

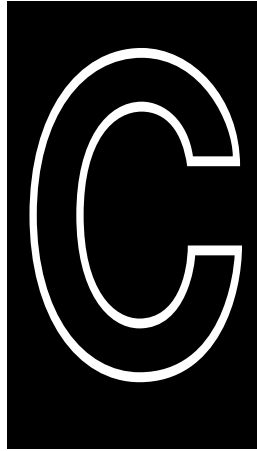


# Requirements and Design Goals for an Internet Printing Protocol

*F. D. Wright*

STRATEGIC AND TECHNICAL ALLIANCES, LEXMARK  
INTERNATIONAL

■ **In the Beginning . . .** For almost as long as there have been computers, there have been people working on standards for them. Computer designers and users soon realized that a computer by itself had a limited number of uses; computers need to communicate with other computers and with peripherals such as storage, input, and output devices.



Communication and networking standards groups like the Internet Engineering Task Force (IETF) have focused much more on the exchange of data between computers and the management of the computer and network infrastructure than on communications with peripherals. While the Internet Printing Protocol is currently defined as “experimental” by the IETF, it is expected to move to the “standards track” in the near future. When that happens, it will become the first “standards track” printing protocol. (RFC1179, “Line Printer Daemon Protocol,” is an informational document.)

The working group, composed of leaders in the printer and printing industries, spent much of its time early in the process developing a design goals or requirements document. That document was then used as a benchmark to assess the validity of design decisions.

## Vision

The Internet Printing Protocol (IPP) was created to provide a standard means of delivering print jobs to a printer using Internet technologies. The vision was that this technology would be used to enable a wide variety of new applications for printing as well as improved ways for existing applications to print. For example:

- Providing an alternative to fax machines that improves print quality and reduces cost through long-distance charge avoidance.
- Providing a means by which a newsletter or other time-sensitive document could be “pushed” to the printer of all subscribers using the Internet.
- Providing a means of using the “Distribute then Print” methodology for documents that must be printed, e.g., contracts to be signed.
- Providing a means to enable print and copy shops to receive print jobs through the Internet.

These are just a few of the uses envisioned for an Internet printing protocol. The rest of this paper will describe in more detail the technical goals and requirements necessary for this vision to become a reality.



## Definitions

The following definitions are used throughout this paper:

- end-user*: The person requesting a job to be printed or requesting information and status about a previously submitted print job.
- operator*: The person responsible for monitoring the status and condition of the printer as well as managing and controlling the jobs on that printer. This person is usually responsible for dealing with “intervention required” conditions such as out-of-paper.
- administrator*: The person responsible for “creating” the printer on the network and controlling the authorization of end-users and operators.
- printer*: An IPP printer can be either a single device or multiple devices which, taken in their entirety, provide IPP services and create output on media.
- URI*: Universal Resource Identifier (see IETF RFC1630).
- URL*: Uniform Resource Locator (see IETF RFC1630).

## Design Influences

In the process of creating the IPP, a consistent view, or model, of the printer and all its components was necessary. While IPP will be the first IETF standard for print job delivery, several other standards related to printers and printing have included a model for a printer. ISO 10175, Document Printing Application (DPA), and RFC1759, Printer MIB, both developed a model of a printer.

Because both of these standards were being developed at the same time, with significant participant overlap, there is good synergy between the models. The decision was made early in the development of IPP to remain consistent with these printer models whenever possible. Compromises and additions were required, but by making the IPP model consistent with the MIB and DPA models, applications that use IPP in conjunction with either DPA or SNMP management, or both, are easier to write and present a consistent view to the end-user.

## Architectural Requirements

At an architectural level, several objectives for the protocol became obvious to the working group:

### SCOPE

The task of developing a comprehensive printing application protocol is monumental. In order to deliver results on a reasonable time schedule, the working group began the definition of IPP by initially limiting the scope of the effort to the wants and needs of the end-user. In the long term, IPP will gradually be increased in function to address operator and administrator wants and needs. The actual timing and detailed plans for these subsequent versions is open.

Throughout this paper, this first version of IPP is referred to as “V1.0” and subsequent versions are generically referred to as “V2.0.”

## OPERATING SYSTEMS

In order to be widely deployed, the protocol defined needed to be independent of the operating system of both the client and the server. Generally, any platform capable of supporting a Web browser should be capable of being a client, and, likewise, any platform providing a Web/HTTP server and printing services should be capable of being a server. The actual usage of the Web Browser and Server is not required by the protocol itself; the operating system, operating system extensions, or other applications will provide IPP functionally directly.

## SPOOLING METHOD

In many environments, such as Windows 95, Windows NT and OS/2, the print data are created and transmitted to the printer on the fly rather than being created, spooled, and then transmitted to the printer (a typical UNIX method.) The Internet Printing Protocol must properly handle either methodology and make this transparent to the end-user.

## SECURITY

The Internet Printing Protocol must be designed to operate within either a secure or an insecure environment. Wherever reasonable, IPP must make use of pre-existing security protocols and services. IPP did not invent new security features or options. The security design goals described in this document can be met using existing protocols and services. Examples of such services include Secure Socket Layer Version 3 (SSL3), HTTP Digest Access Authentication [RFC2069], and Transaction Layer Protocol (TLS) [RFC2246].

Because the security levels or the specific threats that any given IPP print administrator may be concerned with cannot be completely anticipated, IPP must be designed to be capable of operating with different security mechanisms and policies as required by the individual installation. The initial security needs of IPP were derived from two primary considerations. First, the printing environments described in this document take into account that the client, the printer, and the document to be printed may each exist in different security domains. When objects are in different security domains, the design goals for authentication and message protection may be much stronger than when they are all in the same domain.

Secondly, the sensitivity and value of the content being printed will vary from one print job to another. For example, a publicly available document does not need the same level of protection as a payroll document. Message and content protection design goals include data origin authentication, privacy, integrity, and non-repudiation.

In many environments (e.g., Windows, OS/2) a



printer driver may be needed to create the proper datastream for the printer. It was not a requirement for IPP to include support for downloading a driver; however, it was a goal for IPP to enable the locating and downloading of drivers. Of course, downloading and installing any software, including drivers, on a computer exposes that computer to a number of security risks, including but not limited to:

- Defective software
- Malicious software (e.g., Trojan horses)
- Inappropriate software (i.e., software doing something deemed unreasonable by the user).

As such, proper security considerations and actions need to be taken by the user and/or a system administrator to prevent the compromising of the computer. These considerations and actions are outside the scope of IPP. (Examples including security considerations can be found in later sections of this document.)

## FIREWALLS

It was a basic requirement that an Internet printing protocol must support printing from one enterprise to another. As such, the IPP must be capable of passing through firewalls and/or proxy servers (where enabled by the firewall administrator), preferably without modification to the existing firewall technology.

## EXTENSIBILITY

An Internet Printing Protocol must be extensible by a variety of means to facilitate interoperability and prevent implementation collisions by:

- Providing a process whereby implementers can submit proposals for registration of new attributes and new enumerated values for existing attributes that:
  - > Require review and approval. The Internet Assigned Number Authority (IANA) will be the repository for such accepted registration proposals after review.
  - > Do not require review and approval. IANA will be the repository for such registrations.
- Providing syntax in the protocol so that implementers may add private, unregistered attributes and enumerated attribute values.
- Providing versioning and negotiation so as to enable future implementations of IPP to interoperate with implementations of V1.0 of IPP.

## Operational Requirements

Operation requirements for an Internet Printing Protocol address how end-users, operators, and administrators will use the protocol. These requirements are only for the protocol itself. Other wants and needs, such as the operator needing physical access to the printer (e.g., to be able to load paper or clear jams),

are not included in this work, or are included for completeness but specifically excluded as a requirement.

Specific scenarios, which provide more detailed examples of the entire process, including discovery, status, printing, and end-of-job reporting, are included later in this article.

## END-USER

The wants and needs of the end-user are broken down into six categories:

- (1) Finding or locating a printer,
- (2) Creating a local instance of a printer within the user's operating system,
- (3) Viewing printer status and printer capabilities,
- (4) Submitting a print job,
- (5) Viewing print job's status, and
- (6) Altering the attributes of a print job.

*Finding or Locating a Printer.* End-users want to be able to find or locate printers to which they are authorized to print. They want to be able to perform this function using a standard Web browser or other application. Using multiple criteria to find the printers is needed. These criteria include but are not limited to:

- Name (Printer 1, Joes-color-printer, etc.)
- Geographic location (building 1, Kentucky, etc.)
- Capability or attribute (color, duplex, legal paper, etc.)

Additionally, while it is outside the scope of IPP, end-users want to be able to limit the scope of their searching to:

- search inside a functional sub-domain
- include only a particular domain (lexmark.com)
- exclude specified domains

While an Internet Printing Protocol may not necessarily include this function, IPP must define and enable a directory schema which will provide the necessary information for a directory service implementation to consistently represent printers by their IPP attributes. This function could be provided by a directory, e.g., LDAP, or by a service location methodology or protocol.

*Creating a Local Instance of a Printer.* After finding the desired printer, an end-user needs to be able to create a local instance of that printer within the end-user operating system or desktop. This local instance will vary depending upon the printing paradigm of the end-user's operating system. For example, some UNIX users will only want a queue or a reference to a remote printer created on their machine, while other UNIX users, and Windows NT and other operating system users, will want the queue and



also the necessary icons and registry entries to be created and initialized. Where required, drivers may need to be downloaded from some repository and installed on the computer. All necessary decompressing, unpacking, and other installation actions should occur without end-user interaction or intervention, excepting initial approval by the end-user. (Driver download is outside the scope of IPP.) Once the local instance of the printer has been installed, it should appear to the end-user of the operating system and to the applications running there as any other printer (local, local area network-connected, or network operating system-connected) on the end-user desktop or environment. The role of IPP in this goal is simply to *enable* the creation of the printer instance, providing information such as where to locate a printer driver for this printer, as an attribute of an IPP Printer.

#### *Viewing the Status and Capabilities of a Printer.*

Before using a selected printer (or, in fact, at any time), the end-user needs the ability to verify the characteristics and status of both printers and jobs queued for that printer. When checking the characteristics of a printer, the end-user typically wants to be able to determine the capability of the device:

- supported media (commonly paper), by size and type
- paper handling capability, e.g., duplex, collating, finishing
- color capability

When checking the status of the printer and its print jobs, the end-user typically wants to be able to determine:

- Is this printer on-line?
- What are the defaults to be used for printing on this device?
- How many jobs are currently queued for this printer?
- How are job priorities assigned? (This is outside the scope of IPP.)

*Submitting a Print Job.* Once the desired printer has been located and installed, the end-user wants to print to that printer from normal applications using standard methods. These normal applications include such programs as word processors, spreadsheets, database applications, Web browsers, and production printing applications. Additionally, the end-user may want to print a file already existing on the end-user's computer—"simple push." In addition to printing from an application and simple push, the end-user needs to have the ability to submit a print job by reference. Printing by reference is defined to mean submitting a job by providing a reference to an existing document. The reference, a URI, will be resolved by the printer before the actual print process occurs. Submitting a job by reference relieves the user from downloading the document from the remote server

and then sending it via IPP to the printer, saving both time and network bandwidth.

Some means needs to be provided to determine if the format of a job matches the capability of the printer. This can be done by one of the following (all of which are outside of scope of the IPP protocol):

- The end-user selects the correct printer driver
- The printer automatically selects the proper interpreter
- The end-user uses some other manual procedure.

A standard action shall be defined should the job's requirements not match the capabilities of the printer.

Because the end-user does not want to know the details of the underlying printing process, the protocol must support job-to-printer capability matching; however, not all implementations are required to support this function. This matching capability requires knowing both the printer's capabilities and controllable attributes and those capabilities and attributes required by the job. Actions taken when a print job requires capabilities or attributes that are not available on the printer vary and can include but are not limited to:

- rejecting the print job
- redirecting the print job to another printer (not in V1.0)
- printing the job, accepting the differences in the appearance.

An Internet Printing Protocol must also support the submission of print jobs by background or batch applications without human intervention.

End-users need the ability to set certain print job parameters at the time the job is submitted. These parameters include but are not limited to:

- number of copies
- single- or two-sided printing
- finishing
- job priority

*Viewing the Status of a Submitted Print Job.* After a job has been submitted to a printer, the end-user needs a means to view the status of that job, i.e., is the job waiting, printing, or complete, and a means to determine where the job is in the print queue, i.e., how many jobs are ahead of it.

In addition to the need to inquire about the status of a print job, automatic notification of the completion of that job is also required. While out of scope for V1.0 of the protocol, some means of enabling and disabling this notification is needed.

*Canceling a Print Job.* While a job is waiting to be printed or has been started but not yet completed, the original creator/submitter of the print job (i.e., the originating end-user) must be able to cancel the job



entirely (job is waiting) or the remaining portion of it (job is printing). Altering the print job itself is not a V1.0 design goal.

## OPERATOR

The operator has the responsibility of monitoring the status of the printer as well as managing and controlling the jobs at the device. These responsibilities include but are not limited to the replenishing of supplies (ink, toner, paper, etc.), the clearing of minor errors (paper jams, etc.) and the re-prioritization of end-user jobs. The operator's wants and needs are not addressed by V1.0 of the protocol.

The wants and needs of the operator include all those of the end-user, but may include additional privileges. For example, an operator may be able to view all print jobs on a printer, while the end-user might only be able to see his/her own print jobs.

*Alerting.* One of the required operator functions is having the ability to discover (or preferably be alerted to) changes in the status of a printer (particularly those changes that cause a printer to stop printing), and to be able to correct those problems. As such, an Internet printing protocol must be able to alert a designed operator or operators to conditions such as "out of paper," "out of ink," etc. Additionally, the operator shall be able to, asynchronous to other printer activity, inquire as to a printer's or a job's status.

*Changing Print and Job Status.* Another of the required operator functions is the ability to make changes to printer and job status remotely. For example, the operator will need to be able to re-prioritize or cancel any print jobs on a printer to which the operator has authority.

## ADMINISTRATOR

The administrator has the responsibility of creating the printer instances on the network and controlling the authorization of other end-users and operators. The administrator's wants and needs are not addressed by V1.0 of the protocol.

The wants and needs of the administrator include all those of the end-user and, in some environments, some or all of those of the operator. Minimally, the administrator must also have the tools, programs, utilities, and supporting protocols available to be able to:

- create an instance of a printer, i.e., assign it a URI
- create, edit, and maintain the list of authorized end-users
- create, edit, and maintain the list of authorized operators
- create, edit, and maintain the list of authorized administrators
- create, customize, change, or otherwise alter the manner in which the status capabilities and other information about printers and jobs are presented

—create, customize, or change other printer or job features

—administrate billing or other charge-back mechanisms

—create sets of defaults

—create sets of capabilities

The administrator must have the capability to perform all the above tasks locally or remotely to the printer.

## Basic Usage Scenarios

By developing more detailed scenarios of how an Internet Printing Protocol might be used, the working group began the process of detailing the capabilities and functionality of the protocol. Seven basic usage scenarios were developed.

### PRINTER DISCOVERY

The objective of printer discovery is to locate printers that meet the client's wants and needs. While a Directory Service or Service Location Protocol may provide enough information for the client to make an initial choice, they will not generally provide the level of detail necessary in all cases. The client may have to connect to each individual printer offered to get more detail. Not all information available from the Directory Service is obtained by that service using IPP; some information may be administratively provided.

The actual protocol used between client and Directory Service is considered outside the scope of IPP. Printer discovery is included in the scenarios to provide design goals for the directory schema for IPP printers and to further define printer attributes.

Characteristics that might be considered when locating a printer include the following, some of which may be outside of scope of IPP in general or V1.0 specifically:

- capabilities of the printer, e.g., PDLs supported
- physical location, e.g., in Building 010
- driver required and its location
- cost per page to print whether or not printer is access controlled
- whether or not usage requires client authentication
- whether or not printer can be authenticated
- whether or not payment is required for printing
- maximum job size (spool size)
- whether or not the printer supports compression (outside the scope of IPP V1.0)
- whether or not the printer supports encryption
- administrative limits on this printer:
  - > maximum number of copies per job
  - > maximum number of pages per job

Responses could additionally include:



—how to get more information

- > web page
- > telephone number
- > help desk

#### DRIVER INSTALLATION

In this context, a driver refers to the software installed in some client operating systems to generate the print data stream for the intended printer. The actual details for installing a printer driver are operating system-dependent and are outside the scope of IPP. However, an IPP printer or a directory service advertising an IPP printer should be capable of telling a client what drivers are available and/or required, show where they can be found, and provide pointers to installation instructions, installation software, or initialization strings required to install the driver.

#### SUBMITTING A PRINT JOB

The protocol must support these sources of client data:

- Print data is a file submitted with the job
- Print data is generated on the fly by an application
- Print data is a file referenced by a URI

The protocol must handle overrun conditions in the printer and must support overlapped printing and downloading of the file in devices that are unable to spool files before printing them.

The submission of every print request must cause a response to be generated. Responses will indicate success or failure of the request and provide information on failures when they occur. Responses would include things like:

- The printer received the print job and queued it
- The printer received the print job and is printing it
- The printer received the print job, started to print it, but printing failed:
  - > why it failed (e.g., unrecoverable PostScript error)
  - > state of the printer
  - > how much was printed
- The printer received the print job but couldn't print it
  - > why it couldn't be printed
  - > state of the printer
- The printer received the print job but does not know what to do with it
- The printer did not receive a complete print job (e.g., communication failure)

#### MULTIPLE DOCUMENT PRINT JOBS

The protocol must support the submission of print jobs consisting of multiple documents. For example, a

print job could consist of one or more copies of multiple separately created documents, each to be printed in sequence, to create the print job. For example:

- Front matter
- Executive Summary
- Body of document
- Bibliography
- Index

Alternatively, a user could submit a print job asking for one or more copies of a series of independent documents. The protocol must be able to specify the collation order and document properties and attributes for each of these alternative scenarios.

#### GETTING STATUS/CAPABILITIES

Clients will need to be able to get information about:

- The static capabilities of the device
- The dynamic state of the printer (e.g., out of paper)
- The state of a specific job owned by this client
- The state of all jobs owned by this client
  - > queued
  - > printing
  - > completed
- The job submission attributes supported and/or required
  - > scheduling attributes (e.g., priority)
  - > production attributes (e.g., number of copies)

#### ASYNCHRONOUS NOTIFICATION

Clients, i.e., end-users, must be able to request asynchronous notification for printer events such as

- job completion
- a fatal error that requires the job to be resubmitted
- a condition that severely impacts a queued job for this client, e.g., printer is out of paper

End-user notification is a V1.0 design goal, while operator notification is planned for V2.0.

#### JOB CANCELING

Similarly, clients must be able to make changes to jobs which have been submitted and are queued for printing. Changing of job attributes should also be supported. Job modifications such as the holding and releasing of jobs are not included in the design goals for IPP V1.0. Job canceling is a design goal for IPP V1.0.



## Why HTTP?

As previously stated, it was a goal of the group to define the protocol as quickly as possible, using existing technologies wherever possible. Therefore, an early decision of the working group was to utilize the Hypertext Transfer Protocol (HTTP) as the transmission mechanism for IPP. Specifically, the chosen mechanism for transmitting the encoded IPP Model (as described in RFC2566) data is HTTP 1.1 Post (and the associated responses). No modifications to HTTP 1.1 are proposed or required. The sole role of the transmission mechanism is to provide a transfer of encoded IPP Model data from/to the client to/from the server. This could be done using any data delivery mechanism. The key reasons why HTTP 1.1 POST is used are given below. The most important of these is the first. With perhaps this exception, these reasons could be satisfied by other mechanisms. There is no claim that this list uniquely and completely determines a choice of mechanism.

- (1) HTTP 1.0 is already widely deployed and, based on the recent evidence, HTTP 1.1 is being widely deployed as software developers release new products. The performance benefits of HTTP 1.1 have been shown and developers are reacting positively. Wide deployment has meant that many of the problems of making a protocol work in a wide range of environments, from a local area network to an internet to the Internet, have been solved and will stay solved with HTTP 1.1 deployment.
- (2) HTTP 1.1 solves most of the problems that might have required a new protocol to be developed. HTTP 1.1 allows persistent connections that make a multi-message protocol more efficient; for example, it is practical to have separate Create-Job and Send-Document messages. Chunking allows the transmission of large print files without having to pre-scan the file to determine the file length. The HTTP accept headers allow the client's protocol and localization desires to be transmitted with the IPP operations and data. If the IPP Model were to provide for the redirection of job requests, such as Cancel-Job, when a job is moved, the HTTP redirect response allows a client to be informed when a job he is interested in has been moved to another server/printer for any reason.
- (3) Most network printers will be implementing HTTP servers for reasons other than IPP. These network-attached printers want to provide information on how to use the printer, its current state, HELP information, etc. in HTML. This requires having an HTTP server, which would be available for IPP functions as well.
- (4) Most of the complexity of HTTP 1.1 is concerned with the implementation of HTTP proxies and not the implementation of HTTP clients and/or servers. Work is proceeding in the HTTP Working

Group to help identify what a server must do. The required part of HTTP 1.1 that must be implemented, as defined by the protocol document, is relatively small.

- (5) One of the requirements for IPP is support for printing by reference. These documents could be retrievable via HTTP, FTP, or some other Internet file transfer protocol. HTTP implementations provide support for handling a variety of URLs that would have to be provided if a new protocol were defined.
- (6) Security is an obvious requirement for an Internet Printing Protocol. An HTTP-based solution fits well with the Internet security mechanisms that are currently deployed or being deployed. HTTP will run over SSL3 or TLS. The Digest Access Authentication mechanism of HTTP 1.1 provides an adequate level of access control for many environments. These solutions are deployed and in practical use; a new solution would require extensive use to have the same degree of confidence in its security. Note: SSL3 is not on the IETF standards track.
- (7) HTTP provides an extensibility model that a new protocol would have to develop independently. In particular, the headers, content-types (via Internet Media Types), and error codes have wide acceptance and a useful set of definitions and methods for extension.
- (8) As a part of the Working Group's goal of rapid development and deployment, the need for pre-existing prototype and debug tools is obvious. There are many prototyping tools that work with HTTP, CGI scripts can be used to test and debug parts of the protocol.
- (9) Finally, the POST method was chosen to carry the print data because its usage for data transmission has been established. It works and the results are available via CGI scripts or servlets. Creating a new method would have better identified the intended use of the POSTed data, but a new method would be more difficult to deploy. Assigning a new default port for IPP provided the necessary identification with minimal impact to installed infrastructure, so was chosen instead.

## Conclusion

The Internet Printing Protocol was created to provide a standard means of delivering print jobs to a printer using Internet technologies. Whether it is used as an alternative to fax machines, as a means to push information to a printer, or for any of the other uses envisioned for it, the requirements for these uses have been strongly considered and loudly debated by the printing industry. It is expected that IPP will be a base upon which printing can exist for years to come. **SV**



## Bibliography

RFC1179—Line Printer Daemon Protocol.  
RFC2069—An Extension to HTTP: Digest Access Authentication.  
RFC2246—The TLS Protocol Version 1.0.  
RFC2567—Design Goals for an Internet Printing Protocol.  
RFC2568—Rationale for the Structure and Model and Protocol for the Internet Printing Protocol.

RFC2566—Internet Printing Protocol/1.0: Model and Semantics.  
RFC2565—Internet Printing Protocol/1.0: Encoding and Transport.  
RFC2639—Internet Printing Protocol/1.0: Implementer's Guide.  
RFC2569—Mapping between LPD and IPP Protocols.

---

© ACM 1067-9936/99/-\$5.00