The Internet Printing Protocol (IPP)

Carl-Uno Manros

Xerox Corporation, Corporate Research & Technology
701 S. Aviation Blvd. Mailstop: ESAE-231
El Segundo, California, USA
manros@cp10.es.xerox.com

RÉSUMÉ.

ABSTRACT. This article gives an overview of the Internet Printing Protocol (IPP), currently under development in the Internet Engineering Task Force. The intention is to get this new specification accepted as a Proposed Internet Standard, on which implementations can be based to insure future interoperability between print clients and print servers/printers throughout the Internet environment. The text describes some of the history, some of the more important requirements, and the main technical solutions provided by the protocol.

KEY WORDS : IPP, printing, print clients, print servers, Internet protocol, fidelity, URI, HTTP, security, internationalization, DPA, IETF, SSL3, TLS

1. Introduction

The Internet Engineering Task Force (IETF) has defined not only the basic transfer protocols that make the Internet a reality, but has also defined a number of application level protocols, including protocols for e-mail, file transfer, and the World Wide Web. However, the area of printing over the Internet has so far not been addressed. This was the most obvious reason for initiating the Internet Printing Protocol project. Another major reason was that several vendors had started to work on their own proprietary solutions for web based printing, which would have meant competing de-facto solutions, which would probably not have been able to interoperate.

This article gives you some historic background on earlier printing standards, and some more details about the requirements that the Internet Printing Protocol is trying to meet. It also describes some of the development processes and choices that the project has had to make in order to come up with solutions that meets most of the requirements without being too complex. This article also contains all the the main capabilities and restrictions of the protocol, to the extent that they can be considered stable at present. These include the scope limitations that were agreed to early in the project, which limited the functionality to end user operations, as well as the discussion about mandatory versus optional features of the protocol.

This article also discusses a number of other areas that concerns the relationship between IPP and other existing protocols in the Internet environment. This includes the use of transfer protocols, security protocols, object encoding mechanisms, internationalization aspects etc.

Last, but not least, is some discussion about various options for implementing the IPP in products.

2. History

The history of printing standards has two separate starting points:

1. The European Computer Manufacturers Association (ECMA) published their standard ECMA-140 with the title "Document Printing Application" (DPA) in March 1990 [ECMA].
2. The Internet Engineering Task Force (IETF) published a Request for Comments document RFC 1179 with the title "Line Printer Daemon Protocol" (LPD) in August 1990 [LPD].

Let us follow how the history developed from this point on. The ECMA standard was used as the basis for an implementation at MIT in Cambridge, USA as part of the Palladium project. This is the only known implementation of the ECMA standard. More important is that the ECMA work served as basis for

follow-on work in ISO/IEC, which published the first two parts of the "Document Printing Application" (DPA) as standard ISO/IEC 10175 in 1996 [ISO 10175]. DPA was based on the OSI stack of protocols, which by the time the standard was published was not considered worthwhile any more. However, a handful of independent implementations were made by major vendors, using the model and semantics in part 1 of ISO/IEC 10175, but each basing it on some other more popular proprietary protocol stack, which meant that there was little or no interoperability between implementations. As a parallel activity to the ISO/IEC standard development, the Institute of Electrical and Electronics Engineers (IEEE) published an interface specification called "POSIX System Administration - Part 4: Printing Interfaces" in 1994 [POSIX]. This also included administrative features, which were fed back to the ISO/IEC group working on part 3 of the ISO/IEC 10175 standard.

Let us see what happened to RFC 1179. It turns out that this document was really just trying to capture the printing functionality in existing Unix systems at the time. As such, the document had the status of an informational document and never entered the IETF "standards track". Even so, the document formed the basis for many implementations by many vendors. The problem though, was that the scope of RFC 1179 was fairly limited and written for an environment of line printers and mostly unstructured streams of ASCII text. As the printer technology moved on to using cut sheets and page description languages (PDLs), there were no extensions defined for RFC 1179, and each vendor therefore started inventing their own extensions up to the point where today the concept of RFC 1179 as a "standard" is highly inaccurate.

And now finally, the birth of IPP. The initiative to the IPP was taken in a vendor-dominated, but fully open consortium called the Printer Working Group (PWG). Members of this group had previously been the promoters and main contributors to the "Printer MIB", a Management Information Base (MIB) [PRI MIB] for print devices using the Simple Network Management Protocol (SNMP) [SNMP]. After some initial work in the PWG, the concept of IPP was launched in a "Birds-of-a-Feather" session in the December 1996 meeting of the IETF, and chartered as an official working group of the IETF in March 1997. Due to earlier involvement in both the ECMA and ISO/IEC printing standards groups, as well as some experience from other areas in the IETF, I was chosen as one of the IETF working group chairs, with Steve Zilles from Adobe Systems as the other chair person. After initial discussions, it was found that much of the model and semantics defined by ISO/IEC 10175 could serve as input to the new project, but it was also decided to provide a mapping between IPP and RFC 1179 as part of the IPP project.

The IPP project is working along the guidelines used for all projects in the IETF, which means that all important discussions and decisions have to be made over the IETF e-mail list for the project. However, the PWG is providing additional opportunities for the more active participants in the form of weekly phone conferences and one to two day face-to-face meetings in the USA about every 6 weeks. The latter has made the project somewhat suspicious from an international point of view, but looking at the fact that most of the major printer and print server vendors are either based in the USA or has subsidiaries there, which is the case for all the major Japanese vendors, it is has turned out to be less of a problem in reality. Special efforts have been made to encourage European vendors to participate as well, but so far with limited success. The average time for an IETF protocol project to complete is about two and a half years, the IPP project hopes to do it in less than half that time. Some 30 experts from about 20 different companies or organizations are currently actively participating in the IPP specification work, with many more observers and sporadic participants in the wings.

3. Requirements

There are a number of requirements from different sources and of different kinds that IPP is trying to meet. It was clear that the sum of all possible requirements was considered too much to cover in a first version of the standard, and an early decision was to try to cover 80% of the problem, without being too exact about how the 80% was measured and what would go into the 80% versus the 20% bucket. Let me try to group the requirements into current and future printing requirements, some backwards compatibility requirements, and more general Internet and IETF requirements.

3.1. *Current and future printing requirements*

The functional printing requirements included the following:

- ability to find printers on the network
- ability to get a list of capabilities for a particular printer
- ability to send a print job to a selected printer

- ability to use modern PDLs
- ability to override PDL instructions with external instructions
- ability to start printing a job before all document data has been transferred
- ability to include several documents in a print job
- ability to instruct the printer to retrieve the document to be printed
- ability to check up on progress of a particular print job
- ability to cancel a previously submitted print job
- ability to get a notification from the printer in case of problems with a job
- the protocol should support both printer and print server configurations
- sufficient security for printing over the Internet
- fit in with modern Internet concepts, such as web browsing

A number of other requirements, in particular for special administrator and operator functions, printer installation, print driver installation, etc. were also discussed, but the agreement was to consider most of them as part of the 20% bucket, and leave them for possible inclusion in future versions of IPP. The most important outcome of this is that version 1.0 of IPP is limited to normal end user functionality, and that IPP, generally speaking, will provide more functionality than RFC 1179 and less than ISO/IEC 10175.

3.2. *Backwards compatibility requirements*

There were also a number of backwards compatibility requirements:

- enable gateways to be built to existing DPA implementations
- enable gateways to be built to existing LPD implementations
- enable gateways to be built from existing LPD implementations
- enable use of existing interfaces to print drivers
- use of existing PDL formats
- alignment with the Printer MIB specification
- alignment with an emerging Job Monitoring MIB specification [JOB MIB]

3.3. *General Internet and IETF requirements*

There were also a number of more generic Internet and IETF requirements, some of which were quite recent:

- use of the current Internet protocol stack, in particular TCP/IP
- evaluate the use of MIME types [MIME]
- include use of Internet security services as part of the IPP specification
- include support for internationalization of text strings etc.
- generally never reinvent the wheel if there are existing IETF solutions

3.4. *More details on the IPP requirements*

For more details about the IPP requirements, please refer to the document "Requirements for an Internet printing protocol" [IPP REQ].

4. IPP solutions

Let us now look at what solutions IPP provides to answer the requirements.

4.1. *Overall model and semantics components*

The overall model for IPP is the client/server paradigm, where the client usually resides in a PC or workstation, and the server is either embedded in a printer device or resides in a separate print server machine. Hence, the use of the words client and server in this article should be viewed as software entities, which can reside in different hardware environments. The client sends requests to the server, which then answers with responses. In IPP, the client always initiates the communication with the server.

The server, referred to as the "Printer" in the IPP specifications, contains two major types of objects, printers and jobs. Associated with the printer object are a number of attributes, which describe the different capabilities of that particular printer. The printer object also stores default values that are used if nothing is specified by the client. The printer object additionally contains status information that can be requested by the client. Similarly, each job object has a number of associated attributes, which describe client specified processing instructions for that job, as well as status information about how the job is progressing.

In addition to the attribute types described above, there are additional attributes called operation attributes, which are used in a request, but are typically not stored in a printer or job object.

The IPP needs to be able to identify printer objects and job objects uniquely throughout the Internet. It turns out that the already defined Universal Resource Identifier scheme (URI) [URI] serves this purpose very well for IPP. URIs are a superset of the more commonly known Uniform Resource Locator scheme (URL) [URL], which is used to identify web pages on the World Wide Web. The IPP now uses URIs for the following functions:

- to uniquely locate an IPP printer object on the network
- to uniquely locate an IPP job object on the network
- to provide address information for HTML [HTML] pages for printer descriptions and print driver download
- to uniquely locate a referenced document to be printed (more about this later)
- to provide addresses for delivery of asynchronous notifications

4.2. *Operations*

There are a number of operations associated with each of the main objects. The operations fall into two categories, mandatory operations and optional operations. Mandatory operations must be supported by all IPP servers, while the optional operations may or may not be available in a particular instance of an IPP server.

4.2.1. *Mandatory operations*

The mandatory operations for an IPP printer object are:

- Get-Attributes, which is used to find out about the printer's capability and state. It is also used to find out about which optional operations the printer supports.
- Print-Job, which is the operation normally used to submit a simple print job including one document. A job URI is returned.
- Validate-Job, which can be used to verify a set of job attributes, if such a job would be submitted. Similar to Print-Job, but with no document data included.
- Get-Jobs, which is used to find out about a printer's job queue.

The mandatory operations for an IPP job object are:

- Get-Attributes, which is used to recall the attributes associated with the job, and to find out about the job's state.
- Cancel-Job, which is used to cancel a job that users have changed their mind about.

Note that there is no change or modify operation, which means that if a user wants to change any of the originally specified job attributes, the job has to be canceled and resubmitted.

4.2.2. *Optional operations*

The optional operations covers two areas that were not considered necessary for every printer, but would be useful for more powerful printers and print servers. One is the ability to allow for several documents in a job, the other is to allow for documents to be printed using a reference instead of sending the actual document with the print request. The latter allows a user to include one or several references to documents in a print request and have the printer download these documents before printing them, thereby off-loading the user to first retrieve the documents and then send them again.

The optional operations for an IPP printer object are:

- Create-Job, which validates the job attributes and establishes a job object in the printer. No document is attached. A job URI is returned.
- Print-URI, which is similar to a Print-Job operation, but includes the reference to one document rather than the document itself.

The optional operations for an IPP job object are:

- Send-Document, which adds one document to an existing job that was created with the Create-Job operation. Several documents can be added to a job, but a separate Send-Document operaration is needed for each document.
- Send-URI, similar to Send-Document, but adds one document reference to an existing job. A separate operation is needed for each added document reference. Both documents and document references can be added to the same job.

For more information about the IPP operations, see [IPP MOD].

4.3. IPP functionality provided by other protocols

If you start comparing the list of IPP operations with the list of requirements, you will see that a number of requirements are not actually met by the IPP protocol itself. However, most of the "missing" functionality can be provided by existing Internet protocols in an actual IPP implementation. The list of functions provided in some other way are discussed below.

4.3.1. *Printer location*

There are several ways that the location of printers on the Internet can be provided.

- By putting the printer information in a network directory. An appendix of the IPP Model and Semantics document [IPP MOD] defines a generic directory schema for printer attributes that are suitable for inclusion in a directory. The Lightweight Directory Access Protocol [LDAP] is the most likely directory protocol for actual implementation of this, but other alternatives, including proprietary directory solutions, can also be used.
- By including your printer URIs on your organization's web pages.
- By putting your printer URI on your business card.

4.3.2. *Printer description*

Although the IPP printer object includes a number of capability descriptions for the printer, there may always be more information that you might want to know about a printer than can be described in the IPP standard printer attributes. For this purpose, a printer attribute has been defined which points the user to a web page with additional information. This web page can contain all the information that a user might want to know, such as detailed descriptions of the printer functions, a rotating 3D model of the printer, a complete on-line manual, and a map describing the printer location within a building. As all this can be provided with existing web technology, there was no need to try to duplicate this in the IPP.

4.3.3. *Print driver installation*

If you want to use a "new" printer somewhere out on the Internet, you probably do not have the right print driver so you need a way to find and download the right driver. Although IPP does not have an operation for this, a printer attribute is defined with a URL for a web page, from which you can download a driver. The Get-Attributes operation of IPP would be used to get the driver location URL.

4.3.4. *Downloading referenced document to the server*

The IPP operations Print-URI and Send-URI send references to the IPP server about documents to be printed, but there are no defined IPP operations for actually downloading the documents from the repositories where they are stored. The reference URIs actually include the Internet protocol to be used for the download. Current thinking in the IPP project is to mandate that the server support the File Transfer Protocol (FTP) [FTP], and optionally to support other protocols such as the Hypertext Transfer Protocol

(HTTP) [HTTP 1]. Due to limitations in current Internet security solutions, downloads can only be performed for documents that do not require special passwords, such as those accessible by anonymous FTP.

4.4. *More about IPP attributes*

IPP attributes have been mentioned earlier, but let us take a closer look at the different kind of attributes and give some examples to illustrate some of the more important features of the IPP. Attributes are identified by an attribute name (often called type in other standards). Some attributes can only contain one value (single value), while other attributes may have a sequence of values (multi-value) associated with the attribute name.

4.4.1. *Job template attributes*

The job template attributes describe aspects of how the features of the printer are used to print and finish the document and how to communicate with the user using asynchronous notifications. Most of the job template attributes can appear in three different forms, with slightly different semantics:

1. In a print job request submitted by the user
2. Printer supported attributes and values
3. Printer default attribute values

A common feature of the job template attributes is that they can be defaulted by the IPP server, which means that there is no need for a user to specify any of the attributes in a print request, provided that the user is happy with the predefined defaults.

IPP assumes some override rules if a specific printer function is requested both in a print request as well as in the document PDL format. For example, a Postscript file can contain instructions about double-sided printing and the number of copies to be printed. Today it is fairly uncommon that printers can actually override the PDL content, but the trend to put up more and more public documents on the Internet will increase the demand for this kind of capabilities in future printers, which is why this feature was considered important to include in the IPP architecture. Here are the current rules for overriding:

What a user specifies as attributes in a print request should override any corresponding parameters in the document PDL. There is a printer description attribute, which informs the user whether the printer can actually perform PDL overrides. What is in the document PDL should be used over any printer default values that have been set by the administrator of the printer. Printer default values are only used as a last resort if neither the user nor the PDL has specified anything for a function that the printer needs specification for. As a very last resort the print device may also have some built-in defaults, if there are no administrator defined defaults.

Let me now give some examples to illustrate which kind of information is found in the different sets of job template attributes:

User requested print job attribute values: copies, document format, finishings, media, notify addresses, notify events, number up, print quality, sides, user human language.

Printer supported attribute values: copies/collated copies, document formats, finishings, media ready/media supported, notify addresses supported, notify events, number up, print qualities supported, sides, user human languages.

Note that most attributes in this set are multi-value, while the client supplied attributes are typically single value.

Printer default attribute values: copies, document format, finishing, media notify events, number up, print quality, sides, user human language.

4.4.2. *Job description attributes*

Job description attributes is a set of attributes which provide information that has nothing to do with the actual printing or finishing of the print job and which, unlike the job template attributes, cannot be defaulted by the IPP server.

Examples of job description attributes are: job ID, job more info, job name, job originating user, job state, job state reasons, job URI.

It was earlier mentioned that print jobs were uniquely identified by their job URI. For backwards compatibility with existing print APIs in workstations, an alternative way to uniquely identify a job is to use a combination of the printer URI and the job ID attribute, and to direct the network request, e.g. for job status, to the printer object URI instead of the job object URI.

4.4.3. *Printer description attributes*

The printer description attributes describe the static capabilities of the printer as well as the printer state.

Examples of printer description attributes are: color supported, PDL override, printer info, printer location, printer language supported, printer name, printer state, printer state reasons, printer URI, security mechanisms supported.

4.4.4. *Operation attributes and the fidelity attribute*

Operation attributes are temporary attributes, usually not stored in an IPP object.

One particularly important operation attribute is the "fidelity" attribute. This attribute acts as a higher level attribute that influences the processing of most of the other attributes in a job request. This user submitted attribute, which has also been known as the "best effort" attribute, signal whether or not a user is prepared to accept substitutions if user specified attributes or attribute values are not supported. An attribute value stating that absolute fidelity is not required means that the IPP server will try to make every effort to get the document printed. This might include the use of auto-sensing to try to determine the document format, if that was unknown to the user. If full fidelity was requested however, then the IPP server will refuse to accept the print job if it cannot guarantee full fidelity.

4.5. *Attribute syntax and types for extensibility*

The IPP attributes are described using a simple ASCII oriented syntax called Augmented Backus-Naur Form (ABNF) [ABNF], a slightly updated version of the syntax used a decade and a half ago to define the Internet e-mail format in RFC 822 [822].

The IPP project has tried to make sure that the protocol is easily extendible, so that a revision of the standard will not be necessary for every new attribute that needs to get added to meet the continuing addition of new features in printers.

To meet this requirement, four attribute types have been defined:

Type 1 - attributes that are in the IPP standard set
Type 2 - attributes that get registered by the PWG and IANA
Type 3 - attributes registered by IANA by a vendor
Type 4 - customer installation defined attributes

IANA stands for the Internet Assigned Numbers Authority and keeps a number of lists related to Internet standards for which extensible sets have been defined. Types 1, 2 and 3 will be easily found by all implementers, only type 4 attributes are limited to an individual customer installation. Vendors should always register any extensions as type 2 or 3 with IANA.

For more information about the IPP attributes, see [IPP MOD].

4.6. IPP security

A more recent IETF requirement is that all new Internet protocols have to identify needs for security and include descriptions on how to use existing security services. IPP was one of the first projects to get hit with this requirement and considerable time and effort was spent on analyzing what was needed and what was really available. Defining the IPP security needs was not too difficult, but to find security solutions that were both approved by the IETF and realized in products turned out to be an almost impossible task.

4.6.1. *IPP security threats*

Before presenting the solutions, let us take a look at some of the security threats.

Printer spamming was considered one of the most obvious threats, which would probably take the form that somebody sends you massive documents for printing, resulting in "denial of service" to other users wanting to use the printer. Another side effect is that costly media might be used up for no purpose.

Other threats include unauthorized usage of a printer or certain features of a printer, by bypassing authentication and authorization controls using false identity or credentials.

There is also the risk that an IPP server might masquerade as belonging to a different organization or location than by providing a false identity.

Finally, the document content might contain private or confidential information which could be intercepted and read, modified or made to disappear.

4.6.2. *IPP needs for security services*

The generic security services needed to meet the threats above are:

- Authentication of the IPP client (user)
- Authentication of the IPP server (printer)
- Mutual authentication of client and server
- Encryption of request and document information

Note that authorization (access control) is considered an IPP server responsibility, based on the authentication service.

4.6.3. *IPP security solutions*

As indicated earlier, to actually find finalized and implemented Internet security solutions turned out to be difficult. A number of security project have been ongoing in the IETF for many years, but either the specifications were not yet approved, or there were not enough implementations generally available to satisfy our requirement for immediate implementation of IPP in a greater scale. The recommended solution in current drafts are:

- All IPP clients and serves must support the security features described in RFC 2068 [HTTP 1] and RFC 2069 [HTTP 2]. This covers the need for client authentication.
- The remaining security services are expected to be provided by the Transport Layer Security (TLS) [TLS] services, which is still under development in the IETF. As an intermediate solution, implementers are advised to use the Netscape defined Secure Socket Layer (SSL3) [SSL3] services. TLS will offer SSL3 as one of the more important options, but will also include additional negotiation mechanisms.

For more information about IPP security, see the [IPP MOD] and [IPP PRO] documents.

4.7. *IPP internationalization features*

Another, even more recent IETF requirement, is for all new Internet protocol to support multiple human languages and corresponding encoding schemes. At the time of writing, the IPP project is still struggling with the exact solution to this requirement, but the following can be said about the current status.

The user can find out from the IPP server, which languages are supported. It might for example support English, French, and Spanish. The user can then specify when sending a print requests or attribute request that he/she wants to get for example French back. The IPP server will then return all responses in the requested language. This multi-language capability is dependent on what the server actually implements and is limited to text strings. User interface texts for other values, such as standard error messages and state values, will need to be supported in the IPP client.

The final IPP specification is expected to require that international character sets such as Unicode, a subset of ISO/IEC 10646 [ISO10646], using international encoding schemes such as UTF-8 [UTF-8], be supported by all IPP clients and servers.

4.8. IPP protocol specification

The protocol specification for IPP has been another highly debated subject during the project development. Considering that IPP is a client/server type protocol, the first inclination was to use one of several possible Remote Procedure Call (RPC) mechanism defined for the Internet. However, over time more and more support was given to another proposal suggesting to map the IPP semantics on top of the Hypertext Transfer Protocol (HTTP), which is used by all web browsers. HTTP would offer some advantages in being available on most user's desktops and would provide a number of features that would be required for IPP anyway. Examples of the latter are the use of multiple requests/responses over one connection, and to use "chunking" for the transfer of larger documents. Another reason for using HTTP is that an increasing number of new printers will actually contain a small web server used for configuration, management, operator manuals, etc. The document "Rationale for the Structure of the Model and Protocol for the Internet Printing Protocol" [IPP RAT] gives more background to the choice of HTTP over other transfer mechanisms.

The version of HTTP that will be used for IPP is version 1.1 [HTTP 1], which is now available in the latest generation of web browsers and servers, and is the only version that is accepted by the IETF as candidate for an Internet standard. The POST method of HTTP will be used to send over the IPP requests to the server and get responses back to the client. Within the POST, the IPP request and response semantics is packaged in a newly defined MIME type called "application/ipp".

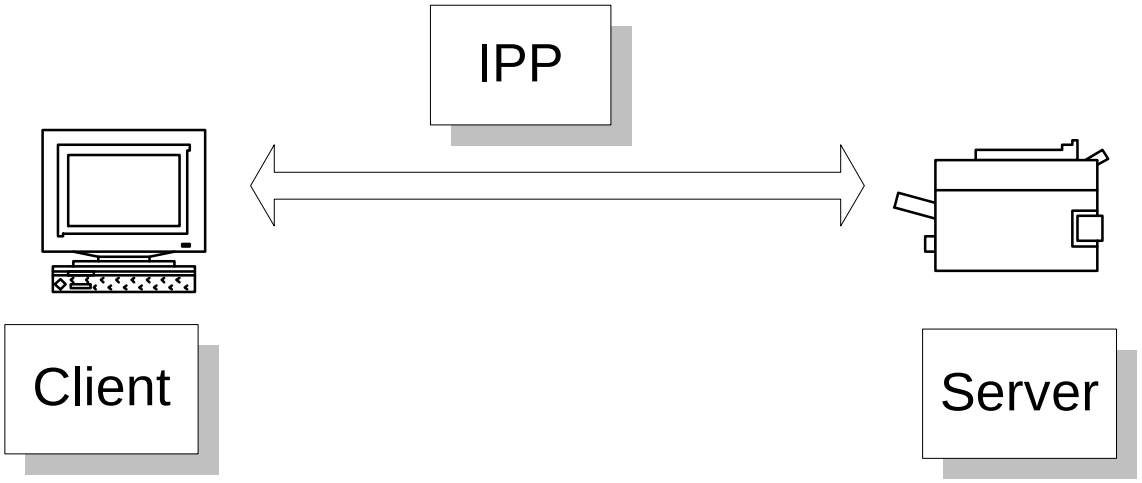
There is a lot more to learn about if you want to get all the technical details for the protocol, see [IPP PRO].

5. Some examples of possible IPP implementations

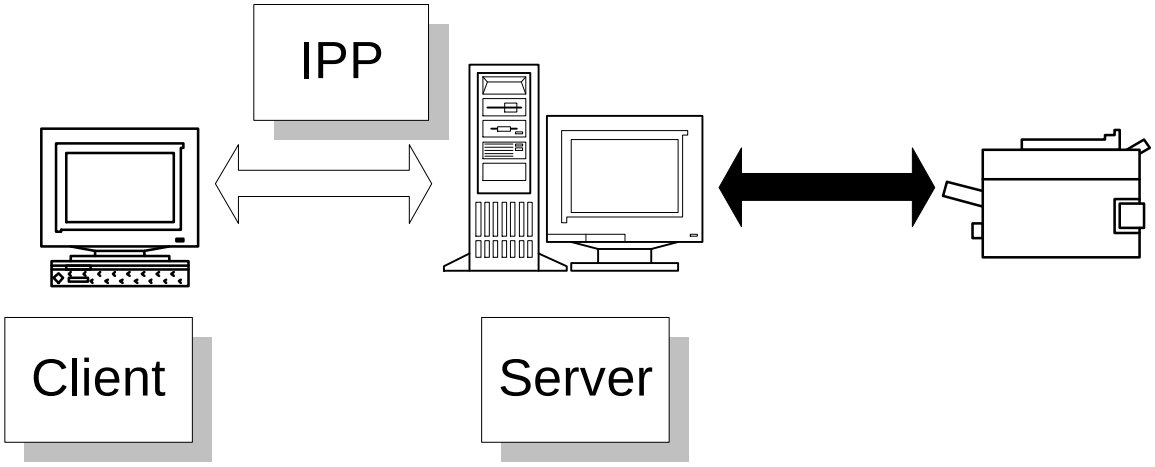
IPP has been designed to fit a number of different environments, from small office work group printers up to high volume print shop printers, as well as for print servers that might front end for a number of different print devices.

5.1 *Different configurations*

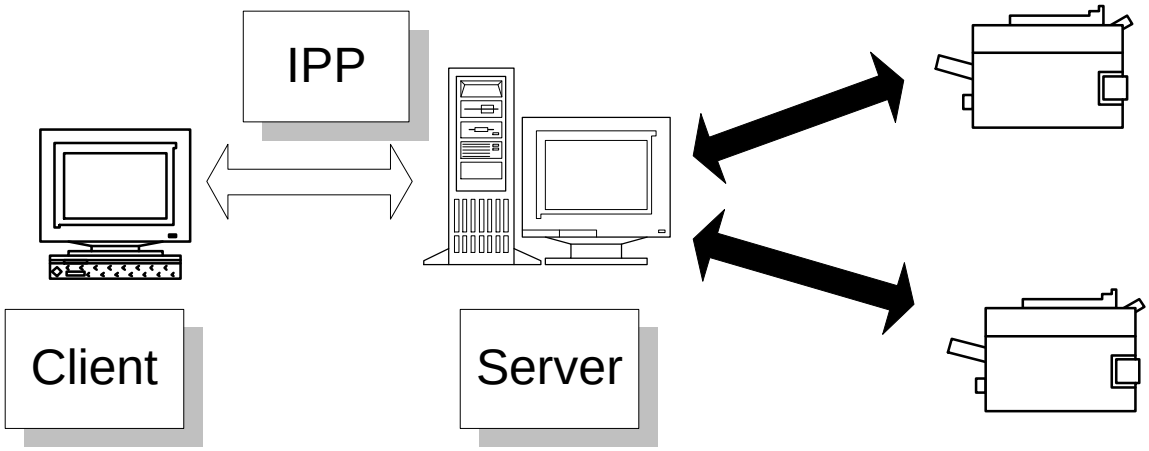
The figure below shows the simplest form of configuration with one client and one server, built into the print device.



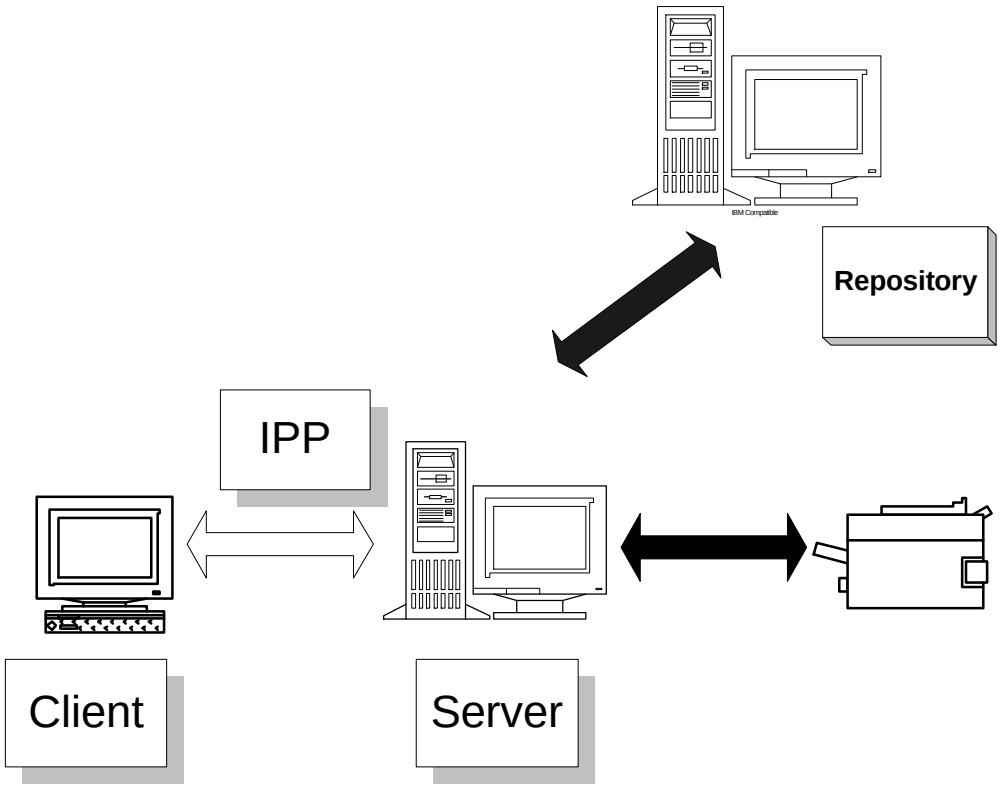
The next configuration shows a case where a separate print server is used between the IPP client and the print device. Note that the protocol used between the IPP server and the print device is not defined. The IPP server can act as a gateway which translates the IPP requests to an earlier printing protocol that can be understood by the print device. It is also possible to use the IPP between the IPP server and the print device.



The next configuration is similar to the previous one, but in this case, the IPP server is a front end for several print devices. The protocols used between the IPP server and the print devices can be the same or different for each device.



The last case illustrates a print-by-reference scenario in which the IPP server has got a document reference from the IPP client and then communicates with a document repository to download the document for printing. The protocol to use for the download is included in the URI for the document and is typically the FTP or HTTP protocol.



Yet another configuration, which is not illustrated, is that several IPP servers can be used as front-ends for the same printer device. This might be useful for instance in cases where you want to use a different URI for print jobs that come from outside a company's firewall, from the URI used by people inside the firewall.

5.2 Different applications

There are a number of different possible applications for using the IPP. Here are some examples:

- as replacement for a number of currently used LAN printing protocols
- as an alternative to facsimile
- as an access protocol to send print jobs to print shops
- as a way of sending documents for printing directly to a partner's printer, without having to first include them as e-mail attachments, or sending them ahead with a file transfer protocol
- as a way to distribute printed material out to shops or branches of a business chain
- as way to send printed advertising material to customers

6 Time table

By now, you will probably start asking about when IPP will be ready.

The IPP working group is currently in the final phase to complete the technical specifications. The next step is to issue the two main IPP documents on Model & Semantics, and on Protocol Specification to the Internet Engineering Steering Group (IESG) with the recommendation that they get accepted as

Proposed Internet Standards, and published as RFC documents. The attempt is to reach this still in 1997, but it might drift in to early 1998.

A number of companies, whose experts have been involved in the IPP specification work, and possibly others, are currently running IPP prototypes as preparation for products. My expectation is that you will see the first IPP products showing up on the market in 1998 with a wider spread over the next few years. Look out for IPP clients as component in your next generation PC and workstation operating system, and an increasing number of print devices and print servers ready to "speak" IPP.

If you want to follow up on the progression of the IPP specifications, please visit the IPP web site at:

http://www.pwg.org/ipp

There you will also find URLs to many of the referenced documents for downloading over the Internet.

7. References

Please note that many of the references below point to IETF documents that are temporary Internet-Drafts, which have no formal standards status and no guaranteed lifetime. If you do not find the draft listed below, look for a higher draft number or search for the title among recently published RFC documents.

[ABNF] D. Crocker et al., *Augmented BNF for Syntax Specifications: ABNF*, draft-ietf-drums-abnf-05.txt, October 1997, or later

[ECMA] ECMA-140, *Document Printing Application (*DPA), March 1990

[FTP] Postel, J. Reynolds, J., *File Transfer Protocol (FTP)*, RFC 959, October 1985

[ISO 10175] ISO/IEC 10175, *Document Printing Application (*DPA), June 1996

[ISO 10646] ISO/IEC 10646-1, *Universal Multiple-Octet Coded Character Set (UCS) -- Part 1: Architecture and Basic Multilingual Plane*, 1993

[HTML] Berners-Lee, T., *Hypertext Markup Language - 2.0*, RFC 1866, November 1995

[HTTP 1] R Fielding, et al, *Hypertext Transfer Protocol – HTTP/1.1*, RFC 2068, January 1997

[HTTP 2] J. Franks, et al, *An Extension to HTTP: Digest Access Authentication*, RFC 2069, January 1997

[IPP MOD] Isaacson, S. (editor) *Internet Printing Protocol/1.0: Model and Semantics* draft-ietf-ipp-model-06.txt, October 1997, or later version.

[IPP PRO] Herriot, R. (editor) *Internet Printing Protocol/1.0: Protocol Specification* draft-ietf-ipp-protocol-02.txt, October 1997, or later version

[IPP RAT] Zilles, S. *Rationale for the Structure of the Model and Protocol for the Internet Printing Protocol* draft-ietf-ipp-rat-01.txt, October 1997, or later version

[IPP REQ] Wright, D. *Requirements for an Internet Printing Protocol* draft-ietf-ipp-req-01.txt, October 1997, or later version

[JOB MIB] Hastings. T. et al, *Job Monitoring MIB* draft-ietf-printmib-job-monitor-06.txt, September 1997, or later version

[LDAP] Kille, S. et al, *Lightweight Directory Access Protocol*, RFC 1177, March 1995

[LDP] McLaughlin, L. III, (editor), *Line Printer Daemon Protocol*, RFC 1179, August 1990

[MIME] N. Freed, K. Moore, *MIME Parameter Value and Encoded Word Extensions: Character Sets, Languages, and Continuations*, RFC 2184, August 1997

[POSIX] Kirk, M. (editor*), Draft - POSIX System Administration - Part 4: Printing Interfaces, 1387.4/D8*, 1994

[PRI MIB] Smith, R. et al., *Printer MIB*, RFC 1759, March 1995

[SMTP] Postel, J., Simple Mail Transfer Protocol, RC 821, August 1982

[SNMP] Case, J. et al, *A Simple Network Management Protocol (SNMP)*, RFC 1157, May 1990

[SSL3] Netscape, *The SSL Protocol*, Version 3, (Text version 3.02) November 1996

[TLS] Dierks, T., Allen, C., *The TLS Protocol - Version 1.0*, draft-ietf-tls-protocol-03.txt, May 1997, or later version

[URL] Berners-Lee, T. et al., *Uniform Resource Locators (URL)*, RFC 1738, December, 1994

[URI] T. Berners-Lee, *Universal Resource Identifiers in WWW: A Unifying Syntax for the Expression of Names and Addresses of Objects on the Network as used in the Word-*

Wide Web, RFC 1630, June 1994

[UTF-8] Yergeau, F., *UTF-8, a transformation format of Unicode and ISO 10646*, RFC 2044, October 1996

[822] Crocker, D., *Standard for the format of ARPA Internet text messages*, RFC 822, August, 1982