't want to collate sheets, but rather collate copies of the Job. For more details on the relationship between these 3 IPP concepts and their relationship with one another, please see [PWG 5100.13] §10.

(It is interesting that the definition of "multiple-document-handling" in RFC 2911 includes the statement that this attribute "is relevant only if a job consists of two or more documents".)

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

Smith Kennedy 10/2/13 10:57 PM
Comment [24]: Method to do this is TBD - either ipp-features-supported keyword, or defining a job-mandatory-attributes-supported Printer Description attribute

Smith Kennedy 10/2/13 11:34 PM
Deleted: Attribute to

Smith Kennedy 9/23/13 9:40 AM
Comment [25]: This was called "job-collate" in the minutes, but that attribute doesn't seem to exist. Assuming what was intended was the "sheet-collate" attribute from RFC 3381

Smith Kennedy 10/2/13 11:34 PM
Deleted: " Attribute ... [43]

Smith Kennedy 10/2/13 11:34 PM
Deleted: over

Smith Kennedy 10/2/13 11:34 PM
Deleted: ".

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

5.7 Prefer Supplying Explicit "orientation-requested" Attribute

An IPP Client should provide the "orientation-requested" attribute with each document submission to an IPP Job, with a value that correctly corresponds to the orientation of the pages in the submitted document itself.

5.8 Evaluating Printer Capability Attributes

Before formulating and submitting a job, an IPP Client SHOULD query the IPP Printer to evaluate its capabilities. Techniques for doing this using different sequences of operations are described in §4.3. For these operations, certain attributes should be included so that the operation responses can return as much relevant information as possible.

ipp-features-supported

TBD

xxx-supported

TBD

job-creation-attributes-supported

An IPP Client SHOULD include the 'job-creation-attributes-supported' attribute in its Get-Printer-Attributes queries, are attributes that an IPP Client should include when evaluating the IPP Printer's capabilities.

printer-settable-attributes-supported

TBD

operations-supported

TBD

job-constraints-supported / job-resolvers-supported

TBD

5.9 Interpreting Object Status Attributes

An IPP Client...

Smith Kennedy 10/2/13 10:57 PM
Comment [26]: Fill this out

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

6. IPP Server Best Practices

This section enumerates practices that should be followed by implementers of IPP servers, whether embedded in a Printer or in a separate Print Server.

6.1 Resource URIs

URIs used to direct an IPP client to various resources should be reachable even if all other network services other than IPP have been disabled in the Printer or Print Server.

As an example, the "printer-icons" URI should still be reachable even if the printer has been configured to disable its embedded web server. Since that attribute is currently defined to be a set of URIs, and those URIs must use the "http:" or "https:" schemes, the URI should include the port number of the IPP service in the URI, like so: "http://myprinter.mydomain.com:631/icon.png".

7. HTTP Protocol Use

IPP currently uses HTTP/1.1 for its transport. IPP/2.0 and other IPP specifications have specified some of the facilities of HTTP that IPP clients and servers should support in order to provide the semantics that IPP needs to provide a great user experience. Even so, there are best practices that should be followed.

7.1 HTTP/1.1 Expect Header

As defined in [RFC 2616 "HTTP/1.1"], the "Expect" header allows the client to check with the server on the HTTP connection negotiation before sending the HTTP request payload.

The IPP client should implement the following:

- On first request to a printer, include the "Expect: 100-continue" header.
- Wait up to 1 second for a response.
- If no response is received, remember this for the next request so that you don't have the 1-second delay; continue sending the request.
- If an HTTP 100 Continue status code is returned, continue sending the request
- If an HTTP 301 Moved Permanently or 302 Moved Temporarily status code is returned, redirect the request to the new URI *or* fail/report an error depending on the security requirements of the Client (redirection is generally unexpected)

Smith Kennedy 10/2/13 11:34 PM
Deleted: a

Smith Kennedy 10/2/13 11:34 PM
Deleted: (continue)

Smith Kennedy 10/2/13 11:34 PM
Deleted: a

Smith Kennedy 10/2/13 11:34 PM
Deleted: (moved permanently)

Smith Kennedy 10/2/13 11:34 PM
Deleted: (moved temporarily)

- If an HTTP 400 Bad Request status code is returned, remember this (don't use Expect header) and re-send the POST request. This Printer is technically non-conforming since it fails RFC 2616 requirements for a HTTP/1.1 server.
- If an HTTP 401 status code is returned, re-send the POST request with the requested credentials.
- If an HTTP 403 status code is returned, the client should fail/report an error.
- If a 417 Expectation Failed, the client should resend with the Expect header
- If a 426 status code is returned, the client should send an OPTIONS * request to upgrade to TLS, and then re-send the POST request.

The IPP server should implement the following:

- Return status code 403 for unauthorized client addresses when the HTTP level authentication or authorization is not adequate
- Return status code 200 with an IPP response containing the client-error-not-authorized status code when the IPP level authentication or authorization is not adequate
- Status codes 301 and 302 are not recommended
- Return status code 400 only if problems are detected with the HTTP request itself
- Return status code 200 with an IPP response containing the client-error-bad-request status code if problems are detected with the IPP operation

7.2 "Host" Header Field

Printers MUST validate the Host request header and SHOULD use the Host value in generated URIs.

7.3 If-Modified-Since, Last-Modified, and 304 Not Modified

Printers MUST support the If-Modified-Since request header (section 14.25 [RFC2616]), the corresponding response status ("304 Not Modified", section 10.3.5 [RFC2616]), and the Last-Modified response header (section 14.29 [RFC2616]).

The If-Modified-Since request header allows a Client to efficiently determine whether a particular resource file (icon, ICC profile, localization file, etc.) has been updated since the last time the Client requested it.

7.4 Cache-Control

Printers and Clients MUST conform to the caching semantics defined in section 13 [RFC2616]. Typically, most resource files provided by a Printer in a GET response will be

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

Smith Kennedy 10/2/13 11:34 PM
Deleted: a

Smith Kennedy 10/2/13 11:34 PM
Deleted: (

Smith Kennedy 10/2/13 11:34 PM
Deleted:)

Smith Kennedy 10/2/13 11:34 PM
Deleted: a

Smith Kennedy 10/2/13 11:34 PM
Deleted: a

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

cacheable but IPP responses in a POST response are not. Therefore, Printers MAY provide a Cache-Control header in GET responses with an appropriate "max-age" value and MUST provide a Cache-Control header in IPP POST responses with the value "no-cache".

8. Security Considerations

8.1 Client Security Considerations

8.1.1 Using the "ipps:" URI Scheme

TBD

8.1.2 Document Encryption

TBD

8.2 Server Security Considerations

8.2.1 Support For The "ipps" URI Scheme

TBD

8.2.2 Document Encryption

TBD

9. Important Implementation Options

9.1 Using "xxx-actual" attribute

9.2 Using "preferred-attributes" attribute

10. Internationalization Considerations

10.1 Client Considerations

An IPP Client should use localization catalogs provided by the IPP Printer, if available.

10.2 Server Considerations

An IPP Printer should provide localization catalogs to IPP Clients.

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

11. References

11.1 Informative References

[PWG5100.8] D. Carney, H. Lewis, "Internet Printing Protocol: '-actual' Attributes", PWG 5100.8-2003, March 2003, ftp://ftp.pwg.org/pub/pwg/candidates/cs-ippactuals10-20030313-5100.8.pdf

[PWG5100.12] R. Bergman, H. Lewis, I. McDonald, M. Sweet, "IPP/2.0 Second Edition", PWG 5100.12-2011, February 2011, ftp://ftp.pwg.org/pub/pwg/candidates/cs-ipp20-20110214-5100.12.pdf

[PWG5100.13] M. Sweet, I. McDonald, P. Zehler, " IPP: Job and Printer Extensions – Set 3 (JPS3)", 5100.13-2012, July 2012, ftp://ftp.pwg.org/pub/pwg/candidates/cs-ippjobprinterext3v10-20120727-5100.13.pdf

[PWG5100.14] M. Sweet, I. McDonald, A. Mitchell, J. Hutchings, "IPP Everywhere", 5100.14-2013, January 2013, ftp://ftp.pwg.org/pub/pwg/candidates/cs-ippeve10-20130128-5100.14.pdf

[RFC2616] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, " Hypertext Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999, http://www.ietf.org/rfc/rfc2616.txt

[RFC2910] R. Herriot, S. Butler, P. Moore, R. Tuner, J. Wenn, "Internet Printing Protocol/1.1: Encoding and Transport", RFC 2910, September 2000, http://www.ietf.org/rfc/rfc2910.txt

[RFC2911] R. deBry, T. Hastings, R. Herriot, S. Isaacson, P. Powell, "Internet Printing Protocol/1.1: Model and Semantics", RFC 2911, September 2000, http://www.ietf.org/rfc/rfc2911.txt

[RFC3196] T. Hastings, C. Manros, P. Zehler, C. Kugler, H. Holst, "Internet Printing Protocol/1.1: Implementer's Guide", RFC 3196, November 2001, http://www.ietf.org/rfc/rfc3196.txt

[RFC3381] T. Hastings, H. Lewis, R. Bergman, "Internet Printing Protocol (IPP): Job Progress Attributes", RFC 3381, September 2002, http://www.ietf.org/rfc/rfc3381.txt

[RFC6762] S. Cheshire, M. Krochmal, "Multicast DNS", RFC 6762, February 2013, http://www.ietf.org/rfc/rfc6762.txt

[RFC6763] S. Cheshire, M. Krochmal, "DNS-Based Service Discovery", RFC 6763, February 2013, http://www.ietf.org/rfc/rfc6763.txt

Smith Kennedy 10/2/13 11:34 PM
Deleted: ftp://ftp.pwg.org/pub/pwg/candidates/cs-ippactuals10-20030313-5100.8.pdf

Smith Kennedy 10/2/13 11:34 PM
Deleted: ftp://ftp.pwg.org/pub/pwg/candidates/cs-ipp20-20110214-5100.12.pdf

Smith Kennedy 10/2/13 11:34 PM
Deleted: ftp://ftp.pwg.org/pub/pwg/candidates/cs-ippjobprinterext3v10-20120727-5100.13.pdf

Smith Kennedy 10/2/13 11:34 PM
Deleted: ftp://ftp.pwg.org/pub/pwg/candidates/cs-ippeve10-20130128-5100.14.pdf

Smith Kennedy 10/2/13 11:34 PM
Deleted: http://www.ietf.org/rfc/rfc2616.txt

Smith Kennedy 10/2/13 11:34 PM
Deleted: http://www.ietf.org/rfc/rfc2910.txt

Smith Kennedy 10/2/13 11:34 PM
Deleted: http://www.ietf.org/rfc/rfc2911.txt

Smith Kennedy 10/2/13 11:34 PM
Deleted: http://www.ietf.org/rfc/rfc3196.txt

Smith Kennedy 10/2/13 11:34 PM
Deleted: http://www.ietf.org/rfc/rfc6762.txt

Smith Kennedy 10/2/13 11:34 PM
Deleted: http://www.ietf.org/rfc/rfc6763.txt

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

Smith Kennedy 10/2/13 11:34 PM
Deleted: http://docs.oasis-open.org/ws-dd/discovery/1.1/os/wsdd-discovery-1.1-spec-os.html

[WSDiscovery-1.1] OASIS, "OASIS Web Services Dynamic Discovery (WS-Discovery) Version 1.1", July 2009, http://docs.oasis-open.org/ws-dd/discovery/1.1/os/wsdd-discovery-1.1-spec-os.html

12. Authors' Addresses

Primary authors:

Smith Kennedy
Hewlett-Packard Co.
11311 Chinden Blvd. MS 506
Boise, ID 83714
smith.kennedy@hp.com

The authors would also like to thank the following individuals for their contributions to this white paper:

Evan Williams - Kentucky State Board of Recreation

13. Change History

13.1 February 5, 2013

Initial revision.

13.2 March 20, 2013

Resolved issues from feedback provided during the IPP conference call on February 25, 2013, as documented in teleconference meeting minutes and author's own notes.

1. Added Validate-Job operation as operation to be used during printer selection process to validate access by client / user
2. Replaced previous Section 5 "Conformance Requirements" with new Section 5 "Attributes and Their Use in Operations"
3. Replaced previous Section 6 "Internationalization Considerations" with new Section 6 "HTTP Protocol Usage"
4. Added updated list of references

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

13.3 May 13, 2013

Resolved remaining issues from February 25, 2013 teleconference meeting notes and other feedback received since then.

1. Added PWG 5100.8, 5100.13 and WS-Discovery references; removed boilerplate [REFERENCE] reference entry; fixed references for 5100.14.

2. Updated with meeting notes fixes from April 8, including updated notation for comments (now in italic), revised notation for indicating iterative looping and parallel execution, and many other changes to section 4. Also changed document naming convention and moved URL where it is to be located on PWG FTP site.

13.4 July 19, 2013

Updated with feedback and discussions from the May 2013 F2F in Cupertino, and other piecemeal feedback.

1. Changed title to "IPP/2.0 Implementer's Guide

2. Changed Abstract to "Updates and extends RFC 3196 for IPP/2.0."

3. Renamed to follow the wd-ippig20-yyyymmdd.docx/pdf naming convention given the title change.

4. Made a PWG Working Draft, not a best practice

5. Added normative reference to RFC 3196.

13.5 September 22, 2013

Updated with feedback from reflector discussions and conference calls. This is a "new baseline", and much more work is needed to fill in TBD sections.

1. Substantially elaborated section 5.5.

2. Updated comments about using xxx-actual in detecting whether the printer uses the requested / recommended attribute values (more discussion needed)

3. Editorial changes (fixed "Implementors" to be "Implementer's", renamed some section titles

Smith Kennedy 10/2/13 11:34 PM
Deleted: IPP/2.0 Implementer's Guide

13.6 October 2, 2013

1. Changed the title to "IPP Implementor's Guide v2.0" (in part reverting change #3 from 2013-09-23) and the abstract to make it clear this applies to all IPP versions, not just IPP/2.0.

2. Expanded §5.8 to enumerate the attributes used in evaluating printer capabilities and how to properly use them or handle the cases where they are absent and assumed to be not supported. This is mostly a set of placeholders at this point.

3. Updated §5.6 to clarify guidance on "multiple-document-handling" vs. "sheet-collate".

4. Updated §4.2 to include a placeholder statement to make note that an IPP Client should maintain a sense of user identity between 4.2 and 4.3 to ensure that the capabilities evaluated are really those that the user has access to use and more importantly that none are missing.

5. Comments in section 4.5 to catch all the places where using "xxx-actual" to recognize substitutions before the job has been processed is suggested, because this seems to be a problematic suggestion.

