INTERNET-DRAFT   Roger deBry
                                                       IBM Corporation
                                                          T. Hastings
                                                     Xerox Corporation
                                                          R. Herriot
                                                     Sun Microsystems
                                                       Scott Isaacson
                                                         Novell, Inc.
                                                      dateNovember 1996

            Internet Printing Protocol/1.0: Directory Schema   IPP/1.0
                      draft-isaacson-ipp-info-00.txt
                         Expires May 27, 1997

Status of this Memo

    This document is an Internet-Draft. Internet-Drafts are working
    documents of the Internet Engineering Task Force (IETF), its areas,
    and its working groups. Note that other groups may also distribute
    working documents as Internet-Drafts.

    Internet-Drafts are draft documents valid for a maximum of six months
    and may be updated, replaced, or obsoleted by other documents at any
    time. It is inappropriate to use Internet-Drafts as reference
    material or to cite them other than as "work in progress."

    To learn the current status of any Internet-Draft, please check the
    "1id-abstracts.txt" listing contained in the Internet-Drafts Shadow
    Directories on ftp.is.co.za (Africa), nic.nordu.net (Europe),
    munnari.oz.au (Pacific Rim), ds.internic.net (US East Coast), or
    ftp.isi.edu (US West Coast).

Abstract

    This Internet-Draft specifies an Internet Printing Protocol (IPP)that
    is intended to be version 1.0. This protocol is heavily influence by
    the semantic operations and attributes defined in ISO/IEC 10175
    Document Printing Application (DPA) parts 1 and 3. It also
    incorporates some of the implementation and interoperability lessons
    learned from other printing related standards such as POSIX System
    Administration - Part 4 (POSIX 1378.4) and X/Open A Printing System
    Interoperability Specification(PSIS).

    IPP is defined as a set of abstract data types and operations. The
    operations are implemented using a simple request and response
    mechanism built on top of HTTP. The abstract data types are encoded
    as simple ASCII text strings.

47   The IPP protocol covers only end user operations on basic print
48   service objects. Authentication is realized by mechanisms outside the
49   scope of the protocol, but the protocol does introduce some access
50   control functionality so that only authorized end users are allowed
51   to submit print jobs to printers whose implementation and site policy
52   support access control. Also, the Cancel Job operation requires some
53   authentication so that jobs can only be canceled by the end user who
54   submitted the job. Extended monitoring and management is possible
55   through other protocols such as the SNMP Printer MIB. In the areas
56   where there are no existing standards, some proposed and emerging
57   standards are being worked (management, security, etc.). As these
58   services become more stable, this document (and hence the protocol)
59   can be updated to reflect the integration and relationships with
60   these other standards.

61 Table of Contents

201 1. Introduction

202   The Internet Printing Protocol (IPP) is an application level protocol
203   that can be used for distributed printing on the Internet. The
204   protocol is heavily influenced by the printing model introduced in
205   the Document Printing Application (ISO/IEC 10175 DPA) standard, which
206   describes a distributed printing service. DPA identifies the end user
207   and administrative roles associated with a distributed printing
208   service, and defines the set of operations supported by the service.
209   This IPP specification (version 1.0) deals only with the end user
210   role. These ideas and concepts, when unified with other Internet
211   protocols and services, realize a distributed print service for the
212   Internet.

213   This specification uses the verbs: "shall", "should", "may", and
214   "need not" to specify conformance requirements as follows:

215   - "shall": indicates an action that the subject of the sentence
216   must implement in order to claim conformance to this specification

217   - "may": indicates an action that the subject of the sentence does
218   not have to implement in order to claim conformance to this
219   specification, in other words that action is an implementation
220   option

221   - "need not": indicates an action that the subject of the sentence
222   does not have to implement in order to claim conformance to this
223   specification. The verb "need not" is used instead of "may not",
224   since "may not" sounds like a prohibition.

225   - "should": indicates an action that is recommended for the subject
226   of the sentence to implement, but is not required, in order to
227   claim conformance to this specification.

228 2. Distributed Printing

229   This document assumes a distributed computing environment where
230   requesters of print services (clients, applications, PC drivers,
231   etc.) cooperate and interact with print service providers. Although
232   the underlying configuration may be a complex n-tier client/server
233   system, an important simplifying step in this protocol is that the
234   only object the requester of the print service ever sees is a
235   "printer". It is important, however, to understand that in a real
236   system, other components of a print service exist.

237 1.1 Generic Print System Components

238   Every distributed print service, including those using the Internet
239   Printing Protocol, includes elements from the following list.

240   — End Users: End Users are humans (or agents or applications who
241   work on behalf of a human) who submit print jobs.

242   — Print clients: Print clients are computer network nodes with
243   which humans interact in order to manipulate the distributed print
244   service. A print client uses some protocol to invoke print service
245   operations on another node. Each operation has arguments and
246   results associated with it. The print client provides arguments
247   which add information about the operation requested, and receives
248   results which describe the status and outcome of the operation.

249   — Print servers: Print servers may be embedded in an output device
250   or implemented in a separate system which is associated with an
251   output device. The print server receives requests from the print
252   client and sends back results which describe the status and
253   outcome of the operation requested. A print server normally
254   provides queuing, job management, and device management functions.

255   — Queues: Print jobs may be queued or stored on a spool prior to
256   printing. This allows a print service provider to accept one or
257   more print jobs while the printer (or printers) is busy processing
258   another job. Queues, if present, may be implemented in the client,
259   in the server, in the output device, or in some combination of the
260   three.

261   — Output Devices: Output devices interpret the print data and
262   generate some form of output. In the case of a laser printer, for
263   example, this normally means rasterizing the print data and
264   putting the resulting marks on paper. An output device may
265   receive print data directly from a client or through a Print
266   server.

267   A specific implementation of a print service may not include all of
268   the elements described here, and the physical packaging of elements
269   is up to the implementation. For example, an output device may
270   include a queue or a print server may include a rasterizer.

271   1.1 IPP Components

272   The print model defined by the Internet Printing Protocol simplifies
273   the user's view of the system components described in the previous
274   section by encapsulating the important elements of the system into
275   five simple objects:

276   — End Users (no specific object definition via attributes)
277   — Clients (no specific object definition via attributes)
278   — Printers (section 6.4)
279   — Print Jobs (section 6.2)
280   — Job Templates (section 6.5)
281
282   Clients use the following operations:

283   Print (section 5.4.1)
284   Cancel Job (section 5.4.2)
285   Get Attributes (section 5.4.3)
286   Get Jobs (section 5.4.4)
287
288   2. IPP Objects

289   This section describes the IPP objects.

290   1.1 Printer

291   One of the most significant objects in the IPP model is the Printer.
292   To the end user, the Printer object represents the functionality of
293   the actual output device along with the queuing, job management, and
294   device management functions often associated with a print server. An
295   IPP Printer object implements the Internet Printing Protocol. Using
296   the protocol, end users may query the attributes of the Printer,
297   submit jobs to the Printer, determine subsequent states of submitted
298   and queued jobs and state of the Printer, and cancel their own print
299   jobs. The realization of a Printer object may take on different forms
300   for any given configuration of real components. However, the details
301   of the configuration of real components must be transparent to the
302   end user.

303   In addition, a Printer is an abstraction for any document Output
304   Device. This means that a Printer could be used to represent any
305   real or virtual device which can support the Printer operations and
306   interfaces. For example, a Printer could be used to front end a fax
307   out device, any kind of imager, or even a CD writer.

308   Some examples of configurations containing IPP Printer object
309   include:

310   An output device, with no spooling capabilities, supporting IPP
311   An output device, with a built-in spooler, supporting IPP
312
313
314   A print server with one or more associated output devices with
315   the print server supporting IPP.
316   The associated output devices may or may not be capable of
317   spooling jobs
318   The associated output devices may or may not support IPP
319
320   See the following figures for some examples on how to view IPP
321   Printer objects on top of other printing system models:

```
322      Legend:
323
324      ##### indicates an IPP Printer object which is
325            either embedded in an output device or is
326            hosted in a server.  An IPP Printer object
327            may or may not queue/spool.
328
329      any   indicates any network protocol or direct
330            connect, including IPP
331
332
333      embedded printer:
334      ─────────────────────────────────────── output device
335      ─────────────────────────────────┬───────────────────┬
336       O    ┌───────┐      ┌───────┐   │  ###########      │
337      /|\   │ client│      │  IPP  ───>│ # Printer #      │
338      / \   └───────┘      └───────┘   │  ###########      │
339      ─────────────────────────────────┴───────────────────┴
340
341
342      hosted printer:
343      ─────────────────────────────────┬───────────────────┬
344       O    ┌───────┐   ┌  ###########  │                   │
345      /|\   │ client│   IPP ># Printer # any >│ output device │
346      / \   └───────┘   └  ###########  │                   │
347      ─────────────────────────────────┴───────────────────┴
348
349
350      ─────────────────────────────────┬───────────────────┬
351      fan out:                         │                   │
352      ──────────────────────────┌  >│ output device │
353                         any/    │                   │
354       O    ┌───────┐   ┌  ###########  /   │               │
355      /|\   │ client│   IPP ># Printer # *
356      / \   └───────┘   └  ###########  \   │               │
357                         any\    │                   │
358      ──────────────────────────┌  >│ output device │
359      ─────────────────────────────────┤                   │
360      ─────────────────────────────────┴───────────────────┴
361
362
363      ──
364 ──
365  1.1 Job
366      A Job object is used to model a job.  A job can contain  one or more
367      documents.  However, there are no separate document objects.  The
368      impact of this is that there are no attributes that pertain to one
```

369 document in a job but not to others, except for a single attribute
370 that specifies the document data, its location, and its format. Note:
371 In future versions, documents may become separate objects with
372 attributes whose scope and application are different from the
373 corresponding job attributes.

374 Job attributes are broken up into the following groups:

375   - Job Informational (sections 6.2.1, 6.2.2)
376   - Job Status (section 6.2.3)
377   - Job Sheet (section 6.2.4)
378   - Notification (section 6.2.5)
379   - Job Scheduling (section 6.2.6)
380   - Job Production (section 6.2.7)
381   - Conversion of Text Files (section 6.2.8)
382   - Job Resources (section 6.2.9)
383   - Number of Documents (section 6.2.10)
384   - Document Attributes (6.2.11)

386 1.1 Job Template

387 A Job Template object is used to model job defaults. A Job Template
388 is essentially a set of job attributes that initialize a newly
389 created job object.

390 Issue: The notion of Job Template needs more work.

391 1.1 Object Relationships

392 Instances of objects within the system have relationships which must
393 be maintained persistently along with the persistent storage of the
394 objects themselves. A Printer can contain zero or more Job objects.
395 Therefore, a job object is contained in exactly one Printer object.
396 A Job object contains one or more Documents.

397 A Printer object is associated with zero or more Job Template
398 objects.

399 1.1 Object Identity

400 All instances of all objects have an identifier attribute that makes
401 them unique so that they can be unambiguously referenced.

402 The following objects have the following mandatory identifier
403 attributes:

Object	Identifier	Containing Object
Printer	printer name	None
Job	job identifier	Printer
Job Template	job-template-name	None

404

405   2. Naming

406   Clients identify Printer objects by using an HTTP type URL. For
407   example, a URL for a Printer object named "printer-1" whose network
408   node's domain name is "some.domain.com", might look like:

409   http://some.domain.com/printer-1

410   In this case, the URL identifies the use of the HTTP protocol. The
411   Printer is located at the node identified by the DNS name
412   "some.domain.com" and "printer-1" is the name of the Printer.

413   Another example is the following URL:

414   http://1.2.3.4:nnn/printer-2

415   In this case, the URL identifies the use of the HTTP protocol. The
416   Printer is located at the node identified by the IP address of
417   "1.2.3.4" using port nnn for the HTTP server, and "printer-2" is the
418   name of the Printer. (The actual value of nnn is to be assigned by
419   IANA as part of this standards project).

420   It is not necessary to expose the Job Template objects that might be
421   associated with a given printer as separate objects. They can be
422   exposed in two ways through URL naming.

423   -   The Job Template can be hidden from the end user by a URL that
424   represents just the Job Template name (but does not expose the
425   Printer object name) as the two URLS

426   1) http://some.domain.com/two-sided-printer, and
427   2) http://some.domain.com/draft-printer.
428
429   These look like two different Printers , but underneath they
430   represent the same Printer object, but that Printer object has two
431   associated Job Templates and each is exposed through a different
432   URL for the same Printer object. Each one of the Job Templates
433   specified by a URL would contain a different Job Template default
434   attribute set. One Job Template would contain the defaults for
435   two-sides printing and the other would contain the defaults for
436   draft printing.

437   -   The Job Template can be exposed along with the name of the Printer
438   object directly in the URL as in:

439   1) http://some.domain.com/hr-printer/resumes
440   2) http://some.domain.com/hr-printer/1040forms

441   In this case there are "resumes" and "1040forms" Job Templates
442   associated with the "hr-printer" Printer.

443   This specification establishes, through IANA, a new well known port,
444   port nnn, for the use of IPP over HTTP. The purpose of this new well
445   known port would be to distinguish printing from non-printing
446   content. While any acceptable HTTP content could be inter-mixed over
447   HTTP well known port 80, only IPP printing would be acceptable on
448   port nnn.

449   2.1 Directory Services

450   IPP does not require any specific directory service. However, this
451   specification does define a generic schema that can be used for any
452   specific instance of a directory service. That is, some of the
453   attributes from the Printer object are called out as attributes that
454   may be added to a directory entry which represents that Printer.
455   This allows directory users to find and locate IPP Printers by either
456   a simple name look up or by some filtered attribute search.
457
458
459   2.2 Directory Entry Schema

460   The following attributes define the generic directory entry schema.
461   All directories entries for IPP Printers in all types of directories
462   should support at least these attributes.

463   Issue: The use of "objective" attributes vs. "subjective" attributes
464   still needs to be resolved. For example, for Maximum Print Quality
465   is it better to have values like "high", "medium", "low" or to have
466   explicit, quantified, measurable values? Some of the issues are: end
467   users don't often know what explicit objective values are or what
468   they really mean and they want to depend on an administrator to
469   define what is "high" quality printing and what is "low" quality,
470   especially since today's objective values that equate to "high" are
471   tomorrow's objective values that equate to "medium". On the other
472   hand, some end users demand the control and power explicit values can
473   give them when they do filtered searching. For example, they know
474   and appreciate the difference between 20 ppm printers and 23 ppm
475   printers.

476   Issue: We must specify which attributes are "mandatory" and which are
477   "optional". LDAP uses the terms "must" and "may" to identify
478   attributes that "must" appear and attributes that "may" appear in a
479   given entry in the directory.

480   2.2.1 Name

481   This directory attribute is the printers name. It is a URL so it
482   contains sufficient information to not only name, but to address the
483   printer using IPP as well.

484   2.2.2 Description

485   This directory attribute is a free form string that can contain any
486   site-specific descriptive information about this printer.

487   2.2.3 Location

488   This directory attribute is a free form string that can contain any
489   site specific location information.

490   In order for filtered searches to be more effective, a given site may
491   use some regular structuring within the string values such as
492   "SITE:USA-San Jose,BUILDING:A1,FLOOR:2,ROOM:555" or "department5-
493   2ndFloor-A5-IndianHills-Chicago-IL-USA".

494   2.2.4 Maximum Print Quality

495   This directory attribute indicates a somewhat subjective evaluation
496   of the overall printing quality. The syntax and values shall be the
497   same as for the print-quality Job attribute.

498   2.2.5 Cost

499   This directory attribute indicates a somewhat subjective evaluation
500   of the overall cost of printing at this printer: "high", "medium", or
501   "low".

502   2.2.6 Resolution

503   This directory attribute is the maximum resolution of the Printer in
504   dpi.

505   The syntax and semantics shall be the same as for the printer-
506   resolution-select job attribute.

507   2.2.7 Color Supported

508   This directory attribute specifies whether the Printer supports color
509   and, if so, what type. The values are a type2Enum (see section 6).
510   Standard values are: "none", "highlight", "three color (CMY)", "four
511   color (CMYK)", "monochromatic".

512   2.2.8 Fonts Supported

513   This directory attribute takes on a list of fonts that are supported
514   by the printer. The syntax and values shall be the same as for the
515   fonts-used job attribute..

516 2.2.9 Maximum Speed

517   This directory attribute is the maximum speed of the printer ppm,
518   ipm, spm, lpm, or cps. The syntax and values shall be the same as
519   for the maximum-printer-speed Printer attribute.

520 2.2.10 Device Id

521   This directory attribute can be used for automatic driver download,
522   database access, or other automatic configuration tasks. It might be
523   used to generate a platform specific id such as the Windows Plug-and-
524   Play id.

525   Issue: Is this the IEEE 1284-1994 device id, the Object Identifier as
526   used in the Host Resource MIB hrDeviceId object, or some other
527   identifier?

528 2.2.11 Make and Model

529   This directory attribute is a simple text string defined by the
530   manufacturer that contains some reference to the make and model of
531   the entity being represented to the end-user by this Printer object.
532   The syntax shall be:

533   vendor-name "/" model-name

534   where the vendor-name is the same as that registered with IANA for
535   use in domain names.

536   For example: "vendor-x/super-duper-printer".

537 2.2.12 Marker Type

538   This directory attribute is the printing mechanism of the print
539   device: electrophotographic-laser, inkjet-aqueous, thermal-transfer,
540   etc. The syntax and values shall be the same as for the printer-
541   types Printer attribute, except the value of the Marker Type
542   directory attribute shall be single-valued

543 2.2.13 Document Formats Supported

544   This directory attribute is a list of all of the document formats
545   that the printer and/or its interpreter(s) support. The syntax and
546   values shall be the same as for the document-format Job attribute.

547 2.2.14 Sides Supported

548   This directory attribute specifies the capabilities of the Printer
549   for marking on sides of the medium. The syntax and values shall be
550   the same as the sides Job attribute.

551   2.2.15 Finishings Supported

552   This directory attribute identifies the finishing operations
553   supported by the Printer. The syntax and values shall be the same as
554   the finishing job attribute.

555   2.3 Directory Entries Using LDAP

556   To allow directory users to locate an IPP Printer, a corresponding
557   entry must be defined within a directory. This section describes how
558   this is done using the Lightweight Directory Access Protocol (LDAP).

559   The LDAP directory entry includes the name of the entry and the
560   attributes as defined in "4.2 Directory Entry Schema". The following
561   is an example of how to define a directory entry for a Printer object
562   using LDAP. It is given to assist the reader's understanding of this
563   specification.

564   To create a Printer object directory entry using LDAP:

565   1. An administrator uses a program to create an entry for the Printer
566   object on a directory server that supports LDAP. The administrator
567   defines the Distinguished Name (dn) and the default subjective
568   attributes for the Printer object directory entry.

569   Issue: Should the administrator also define default objective
570   attributes or wait for the Printer object itself to initialize these
571   attributes?

572   2. The Printer object invokes the ldap_open API to open a connection
573   to the directory server:

574   Example: ld=ldap_open ("dir.host.name", LDAP_PORT)

575   where ld is the connection handle for subsequent LDAP APIs.

576   3. The Printer object invokes an ldap "bind" API to authenticate with
577   the directory server.

578   Example: ldap_simple_bind_s (ld, dn, NULL) (which does a simple
579   authentication without a password).

580   4. The Printer object invokes the ldap_modify or ldap_modify_s API to
581   define the objective attributes for the Printer object entry as
582   identified by its Distinguished Name (dn).

583   Example: ldap_modify_s (ld, dn, mods) (where mods is a NULL-
584   terminated array of objective attributes and values to add or modify
585   in the directory entry)

586   5. The Printer object invokes the ldap_unbind API to close the
587   connection to the directory server.

588   Example: ldap_unbind (ld)

589   When one or more objective attributes are modified for a Printer
590   object, the Printer object repeats steps 2-5 to update the modified
591   objective attributes in its directory entry.

592   To locate a Printer object entry using LDAP, a program can use the
593   ldap_search or ldap_search APIs or a user can specify an LDAP URL.

594   For example, to locate all Printer objects that support duplex, a
595   user can specify URL:

596   ldap:///dir.host.name???(&(objectClass=printer)
597   (sides-supported=2-sided-long-edge))
598
599   Issue: Is it allowed to filter the search based on the object class
600   itself, in this case the object class of Printer? We need to define
601   this new object class. How do we do this? One proposal is to
602   subclass the device class defined in X.500:

603   printer OBJECT-CLASS ::= {
604   SUBCLASS OF {device}
605   MUST CONTAIN {<list of mandatory attributes>}
606   MAY CONTAIN {<list of optional attributes>}
607
608
609   3. IPP Operations

610   This section introduces the IPP operations. Since IPP specifies the
611   use of HTTP as the underlying communication protocol, the mapping of
612   IPP operations on top of HTTP methods is also shown.

613   2.4 HTTP Overview

614   IPP is based on the existing HTTP standard. IPP is a lightweight
615   application-level protocol designed with the Internet in mind. It is
616   a generic, stateless, object-oriented protocol which can be used for
617   any task through extension of its request methods (commands).

618   HTTP allows an open ended set of methods to be used to indicate the
619   purpose of a request. It builds on the discipline of reference
620   provided by the Uniform Resource Location (URL) and message formats
621   similar to those used by Internet Mail and the Multipurpose Internet
622   Mail Extensions (MIME).

623   HTTP is based on a request response paradigm. A requesting program (a
624   client) establishes a connection with a receiving program (a server)
625   and sends a request to the server in the form of a request method, a
626   URL, and protocol version, followed by a MIME-like message containing

627   request modifiers, client information, and possibly print data. The
628   server responds with a status line, including its protocol version,
629   and a success or failure code, followed by a MIME like message
630   containing server information, entity meta information, and possibly
631   some content.

632   Current practice requires that the connection be established by the
633   client prior to each request and closed by the server after sending
634   the response. Both clients and servers shall be capable of handling
635   cases where either party closes the connection prematurely, due to
636   user action, automated time out, or program failure.

637   2.4 IPP Operation Encoding

638   IPP messages consist of requests from client to server and responses
639   from server to client.

640   IPP MESSAGE = Request | Response
641
642   Requests and responses use the generic message format of RFC 822 for
643   transferring entities. Both messages may include optional header
644   fields and an entity body. The entity body is separated from the
645   headers by a null line (a line with nothing preceding the CRLF).

646   Request = Request-line
647   * (General-Header
648 | Request-Header
649 | Entity Header)
650   CRLF
651   [Entity-Body]
652
653   Response = Status-line
654   * (General Header
655 | Request Header
656 | Entity-Header)
657   CRLF
658   [Entity-Body]
659
660   All IPP headers conform to the syntax

661   IPP-Header = field-name ":" [field-value] CRLF.
662
663   IPP/1.0 defines the octet sequence CRLF as the end-of-line marker for
664   all protocol elements except the entity body.

665   Note that HTTP 1.1 defines a slightly different syntax, allowing for
666   dynamically generated messages to be transmitted. This would be
667   required for cases such as PC driver generated Print Operations.
668   HTTP 1.1 defines a message header which specifies a transfer encoding
669   called "chunks".

670   IPP messages are contained within HTTP methods. The HTTP POST method
671   is used for the Print operation and the Cancel Job operation. The
672   HTTP GET method is used for the Get Attributes operation and the Get
673   Jobs operation (section 5.4).

674 2.3.1 HTTP Request-Header Fields

675   HTTP request header fields allow the client to pass additional
676   information about the request, and about the client itself, to the
677   server. All header fields are optional and when used it is assumed
678   that IPP would use these headers in a standard way. IPP requests
679   will be completely encapsulated within the entity body of an HTTP
680   request. The HTTP Entity-Header has the form

681
682   HTTP Entity Header =   Content Encoding
683 | Content Length
684 | Content-Type
685 | extension-header
686
687   The Content Length field must always be a valid length, This means
688   that for any Print Operations based on HTTP 1.0, the entire content
689   must be generated before this header can be built. HTTP 1.1 provides
690   the notion of "chunks" which will allow the content to be generated
691   dynamically as the data is sent.

692
693   Content Type will always be "Application/IPP".

694 2.3.1.1 IPP Request-Line

695   The first line of the entity body in an IPP operation is the IPP
696   Request-Line. The Request-Line defines the Operation and the IPP
697   Version.
698
699   IPP-Request-Line = Operation-token IPP/1.0 CRLF
700
701   Operation-token = Print | Cancel-Job |
702   Get Attributes | Get Jobs
703

704 2.3.1 HTTP Response-Header Fields

705   HTTP response fields allow the server to pass additional information
706   about the response back to the client. IPP will use these headers in
707   a standard way. IPP responses will be completely encapsulated within
708   the entity body of an HTTP response.

709   2.3.1.1 IPP Status-Line

710   The first line of the entity body in an IPP response is the IPP
711   Status Line. The status line consists of a protocol version followed
712   by a numeric status code and an associated text message.
713
714   IPP-Status-Line = IPP/1.0 Status-Code Reason-Phrase CRLF

715   2.4 The Print Job

716   In section 5.4.1, the Print Operation is described. In order to
717   understand that operation better, we first present the notion of a
718   Print Job. The entity body of a print operation request will contain
719   a Print Job, as defined below. The headers defined here are IPP
720   headers, but follow the same syntax as the basic HTTP headers.

721
722   Print Job = Print Job Object Header ;section (5.3.1)
723   [Job-Attributes]   ;section (5.3.4)
724   *(Documents)
725
726   Document =   Document Header   ;section (5.3.2)
727   [Document attributes] ;section (5.3.5)
728   [Content-Header   ;section (5.3.3)
729   content]
730
731   2.3.1 Print Job Object Header

732   Print Job Object Header = Content Encoding
733 | Content-Length
734 | Content-Type
735 | extension-header
736
737   Content Type is always "IPP Print Object". Other header fields are as
738   defined for HTTP 1.0.

739   2.3.1 Document Header

740   The document header allows the insertion of multiple documents within
741   a job. At this point only a limited number of document attributes are
742   defined. However, this structure allows the addition of other
743   attributes which can be specified on a document boundary.

744   Document-Header = Content-Encoding
745 | Content-Length
746 | Content Type
747 | extension header
748
749   Content type is always "IPP Document". Other header fields are as
750   defined in HTTP 1.0.

751 2.3.1 Document-Content Header

752   The document-content-header provides additional meta-information
753   about the document. The document content header is an optional field
754   and would not be present if the document was pointed to by a document
755   URL attribute. It is composed of a number of document header fields
756   as follows:

757   Document-Content-Header =   Content-Encoding
758 | Content Length
759 | Content Type
760 | extension-header
761
762   Content-Type is defined as :

763   Content-Type = Data Stream Format "/" Version
764
765   Thus, for example, if the document to be printed was a Postscript
766   Level 2 document, the Content-Type would be specified as:

767   Content-Type: Postscript/2.0
768
769   Other header fields are as defined by HTTP 1.0.

770 2.3.1 Job Attributes

771   Job attributes are defined in section 6.2. Attributes will always be
772   sent as

773   Job Attribute = Attr-name ":" Attr-value   CRLF
774
775   Attr-value = 1#Value
776
777   In the above example, "1#Value" means one or more "," separated
778   values.

779 2.3.1 Document Attributes

780   Document attributes are defined in section 6.2.11. The syntax for a
781   document attribute is

782   Document-Attribute = Attr-Name ":" Attr-Value   CRLF
783
784   Attr Value = 1#Value
785
786   In the above example, "1#Value" means one or more "," separated
787   values.

788 2.4 Operation Semantics

789   In this section the four IPP operations are described in terms of
790   their contents and semantics.

791   2.3.1 Print Operation

792   When an end user submits a job, the client submits a Print Request
793   and receives a Print Response.

794   Note that the Printer name is not needed since it is the target of
795   the entire operation. A Print Job contains the information needed by
796   the Printer object to print a document or set of documents. When the
797   print operation is invoked, the Entity-Body in the HTTP request
798   includes an IPP Print Job. The concrete syntax of the Print Job is
799   defined in section 5.3.

800   Each Printer object has an associated Job Template object assigned by
801   the Administrator. When accepting a Print operation, the Printer
802   shall use the corresponding value of an attribute from the Printer's
803   Job Template as the default value for any job attribute that the
804   submitting client omits from the Print operation.

805   If neither the client nor the Printer's Job Template supplies a value
806   for a job attribute, then the output device shall supply its own
807   default value for that job attribute, if necessary, in order to
808   produce output.

809
810   2.3.1.1 Print Request

811   The following abstract data types are part of the Print Request:

Job and Document Attributes	A set of Job object and Document attributes as defined in section 6.2
Requested Attributes	A set of attributes without values in whose values the requester is interested.
Document Contents	Document content is optional and shall not be included when a URL is provided in the document-URL attribute which points to the content.

812
813
814   2.3.1.1 Print Response

815   The following abstract data types are part of the Print Response:

816

Job Identifier	A URL Used for all other operations on this Job.
Job Status	Current job state
Printer State	Printer-state

	Result	The requested attributes with their current
	Attributes	values, if the requester supplied any Requested
		Attributes
	Message	Optional message
	Errors	Optional Error Information

817
818
819 ## 2.3.1 Cancel Job Operation

820 This operation allows a user to cancel one specific Print Job any
821 time after the print job has been established on the Printer Object.
822 Some pages may be printed before a job is terminated if printing has
823 already started when the Cancel Job operation is received. Only the
824 end-user who is also the job originator (job-originator Job
825 attribute) can cancel the job.

826 The Cancel HTTP request will be sent to the URL identifying the job
827 to be canceled.

828 ### 2.3.1.1 Cancel Job Request

829 The following abstract data types are part of the Cancel Job Request:

830

	Message	Optional message to the operator.
	job retention period	The number (cardinal) of minutes that that job is to be retained after the job has been canceled. This parameter updates the value of the job-retention-period that may have been submitted by the submitter in the Print operation.

831
832 ### 2.3.1.1 Cancel Job Response

833 The following abstract data types are part of the Cancel Job
834 Response:
835

	Job Status	Optional Job status information
	Errors	Optional Error Information

836

837 ## 2.3.1 Get Attributes Operation

838 This operation allows an end user to obtain information from the
839 Print object concerning jobs, printers, and print queues, based on
840 ISO 10175. The entity-body of the Get Attributes operation contains

841   the set of attributes that the requester is interested in. The
842   requester should not supply values in the Requested Attributes input
843   parameter; the Printer shall ignore the values of any supplied by the
844   requester. The attribute list is returned in the response with the
845   appropriate attribute values filled in. If no attribute list is
846   supplied, then all attributes defined for that object are returned.

847   2.3.1.1 Get-Attributes Request

848   The following abstract data types are part of the Get Attributes
849   Request:

         Selector   Job-Identifier (URL) or
                             Printer URL or
                             Job Template URL

         Requested   A set of attributes without values in whose
         Attributes   values the requester is interested

850
851   2.3.1.1 Get Attributes Response

852   The following abstract data types are part of the Get Attributes
853   Response:

         Result   The requested attributes of the object with
         Attributes   their current values, if the requester supplied
                             any Requested Attributes

         Errors   Optional error information

854

855   2.3.1 Get Jobs Operation

856   This operation allows a client to retrieve a list of print jobs
857   belonging to the target Printer object. A list of attributes the
858   client is interested in seeing may be appended to the request. If no
859   attributes are asked for the default set of job-name and total-job-
860   octets is returned for each job along with the job identifier. Jobs
861   will be returned in the order in which they are scheduled to print.

862   2.3.1.1 Get-Jobs Request

863   The following abstract data types are part of the Get Jobs Request:
864
         selector   Indicates which jobs the requester seeks. The
                                values are type2Enum (see section 6). Standard
                                values are: "
                                all-jobs" - including completed jobs

"pending" – all jobs which are pending and
processing

"my-jobs" – my jobs that are pending or
processing

Requested   A set of attributes without values in whose
Attributes   values the requester is interested.

865
866
867   2.3.1.1 Get Jobs Response

868   The following abstract data types are part of the Get Jobs Response:
869
Jobs   A list of Job URLs is returned. The list is in
"scheduled" order. The job-identifier attribute
shall be returned as the first attribute of each
job to mark the beginning of the set of
attributes for the next job.

Result   In addition to the job-identifier attribute
Attributes   which is always returned, either the Requested
Attributes are returned or the following
attributes by default, if the requester did not
supply any Requested Attributes: job-total-
octets and number-of-intervening-job. This last
attribute is necessary since an end user may
request just their own jobs and they need some
relative position indicator if there are other
jobs interspersed in the waiting list which are
not returned in the response or cannot be
because of site security policy restrictions.

Errors   Optional Error Information

870
871   3. Object Attributes

872   This section describes the attributes, syntaxes, and values that are
873   part of IPP. The sections below show the objects and their associated
874   attributes which are included within the scope of this protocol. The
875   text in these sections has been heavily influenced by the ISO/IEC
876   10175 DPA (Final, June 1996).

877   2.4 Attribute Syntaxes

878   The syntax for attribute values is specified using the notation of
879   RFC 822.

880   The special syntax State is used to form other syntaxes for xxx-
881   supported attributes of the Printer object that indicate job
882   attributes that the Printer supports. Such support may include
883   operator intervention, delivery of an order that the provider has
884   previously placed, or may require that the provider place a special
885   order. The syntax for State is itself a type2Enum. The standard
886   values are: [":not-ready" / ":on-order" / ":special-order"]

887   An attribute value with an empty State means that the indicated value
888   is ready to be used without human intervention.

889   An attribute value with a ":not-ready" State means that operator
890   intervention is required.

891   An attribute value with a ":on-order" State means that the provider
892   has placed an order for the indicated value and that the operator
893   must wait until the resource is delivered before the job can be
894   printed. However, an end-user may submit a job that requires such a
895   resource and the Printer shall accept such a job.

896   An attribute value with a ":special-order" State means that the
897   provider shall make a special order for the resource, when a job is
898   submitted that needs such a resource. However, an end user may
899   submit a job that requires such a resource and the Printer shall
900   accept such a job.

901   For example, the media-supported printer attribute might contain the
902   following values:

903   media supported = na letter white, na letter transparent,
904   b:not-ready
905
906   Meaning that na-letter-white and na-letter-transparent are loaded
907   into the two trays of the output device and that b is supported, but
908   requires the operator to change the trays.

909   The sections below reference the following syntax items:

        string   arbitrary ASCII strings, no control
                                characters, except <SPACE>.
        StringPair   string ":" string
        stringState   string State
        name   arbitrary ASCII strings, no control
                                characters, and no <SPACE>
                                characters.
        Url   Universal Resource Locator
        dateTime   date and time in RFC 822 format
        deltaTime   [hours ":"] minutes
        cardinal   0 .. n represented as ASCII digits
        type1Enum   standard names, must revise the IPP
                                standard to add a new name. No
                                private names are allowed.

type2Enum	standard names, but an implementor can, at any time, add new values by proposing them to the PWG for registration (or an IANA appointed registry advisor after the PWG is no longer certified) where they are reviewed for approvoal.. IANA keeps the registry. Implementors can support private (un registered) with a suitable distinguishing prefix, such as -xxx- where xxx is the company name registered with IANA for use in domain names.
Type3Enum	standard names, but an implementor can add new values by submitting a registration request directly to IANA, no PWG or IANA-appointed registry advisor review is required. Implementors can support private (un-registered) names with a suitable distinguishing prefix, such as -xxx- where xxx is the company name registered with IANA for use in domain names.
type2EnumState	type2Enum State
type3EnumState	type3Enum State
boolean	tokens: yes, y, true, or t and no, n, false, or f.
positiveInteger	1 .. n represented as ASCII digits
positiveIntegerCross	positiveInteger ["x" positiveInteger]
positiveIntegerCrossState	positiveIntegerCross State
positiveIntegerRange	positiveInteger ":" positiveInteger
positiveIntegerUnits	positiveInteger units
positiveIntegerState	positiveInteger State
units	"ppm"	"ipm"	"spm"	"cps"	"lpm"
type3Locale	type3Country ":" type3Language ":" type3CodeSet
type3Country	type3Enum - Standard values are the two character country codes from ISO 639.
type3Language	type3Enum - Standard values are the two-character language codes from ISO 3166.
type3CodeSet	type3Enum - Standard values are from the IANA Code Set registry.
type2Format	name ["/" version]
version	name
type3LocaleState	type3Locale State

910

911   Also, the following conventions (from RFC 822) are used:

       "1#" in front of a data   means one or more values separated
       syntax   by ",".

912

913   NOTE - For consistency, no Job (or Job Template) or Printer attribute
914   has the syntax # meaning zero or more values separated by ",".
915   Instead, a distinguished value, such as "none", is used to indicate
916   no value. For the Printer Object, the omission of the attribute
917   entirely, is also used to indicate no value. In all such cases for
918   the Printer object where a conforming implementation may omit the
919   attribute all together, an explicit sentence indicates the meaning of
920   the Printer attribute when the attribute is unspecified.

921   2.4 Job Attributes

922   A job object contains a set of job attributes and one or more
923   documents. A client shall create a job and send it to a server using
924   the Print operation. When accepting a Print operation, the Printer
925   shall use the corresponding value of an attribute from the Printer's
926   Job Template as the default value for any job attribute that the
927   submitting client omits from the Print operation.

928   A client may use a job template associated with the selected printer
929   in order to initialize the job. To do so, the client uses the Get-
930   Attributes operation to get the URLs of the Printer's Job Templates.
931   Then the client may get the default attributes from the Printer's
932   default Job Template in order to initialize a display to the end user
933   with the Printer's defaults. See the printer-job-templates Printer
934   attribute. However, a client need not access the Job Template in
935   order to issue a Print operation; the client can depend on the
936   Printer to supply the default job object attribute values as part of
937   the Print operation.

938   Each section heading below contains the name of an attribute and its
939   syntax in parentheses using the rules of RFC 822.

940   2.3.1 Job Informational Attributes (Set by a Client/End User)

941   The client may specify these attributes in the Print operation to
942   provide information to identify a print job.

943   The client may also specify these attributes in the operations: Get-
944   Attributes, and Get-Jobs.

945   2.3.1.1 job-name (string)

946   This attribute supplies a human readable string for naming the print
947   job.

This attribute is intended to be printed on a start sheet, returned
in a Get-Jobs result, or used in notification messages.

If the client does not specify this attribute, a Printer shall set it
to the value of the document name attribute of the first document in
the job.

2.3.1 Job Informational Attributes (Set by a Printer)

The Print shall add all of these attributes to a job to provide
information to identify a print job.

The client may specify these attributes in the operations: Get-
Attributes and Get-Jobs, but not in Print.

2.3.1.1 job-identifier (url)

This attribute provides the job-identifier for this job on the
Printer. The Printer shall generate a job identifier value as a URL.

The value of the job identifier attribute shall be returned by the
Printer as part of the PrintResult in the Print operation.

2.3.1.1 job-originator (name)

This attribute specifies the name of the person submitting the print
job. The Printer shall set this attribute to the most authentic name
that it can obtain from the client. The operation user name attribute
is intended to be a source of the most authentic name.

2.3.1.1 job-originating-host (name)

This attribute identifies the originating host of the job. The
Printer shall set this attribute to the value of the operation host
name which is intended to be the most authentic host name of the
client.

2.3.1.1 job-locale (type3Locale)

This attribute identifies the locale of the job, i.e, the country,
language, and coded character set. The Printer sets this attribute
from the value of the operation locale.

The Printer shall use this attribute to determine the locale for
notification messages that it sends.

Issue: Is there a more standard syntax for locale?

980   2.3.1   Job Status Attributes (Set by Printer)

981   The Printer shall add these attributes to a job when a client submits
982   a job, and the Printer shall assign appropriate values to each such
983   job status attribute.

984   The Printer uses these attributes to specify the job status before,
985   during and after the processing of the print-job by the Printer.

986   The client may specify job-status attributes in: Get-Attributes and
987   Get-Jobs, but not Print.

988   2.3.1.1 current job state (type1Enum)

989   This attribute identifies the current state of the job. Standard
990   values are:

    Unknown   The job state is not known, or is
                     indeterminate.
    held   The job is waiting to be released for
                     scheduling for any number of reasons as
                     specified by the value of the job's job-state-
                     reasons attribute.
    pending   The job is waiting to start processing on a
                     printer.
    processing   The server is processing the job, or has made
                     the job ready for printing, but the output
                     device is not yet printing it, either because
                     the job hasn't reached the output device or
                     because the job is queued in the output device
                     or some other spooler, awaiting the output
                     device to print it.

                     Or

                     The server has completed processing the job
                     and the output device is currently printing
                     the job. That is, an output device is either
                     printing pages of the job, or failing in its
                     attempt to print pages of the job because of
                     some wait state, such as, start-wait, end-
                     wait, needs-attention, etc. The complete job
                     state includes the detailed status represented
                     in the printer's printer state attribute.
    paused   The job has been paused
    Interrupted   The job has been interrupted by some
                     intervening job, and shall resume processing
                     automatically once the intervening job has
                     completed.

Terminating   The job has been canceled by a Cancel-Job
               request or aborted by the server and is in the
               process of terminating. The job's job state
               reasons attribute contains the reasons that
               the job is being terminated.
Retained   The job is being retained at the server as a
               result of the job's job-retention-period being
               non zero. The job has (1) completed
               successfully or with warnings or errors, (2)
               been aborted while printing by the server, or
               (3) been canceled by the Cancel-Job request
               before or during processing. The job's job-
               state reasons attribute contains the reasons
               that the job has been retained.
               While in the retained state, all of the job's
               document data (and resources, if any) shall be
               retained by the server; thus a job in the
               retained state could be reprinted, using some
               means outside the scope of IPP V1.0.
Completed   The job has:

                   (1) completed successfully or with
               warnings or errors,
                   (2) been aborted by the server while
               printing, or
                   (3) been canceled by the Cancel-Job
               request,

               AND the job's:
                   (1) job-retention-period was zero or has
               expired, or
                   (2) job discard time has arrived.
               The job's job-state-reasons attribute contains
               the reason(s) that the job has been completed.
               While in the completed state, a job's document
               data (and resources if any) need not be
               retained by the server; thus a job in the
               completed state could not be reprinted. The
               length of time that a job may be in this
               state, before transitioning to unknown, is
               implementation dependent. However, servers
               that implement the completed job state shall
               retain, as a minimum, the following attributes
               for any job in the completed state: job-
               identifier, job-originator, job-name, current-
               job state, output device assigned, and job-
               state reasons.

991

992 The IPP protocol supports all values for job states, but Printers
993 need only support those states which are appropriate for the
994 particular implementation.

995 2.3.1.1 output device assigned (name)

996 This attribute identifies the Output Device to which the Printer has
997 assigned this job.

998 If an Output Device implements a Printer, the Printer need not set
999 this attribute.

1000 If a Print Server implements a Printer, the value shall be empty
1001 until the Printer assigns an Output Device to the job.

1002 The value of the job's output-device-assigned attribute shall remain
1003 after the job has completed, so that end users can determine the
1004 Output Device on which the job was printed.

1005 2.3.1.1 submission time (dateTime)

1006 This attribute indicates the time at which this job was accepted by
1007 the Printer. If the Printer does not support the notion of time, the
1008 attribute need not be stored as part of the job object.

1009 2.3.1.1 number of intervening jobs (cardinal)

1010 This attribute indicates the number of jobs that are "ahead" of this
1011 job in the current scheduled order. For efficiency, it is only
1012 necessary to calculate this value when an operation if performed that
1013 requests this attribute.

1014 NOTE   This attribute is necessary since an end user may request just
1015 their own jobs and they need some relative position indicator if
1016 there are other jobs interspersed in the waiting list which are not
1017 returned in the response or cannot be because of site security policy
1018 restrictions.

1019 2.3.1.1 job message from operator (string)

1020 This attribute provides a message from an operator, system
1021 administrator or "intelligent" process to indicate to the end user
1022 the reasons for modification or other management action taken on a
1023 job.

1024 2.3.1.1 completion time (dateTime)

1025 This attribute indicates the time at which this job completed. This
1026 time is useful for jobs which are retained after printing. If the
1027 Printer does not support the notion of time, the attribute is not
1028 stored as part of the Job object.

1029   2.3.1.1 job-state-reasons (1#type2Enum)

1030   This attribute identifies the reason or reasons that the job is in
1031   the state that it is in (e.g., held, terminating, retained,
1032   completed, etc.). The printer shall indicate the particular
1033   reason(s) by setting the value of the job-state-reasons attribute.

1034   The following standard values are defined:

none	There are not reasons associated with the job's current state.
documents needed	The complete job has been accepted by the server, but the server is waiting for its files to be transferred before the job can be scheduled to be printed.
job hold set	The value of the job's job hold attribute is TRUE.
job-print-after-specified	The value of the job's job-print-after or print-off-peak attributes have specified a time specification that has not yet occurred.
Required resources not-ready	At least one of the resources needed by the job, such as media, fonts, resource objects, etc., is not ready on any of the physical printer's for which the job is a candidate.
Successful completion	The job completed successfully.
Completed-with-warnings	The job completed with warnings.
Completed-with-errors	The job completed with errors (and possibly warnings too).
Cancelled by user	The job was cancelled by the user using the CancelJob request.
Cancelled-by-operator	The job was cancelled by the operator using the CancelJob request.
Aborted by system	The job was aborted by the system.
Logfile pending	The job's logfile is pending file transfer.
Logfile-transferring	The job's logfile is being transferred.

1035

1036   2.3.1.1 impressions completed (cardinal)

1037   This attribute contains the number of impressions that the Printer
1038   has completed printing. If the Printer cannot report this number,
1039   the Printer leaves this attribute unspecified.

1040   2.3.1.1 media-sheets-completed (cardinal)

1041   This attribute contains the number of media sheets that the Printer
1042   has completed printing. If the Printer cannot report this number,
1043   the Printer leaves this attribute unspecified.

1044   2.3.1 Job Sheet Attributes (Set by Client/End User)

1045   The client shall specify these attributes to control the printing of
1046   job sheets.

1047   The client may also specify job sheet attributes in: Get Attributes
1048   and Get-Jobs.

1049   2.3.1.1 job-sheets (type3Enum)

1050   This attribute determines what type of job-sheets the Printer shall
1051   print with the job.

1052   The standard values are: none, and default sheet.

1053   The value "none" means that the Printer shall print no job sheets.
1054   The value "default-sheet" means that the Printer shall print the job
1055   sheets defined by an administrator. If the administrator's policy is
1056   not to support none, the Printer shall use the default sheet value if
1057   the client supplies the "none" value.

1058   NOTE - The effect of this attribute on jobs and documents is
1059   controlled by the files-are-one-document and files-are-interleaved
1060   job attributes.

1061   2.3.1 Notification Attributes (Set by a Client/End User)

1062   The client shall specify these attributes to indicate events that the
1063   client is interested in, along with the notification address and
1064   method for performing the notification.

1065   The client may also specify notification attributes in: Get-
1066   Attributes and Get Jobs.

1067   2.3.1.1 notification events (1#type2Enum)

1068   This attribute specifies the events about which the end user want to
1069   be notified.

1070   Standard values are: none, job-completion, job-problems and printer-
1071   problems.

1072   If this attribute contains the event none, the Printer shall not
1073   notify. This value is useful if an administrator has set up a
1074   notification Printer default but the end user does not want
1075   notification. If the none value and other values are supplied, the
1076   Printer shall ignore the none value.

1077   If this attribute contains the value: job completion, the Printer
1078   shall notify the client when the job containing this attribute

completes with or without errors or is cancelled by the end-user or
the operator.

If this attribute contains the value: job problems, the Printer
shall notify the client when this job has a problem while this job is
printing. Problems include: paper jam and out-of-paper.

If this attribute contains the value: printer-problems, the Printer
shall notify the client when any job, including this job, has a
problem while this job is waiting to print or printing. Problems
include: paper jam and out of paper.

2.3.1.1 notification-address (url)

This address specifies both the address and mechanism for delivery of
notification events to the client. The client specifies this
attribute in the operation notification address attribute which the
Printer in turn uses to set this attribute.

The Printer shall use this attribute as the address for sending
messages to a job submitter when an event occurs that the end user
has registered an interest in or when certain other events occur,
such as Cancel Job.

If the URL has a "mailto:" scheme, then email is used and the rest of
the URL is used as the email address. If the URL has a "http:"
scheme, then an HTTP method is used to add HTML formatted events to
the end of the specified HTML file.

2.3.1 Job Scheduling Attributes (Set by Client/End User)

The client shall specify these attributes to provide the Printer with
information for the scheduling a print-job.

The client may also specify these attributes in: Get-Attributes and
Get-Jobs.

2.3.1.1 job priority (type1Enum)

This attribute specifies a priority for scheduling the print job.
Printers that employ a priority-based scheduling algorithm use this
attribute.

There are three standard values: high, default, and low. Among those
jobs that are ready to print, a Printer shall print all such jobs
with a high priority before printing those with a default or low
priority, and a Printer shall print all such jobs with a default
priority before printing those with a low priority.

1115   If the client does not specify this attribute, the Printer assumes
1116   that the end user places no constraints concerning priority on the
1117   scheduling of the print job, and it has a priority value of default.

1118   An operator can modify a job to have any priority. An end user is
1119   restricted by the value of the maximum-end-user-priority Printer
1120   attribute.

1121   2.3.1.1 job-print-after (dateTime)

1122   This attribute specifies the calendar date and time of day after
1123   which the print job shall become a candidate for printing.

1124   If the value of this attribute is in the future, the Printer shall
1125   set the value of the job's current-job-state to held and add the job-
1126   print-after-specified value to the job's job-state-reasons attribute
1127   and shall not schedule the print job for printing until the specified
1128   date and time has passed. When the specified date and time arrives,
1129   the Printer shall remove the job-print-after-specified value from the
1130   job's job-state-reason attribute and, if no other reasons remain,
1131   shall change the job's current-job-state to pending so that the job
1132   becomes a candidate for being scheduled to print.

1133   If this attribute is unspecified or the value is in the past, the job
1134   shall be a candidate for scheduling immediately.

1135   2.3.1.1 job-print-off-peak (type3Enum)

1136   This attribute specifies the off-peak period during which the print-
1137   job shall become a candidate for printing.

1138   Standard values are: "evening", "night", "weekend", "second shift",
1139   "third-shift".

1140   If this attribute is specified, it contains a value with which an
1141   administrator has associated allowable print times. An administrator
1142   is encouraged to pick names that suggest the type of off peak period.

1143   If the value of this attribute is in the future, the Printer shall
1144   set the value of the job's current-job-state to held and add the job-
1145   print-after-specified value to the job's job-state-reasons attribute
1146   and shall not schedule the print-job for printing until the specified
1147   date and time has passed. When the specified date and time arrives,
1148   the Printer shall remove the job print after specified value from the
1149   job's job-state-reason attribute and, if no other reasons remain,
1150   shall change the job's current-job-state to pending so that the job
1151   becomes a candidate for being scheduled to print.

1152   If this attribute is unspecified, the job shall be a candidate for
1153   scheduling immediately.

1154 2.3.1.1 job-retention-period (deltaTime)

1155   The retention time is expressed in hours and minutes, e.g. 6:00 (6
1156   hours), or 20 (20 minutes).

1157   This attribute specifies the minimum period of time following the
1158   completion of job processing and printing that the server shall keep
1159   job attributes and document data. The Printer may keep these
1160   attributes and data longer than the value of the job-retention-period
1161   attribute.

1162   NOTE   the requester may change this job attribute using the input
1163   parameter to the Cancel-Job operation.

1164 2.3.1 Job Production Attributes (Set by Client/End User)

1165   The client shall specify these attributes to affect the rendering,
1166   production and finishing of the documents in the job. Similar types
1167   of instructions may also be contained in the document to be printed.

1168   If there is a conflict between the value of one of these attributes,
1169   and a corresponding instruction in the document (either implicit or
1170   explicit), the value of the attribute shall take precedence over the
1171   document instruction.

1172   Job Production and Resource Attributes each address a similar set of
1173   features but they have different uses.

1174   A job production attribute provides a client with a way to request
1175   some feature at print time that may not have been embedded within
1176   the document data when the document was created. A job production
1177   attribute also provides a client with a way to override a feature at
1178   print time that was embedded within the document data when the
1179   document was created.

1180   Note: until companies that supply interpreters for PDL's, such as
1181   PostScript and PCL allow a way to specify overrides for internal job
1182   production instructions, a Printer may not be able to implement these
1183   attributes for some PDL's.

1184   A job resource attribute tells a Printer what features the job needs.
1185   A program that translates document data to a Printer's PDL, and/or
1186   merges production attributes into the document data should add job
1187   resource attributes to a job.

1188   For example, a job production attribute medium-select with the value
1189   of "letter" requests that a job be printed on letter paper, but gives
1190   no information about what resources the job needs. For example, a job
1191   resource attribute media used with the values of "letter" and
1192   "ledger" tell a Printer that the job needs letter and ledger paper,
1193   but gives no information about which pages use each medium.

1194   The client may also specify job production-instruction attributes in:
1195   Get-Attributes and GetJobs.

1196   2.3.1.1 medium select (type2Enum)

1197   This attribute identifies the medium that the Printer shall use for
1198   all pages of the document regardless of what media are specified
1199   within the document.

1200   The values for medium include medium-names, medium-sizes, input-trays
1201   and electronic forms so that one attribute specifies the media.

1202   Standard values are (taken from ISO DPA and the Printer MIB):

         default   The default medium for the output
                               device
         iso-a4-white   Specifies the ISO A4 white medium
         iso a4 colored   Specifies the ISO A4 coloured medium
         iso a4 transparent   Specifies the ISO A4 transparent
                               medium
         iso-a3-white   Specifies the ISO A3 white medium
         iso-a3-colored   Specifies the ISO A3 coloured medium
         iso a5 white   Specifies the ISO A5 white medium
         iso a5 colored   Specifies the ISO A5 coloured medium
         iso-b4-white   Specifies the ISO B4 white medium
         iso-b4-colored   Specifies the ISO B4 coloured medium
         iso-b5-white   Specifies the ISO B5 white medium
         iso b5 colored   Specifies the ISO B5 coloured medium
         jis b4 white   Specifies the JIS B4 white medium
         jis-b4-colored   Specifies the JIS B4 coloured medium
         jis-b5-white   Specifies the JIS B5 white medium
         jis-b5-colored   Specifies the JIS B5 coloured medium
1203

1204   The following standard values are defined for North American media:

         na-letter-white   Specifies the North American letter
                               white medium
         na-letter-colored   Specifies the North American letter
                               coloured medium
         na letter   Specifies the North American letter
         transparent   transparent medium
         na-legal-white   Specifies the North American legal
                               white medium
         na legal colored   Specifies the North American legal
                               coloured medium
1205

1206   The following standard values are defined for envelopes:

         iso-b4-envelope   Specifies the ISO B4 envelope medium
         iso b5 envelope   Specifies the ISO B5 envelope medium

iso-c3-envelope	Specifies the ISO C3 envelope medium
iso-c4-envelope	Specifies the ISO C4 envelope medium
iso-c5-envelope	Specifies the ISO C5 envelope medium
iso-c6-envelope	Specifies the ISO C6 envelope medium
iso-designated-long-envelope	Specifies the ISO Designated Long envelope medium
na-10x13-envelope	Specifies the North American 10x13 envelope medium
na-9x12-envelope	Specifies the North American 9x12 envelope medium
monarch-envelope	Specifies the Monarch envelope
na-number-10-envelope	Specifies the North American number 10 business envelope medium
na-7x9-envelope	Specifies the North American 7x9 inch envelope
na-9x11-envelope	Specifies the North American 9x11 inch envelope
na-10x14-envelope	Specifies the North American 10x14 inch envelope
na-number-9-envelope	Specifies the North American number 9 business envelope
na-6x9-envelope	Specifies the North American 6x9 inch envelope
na-10x15-envelope	Specifies the North American 10x15 inch envelope

1207

1208   The following standard values are defined for the less commonly used
1209   media (white only):

executive-white	Specifies the white executive medium
folio-white	Specifies the folio white medium
invoice-white	Specifies the white invoice medium
ledger-white	Specifies the white ledger medium
quarto-white	Specified the white quarto medium
iso-a0-white	Specifies the ISO A0 white medium
iso-a1-white	Specifies the ISO A1 white medium
iso-a2-white	Specifies the ISO A2 white medium
iso-a6-white	Specifies the ISO A6 white medium
iso-a7-white	Specifies the ISO A7 white medium
iso-a8-white	Specifies the ISO A8 white medium
iso-a9-white	Specifies the ISO A9 white medium
iso-10-white	Specifies the ISO A10 white medium
iso-b0-white	Specifies the ISO B0 white medium
iso-b1-white	Specifies the ISO B1 white medium
iso-b2-white	Specifies the ISO B2 white medium
iso-b3-white	Specifies the ISO B3 white medium
iso-b6-white	Specifies the ISO B6 white medium
iso-b7-white	Specifies the ISO B7 white medium
iso-b8-white	Specifies the ISO B8 white medium
iso-b9-white	Specifies the ISO B9 white medium
iso-b10-white	Specifies the ISO B10 white medium
jis-b0-white	Specifies the JIS B0 white medium

```
            jis-b1-white          Specifies the JIS B1 white medium
            jis-b2-white          Specifies the JIS B2 white medium
            jis-b3-white          Specifies the JIS B3 white medium
            jis-b6-white          Specifies the JIS B6 white medium
            jis-b7-white          Specifies the JIS B7 white medium
            jis-b8-white          Specifies the JIS B8 white medium
            jis-b9-white          Specifies the JIS B9 white medium
            jis-b10-white         Specifies the JIS B10 white medium
1210


1211    The following standard values are defined for engineering media:

            a                     Specifies the engineering A size
                                  medium
            b                     Specifies the engineering B size
                                  medium
            c                     Specifies the engineering C size
                                  medium
            d                     Specifies the engineering D size
                                  medium
            e                     Specifies the engineering E size
                                  medium
1212


1213    The following standard values are defined for input-trays (from ISO
1214    DPA and the Printer MIB):

            top          The top input tray in the printer.
            middle       The middle input tray in the printer.
            bottom       The bottom input tray in the printer.
            envelope     The envelope input tray in the printer.
            manual       The manual feed input tray in the printer.
            large        The large capacity input tray in the printer.
            capacity
            Main         The main input tray
            side         The side input tray
1215


1216    The following standard values are defined for media sizes (from ISO
1217    DPA):

            iso-a0     Specifies the ISO A0 size: 841 mm by 1189 mm as
                       defined in ISO 216
            iso-a1     Specifies the ISO A1 size: 594 mm by 841 mm as
                       defined in ISO 216
            iso-a2     Specifies the ISO A2 size: 420 mm by 594 mm as
                       defined in ISO 216
            iso-a3     Specifies the ISO A3 size: 297 mm by 420 mm as
                       defined in ISO 216
            iso-a4     Specifies the ISO A4 size: 210 mm by 297 mm as
                       defined in ISO 216
```

          iso-a5   Specifies the ISO A5 size: 148 mm by 210 mm as
                     defined in ISO 216
          iso-a6   Specifies the ISO A6 size: 105 mm by 148 mm as
                     defined in ISO 216
          iso-a7   Specifies the ISO A7 size: 74 mm by 105 mm as
                     defined in ISO 216
          iso-a8   Specifies the ISO A8 size: 52 mm by 74 mm as
                     defined in ISO 216
          iso-a9   Specifies the ISO A9 size: 37 mm by 52 mm as
                     defined in ISO 216
          iso-a10   Specifies the ISO A10 size: 26 mm by 37 mm as
                     defined in ISO 216

1218

          iso-b0   Specifies the ISO B0 size: 1000 mm by 1414 mm as
                     defined in ISO 216
          iso-b1   Specifies the ISO B1 size: 707 mm by 1000 mm as
                     defined in ISO 216
          iso-b2   Specifies the ISO B2 size: 500 mm by 707 mm as
                     defined in ISO 216
          iso-b3   Specifies the ISO B3 size: 353 mm by 500 mm as
                     defined in ISO 216
          iso-b4   Specifies the ISO B4 size: 250 mm by 353 mm as
                     defined in ISO 216
          iso-b5   Specifies the ISO B5 size: 176 mm by 250 mm as
                     defined in ISO 216
          iso-b6   Specifies the ISO B6 size: 125 mm by 176 mm as
                     defined in ISO 216
          iso-b7   Specifies the ISO B7 size: 88 mm by 125 mm as
                     defined in ISO 216
          iso-b8   Specifies the ISO B8 size: 62 mm by 88 mm as
                     defined in ISO 216
          iso-b9   Specifies the ISO B9 size: 44 mm by 62 mm as
                     defined in ISO 216
          iso-b10   Specifies the ISO B10 size: 31 mm by 44 mm as
                     defined in ISO 216

1219

          na-letter   Specifies the North American letter size:
                        8.5 inches by 11 inches
          na-legal   Specifies the North American legal size:
                        8.5 inches by 14 inches
          executive   Specifies the executive size (7.25 X 10.5
                        in)
          folio   Specifies the folio size (8.5 X 13 in)
          invoice   Specifies the invoice size (5.5 X 8.5 in)
          ledger   Specifies the ledger size (11 X 17 in)
          quarto   Specifies the quarto size (8.5 X 10.83 in)

1220

          iso-c3   Specifies the ISO C3 size: 324 mm by 458 mm
                        as defined in ISO 269
          iso-c4   Specifies the ISO C4 size: 229 mm by 324 mm
                        as defined in ISO 269
          iso-c5   Specifies the ISO C5 size: 162 mm by 229 mm
                        as defined in ISO 269

iso-c6   Specifies the ISO C6 size: 114 mm by 162 mm
                  as defined in ISO 269
iso-   Specifies the ISO Designated Long size: 110
designated-   mm by 220 mm as defined in ISO 269
long

1221

na-10x13-envelope Specifies the North American 10x13
                   size: 10 inches by 13 inches
na-9x12 envelope   Specifies the North American 9x12 size:
                   9 inches by 12 inches
na-number-10-   Specifies the North American number 10
envelope   business envelope size: 4.125 inches by
                   9.5 inches
na-7x9 envelope   Specifies the North American 7x9 inch
                   envelope size
na-9x11-envelope   Specifies the North American 9x11 inch
                   envelope size
na-10x14 envelope Specifies the North American 10x14 inch
                   envelope size
na-number-9-   Specifies the North American number 9
envelope   business envelope size
na-6x9-envelope   Specifies the North American 6x9
                   envelope size
na-10x15 envelope Specifies the North American 10x15
                   envelope size
monarch-envelope   Specifies the Monarch envelope size
                   (3.87 x 7.5 in)

1222

a   Specifies the engineering A size:
                   8.5 inches by 11 inches
b   Specifies the engineering B size:
                   11 inches by 17 inches
c   Specifies the engineering C size:
                   17 inches by 22 inches
d   Specifies the engineering D size:
                   22 inches by 34 inches
e   Specifies the engineering E size:
                   34 inches by 44 inches

1223

jis-b0   Specifies the JIS B0 size: 1030mm x 1456mm
jis-b1   Specifies the JIS B1 size: 728mm x 1030mm
jis-b2   Specifies the JIS B2 size: 515mm x 728mm
jis-b3   Specifies the JIS B3 size: 364mm x 515mm
jis-b4   Specifies the JIS B4 size: 257mm x 364mm
jis-b5   Specifies the JIS B5 size: 182mm x 257mm
jis-b6   Specifies the JIS B6 size: 128mm x 182mm
jis-b7   Specifies the JIS B7 size: 91mm x 128mm
jis-b8   Specifies the JIS B8 size: 64mm x 91mm
jis-b9   Specifies the JIS B9 size: 45mm x 64mm
jis-b10   Specifies the JIS B10 size: 32mm x 45mm

1224

1225 2.3.1.1 finishing (type2Enum)

1226 This attribute identifies the finishing operation that the Printer
1227 should apply to each copy of the printed document.

1228 NOTE The effect of this atttribute on jobs and documents is
1229 controlled by the files-are-one-document and files-are-interleaved
1230 job attributes.

1231 Standard values for this attribute are:

      none   Perform no finishing.
      staple   This indicates that staples are to be used
                        to bind the document. The exact number and
                        placement of the staples is site-defined;
                        other finishing object attributes may be
                        included to provide this information.
      staple-top   This indicates that one or more staples
      left   should be placed on the top left corner of
                        the document
      staple-bottom-   This indicates that one or more staples
      left   should be placed on the bottom left corner
                        of the document
      staple-top-   This indicates that one or more staples
      right   should be placed on the top right corner
                        of the document
      staple bottom-   This indicates that one or more staples
      right   should be placed on the bottom right
                        corner of the document
      saddle-stitch   This indicates that one or more staples
                        (wire stitches) are to be used to bind the
                        document along the middle fold. The exact
                        number and placement of the stitches is
                        site-defined.
      edge-stitch   This indicates that one or more staples
                        (wire stitches) are to be used to bind the
                        document along one edge. The exact number
                        and placement of the staples is site-
                        defined.
      punch   This indicates that holes are required in
                        the finished document. The exact number
                        and placement of the holes is site defined
                        The punch specification may be satisfied
                        (in a site- and implementation-specific
                        manner) either by drilling/punching, or by
                        substituting predrilled media.
      cover   This value is specified when it is desired
                        to select a non printed (or pre printed)
                        cover for the document. This does not
                        supplant the specification of a printed
                        cover (on cover stock medium) by the
                        document itself.

                bind   This indicates that a binding is to be
                                    applied to the document; the type and
                                    placement of the binding is site defined.
1232

1233   2.3.1.1 number-up (type3Enum)

1234   This attribute specifies the number of source page-images to impose
1235   upon a single side of an instance of a selected medium.

1236   In general, only certain numeric values are valid for this attribute
1237   and the value "none", depending upon the Printer implementation to
1238   which the print-request is directed. Standard values are: "none",
1239   "1", "2", "4".

1240   This attribute primarily controls the translation, scaling and
1241   rotation of page images, but a site may choose to add embellishments,
1242   such as borders to each logical page. The value "none" shall not
1243   include any embellishments and shall place one logical page on a
1244   single side of an instance of the selected medium without any
1245   translation, scaling, or rotation.

1246   2.3.1.1 sides (type2Enum)

1247   This attribute specifies how source page images are to be imposed
1248   upon the sides of an instance of a selected medium.

1249   The standard values are: 1-sided, 2-sided-long-edge, 2-sided-short-
1250   edge.

1251   1-sided imposes each consecutive source page image upon the same side
1252   of consecutive media sheets.

1253   2-sided-long-edge imposes each consecutive pair of source page-image
1254   upon front and back sides of consecutive media sheets, such that the
1255   orientation of each pair of source-pages on the medium would be
1256   correct for the reader as if for binding on the long edge. This
1257   imposition is sometimes called "duplex".

1258   2-sided-short-edge imposes each consecutive pair of source page-image
1259   upon front and back sides of consecutive media sheets, such that the
1260   orientation of each pair of source-pages on the medium would be
1261   correct for the reader as if for binding on the short edge. This
1262   imposition is sometimes called "tumble" or "head to toe".

1263   Issue: How does sides interact with portrait vs. landscape and
1264   reverse-landscape documents?

2.3.1.1 copies (positiveInteger)

   This attribute specifies the number of copies of the job to be
   printed. If this attribute is unspecified by both the client and the
   Printer's Job Template, its default value shall be 1.

   NOTE - The effect of this atttribute on jobs and documents is
   controlled by the files-are-one-document and files-are-interleaved
   job attributes.

2.3.1.1 printer resolution select (positiveIntegerCross)

   This attribute specifies the resolution that the Printer should use.

   The syntax allows a single integer to specify the resolution or a
   pair of integers to specify the resolution when the x and y
   dimensions differ. When two integers are specified, the first is in
   the x direction, ie., in the direction of the shortest dimension of
   the medium, so that the value is independent of whether the printer
   feeds long edge or short edge first.

2.3.1.1 print-quality (type2Enum)

   This attribute specifies the print quality that the Printer should
   use.

   The standard values are:

      draft   Lowest quality available on the printer
      normal   Normal or intermediate quality on the printer
      high   Highest quality available on the printer

2.3.1.1 page-select (positiveIntegerRange)

   This attribute specifies the pages in the document that the Printer
   shall use. This attribute is unlikely to be useful for jobs with more
   than one document or in Job Templates. If this attribute is
   unspecified, then the Printer shall print all pages in a document.

2.3.1.1 files-are-one-document (boolean)

   This attribute is relevant only if a job consists of two or more
   documents. It controls finishing operations, job-sheet placement, and
   the order of documents when the copies attribute exceeds 1.

   If the files for the job are a and b and this attribute is true, then
   files a and b are treated as a single document for finishing
   operations. Also, there will be no slip sheets between files a and b.
   If more than one copy is made, the ordering must be a, b, a, b, ...
   The attribute files are interleaved is ignored.

1303   If the files for the job are a and b and this attribute is false or
1304   unspecified by both the client and the Printer's Job Template, then
1305   each file is treated as a single document for finishing operations.
1306   Also, a client may specify that a slip sheet be between files a and
1307   b. If more than one copy is made, and the attribute files-are-
1308   interleaved false or unspecified, the ordering is a, a, b, b, ... If
1309   more than one copy is made, and the attribute files-are-interleaved
1310   true, the ordering is a, b, a, b, ...

1311   2.3.1.1 files are interleaved (boolean)

1312   This attribute is used in conjunction with files-are-one-document
1313   (q.v.).

1314

1315   2.3.1 Attributes for Conversion of Text and HTML Files (Set by
1316   Client/End User)

1317   The client shall specify these attributes to control formatting for
1318   text documents or HTML documents.

1319   A client need not specify these attributes for other types of
1320   documents, such as PostScript or PCL.

1321   2.3.1.1 width (cardinalUnits)

1322   This attribute specifies the media width for the document in
1323   characters.

1324   2.3.1.1 length (cardinalUnits)

1325   This attribute specifies the media length for the document in
1326   characters.

1327   2.3.1.1 left-margin (cardinalUnits

1328   This attribute specifies the left-margin for the document in
1329   characters.

1330   2.3.1.1 right margin (cardinalUnits)

1331   This attribute specifies the right margin for the document in
1332   characters.

1333   2.3.1.1 top-margin (cardinalUnits)

1334   This attribute specifies the top-margin for the document in lines.

1335   2.3.1.1 bottom-margin (cardinalUnits)

1336   This attribute specifies the bottom-margin for the document in lines.

1337   2.3.1.1 repeated tab stops (cardinalUnits)

1338   This attribute specifies the tab stops for the document in
1339   characters.

1340   2.3.1.1 header-text (string)

1341   This attribute specifies the header text for the document.

1342   2.3.1.1 footer text (string)

1343   This attribute specifies the footer text for the document.

1344   2.3.1.1 number-pages (boolean)

1345   This attribute specifies that the pages should be numbered in the
1346   document.

1347   2.3.1.1 default font (string)

1348   This attribute specifies the font to use for all text in the
1349   document.

1350   2.3.1.1 font-size (cardinalUnits)

1351   This attribute specifies the font-size in points for text in the
1352   document. The value of this attribute affects the size of the other
1353   text attributes.

1354   If this attribute is omitted and the Printer's default Job Template
1355   does not contain this attribute, the Printer shall assume a value of
1356   10. A value of 10 with a fixed pitch font, shall produce 12
1357   characters per inch in the horizontal direction and with 6 lines per
1358   inch in the vertical direction.

1359   2.3.1.1 default-code-set (type3Enum)

1360   This attribute specifies the code-set in which the document is
1361   encoded.

1362   2.3.1.1 content orientation (type2Enum)

1363   This attribute specifies the orientation of the document.

1364   The standard values are:

portrait   The page orientation such that the sides
                 are longer than the top when the page is
                 held in the intended human reading
                 orientation
landscape   The page orientation such that the sides
                 are shorter than the top when the page is
                 held in the intended human readable
                 orientation. Landscape is defined to be a
                 rotation of the page by +90 degrees with
                 respect to the medium (i.e. anti-clockwise)
                 from the portrait orientation
                 NOTE - The +90 direction was chosen because
                 simple finishing on the long edge is the
                 same edge whether portrait or landscape
reverse-   The page orientation defined to be a
portrait   rotation of 180 degrees with respect to
                 portrait
reverse-   The page orientation defined to be a
landscape   rotation of 180 degrees with respect to
                 landscape. Landscape is defined to be a
                 rotation of the page by -90 degrees with
                 respect to the medium (i.e. clockwise) from
                 the portrait orientation
                 NOTE   Reverse landscape was added because
                 some applications rotate landscape -90
                 degrees from portrait, rather than +90
                 degrees.

1365

1366 2.3.1 Job Resource Attributes (Set by the program that produces or
1367 senses the PDL)

1368   A program (described below) shall add these attributes, which
1369   describe the resources needed to print the job.

1370   A Printer may use these attributes to validate and schedule the
1371   print job without interpreting the contents of the document. This
1372   provides the opportunity for a Printer to support a broad set of
1373   document formats yet still support fast efficient scheduling and
1374   validation of each job.

1375   The client/end user shall not specify these attributes. Instead, it
1376   is the duty of the program that translates the document to the
1377   printer's PDL (or analyzes it) to add these attributes and their
1378   values to the job. Such a program may execute at a number of
1379   different points in time:

1380   1. The program produces a final form document and stores these
1381   resource attributes in a file before the end user submits the
1382   print job.

2. The program produces a final form document data stream when the
end-user specifies "Print" to the application program (e.g.,
Windows GDI driver).

3. The program running in the context of the Printer or server
translates a revisable or final form document into a PDL that the
output device understands.

If any of these attributes is unspecified, the Printer shall assume
that the all resources required by the document of the type specified
by the missing attributes are ready, ie., are available to the
Printer and/or output device without human intervention.

These attributes may be unspecified if the translation program fails
to provides such values, or if no translation occurs (e.g. the
document is a PostScript document.

Note: The Printer does not use these attributes during the actual
printing of a document.

Note: these attributes allow more than one value wherever it is
possible for a job to specify more than one value of the
corresponding job attribute, possibly by embedded instructions.

The client may specify these attributes in: Get Attributes and Get
Jobs.

See the section on job production attributes for an explanation of
how the job resource attributes differ from the job production
attributes.

2.3.1.1 document formats used (1#type2Format)

This attribute identifies the document formats needed to print the
document(s) in this job.

A format consists of two elements, a name and a version. The latter
element is optional.

The syntax is for type2Format:

        name ["/" version]

Examples include: PostScript, PostScript/2.0 and PCL/5e

Note: The version component is optional.

The names shall be registered with IANA as "printer languages"
following the procedures established by the Printer MIB (currently
proposed as an ITEF standard by RFC 1759).

1418   2.3.1.1 fonts-used (1#string)

1419   This attribute identifies the font resources used in the document(s)
1420   in the job.

1421   2.3.1.1 code sets used (1#type3Enum)

1422   This attribute identifies the code-sets used in the document(s) in
1423   the Job. This attribute is relevant only for files that are not in
1424   ASCII, such as text files and possibly PCL files. PostScript files
1425   are always ASCII. Normally there is at most 1 code set.

1426   Standard values are defined in the section specifying the default
1427   code-set attribute.

1428   2.3.1.1 media-used (1#type2Enum)

1429   This attribute identifies the media, media-sizes, input-trays or
1430   electronic forms needed to print the document(s) in the job.

1431   Standard values for this attribute are defined in the section
1432   specifying the medium-select attribute.

1433   2.3.1.1 sides-used (type2Enum)

1434   This attribute specifies whether a job needs 1-sided, 2-sided-long-
1435   edge, or 2 sided short edge printing.

1436   Standard values for this attribute are defined in the section
1437   specifying the sides Job attribute.

1438   2.3.1.1 print-quality-used (type2Enum)

1439   This attribute specifies what print quality the job needs.

1440   Standard values for this attribute are defined in the section
1441   specifying the print quality attribute.

1442   2.3.1.1 finishing-used (type2Enum)

1443   This attribute specifies what finishing the job needs.

1444   Standard values for this attribute are defined in the section
1445   specifying the finishing attribute.

1446   2.3.1.1 printer resolution used (positiveIntegerCrossState)

1447   This attribute specifies what resolution the job needs.

1448   The interpretation of the values for this attribute are defined in
1449   the section on printer-resolution-select Job attribute.

1450   2.3.1.1 total-job-octets (positiveInteger)

1451   This attribute specifies the total size of the job in octets. This
1452   attribute is the first of three that a translation program can use to
1453   specify the size of a job.

1454   2.3.1.1 job-impression-count (positiveInteger)

1455   This attribute specifies the total size of the job in impressions.

1456   2.3.1.1 job-media-sheet-count (positiveInteger)

1457   This attribute specifies the total size of the job in media sheets.

1458   2.3.1 Number of Documents (Set by Printer)

1459   This group contains a single attribute which specifies the number of
1460   documents in the job.

1461   The Printer sets the value of this attribute depending on the number
1462   of documents that the client supplies in the Print operation.   The
1463   client shall not specify this attribute (directly) in Print, but may
1464   specify this attribute in: Get-Attributes and Get-Jobs.

1465   2.3.1.1 number-of-documents (positiveInteger)

1466   This attribute specifies the number of documents in the job. Each
1467   document shall contain its own set of document content attributes
1468   described below.

1469   2.3.1 Document Data (Set by a Client/End User)

1470   This group of attributes describes the document data for the job.
1471   These attributes also include the document data or reference it.

1472   All job attributes in other sections of this document occur only once
1473   per job and apply to all documents in a job.

1474   The client may specify document-data attributes in Print. The client
1475   must specify either the document-URL or document-content in Print.

1476   Except for document-content, the client may specify document-data
1477   attributes in: Get Attributes, and Get Jobs.

1478   2.3.1.1 document format (type2Format)

1479   This attribute identifies the document format of this document.

1480   If the client does not specify this attribute, then the Printer shall
1481   attempt to determine the format in order to decide if the document
1482   data needs to be translated. The version component is optional.

1483   2.3.1.1 document-name (string)

1484   This attribute contains the name of the document used by the client
1485   to initially identify the document.

1486   2.3.1.1 document URL (url)

1487   This attribute contains the URL of the document if the client
1488   specified the document with a URL.

1489   If this attribute is specified, then document-content shall be
1490   unspecified.

1491   2.3.1.1 document content (octetString)

1492   This attribute contains the actual contents of the document.

1493   If this attribute is specified, then document-URL shall be
1494   unspecified.

1495   This attribute shall be used during the transmission of the Print
1496   operation over a network. A Printer shall save the document data to a
1497   file and reference it with the document-URL. A Get-Attribute or Get-
1498   Jobs operation shall always find that this attribute is unspecified.

1499   2.4 Operation Attributes (Set by Client)

1500   NOTE: These attributes have just been introduced and they are not as
1501   stable as the attributes in the other sections.   Some work is still
1502   needed to show the relationship between these attributes, job
1503   attributes, printer attributes, and authentication and authorization.

1504   The client shall set these attributes and associate them with an
1505   operation rather than an object.

1506   It is intended that a client program rather than an end user has
1507   control over the setting of these values so that they cannot be
1508   easily forged.

1509   2.3.1 operation-locale (type3Locale)

1510   This attribute identifies the locale of the client. The Printer uses
1511   this attribute to determine the locale of (1) messages in the result
1512   of the operation, (2) in errors returned by the operation or (3)
1513   notification events sent to the submitter.

1514   The standard values are defined in the section on the job-locale
1515   attribute.

1516   If an operation does not specify this attribute, the Printer shall
1517   assume that the operation has the same locale as the Printer.

2.3.1 operation-notification-address (url)

   This attribute specifies both the address and mechanism for delivery
   of events. If the URL has a "mailto:" scheme, then email is used and
   the rest of the URL is used as the email address. If the URL has a
   "http:" scheme, then an HTTP APPEND method is used to add HTML
   formatted events to the end of the specified HTML file.

2.3.1 operation-user-name (name)

   This attribute identifies the most authenticated end user name that
   the client can supply. This name identifies the end user performing
   the operation.

   This value shall be set by the system rather than the end-user in
   order to minimize the chance of forgery.

2.3.1 operation host name (name)

   This attribute identifies the most authenticated host name that the
   client can supply. This name identifies the host from which the
   operation comes.

   This value shall be set by the system rather than the end-user in
   order to minimize the chance of forgery.

2.4 Printer Attributes (Set by the Administrator)

   A printer object may be realized in either a Print Server or Output
   Device. Note: How these attribute are set by an Administrator is
   outside the scope of this specification.

   A Printer Object in an Output Device contains a set of printer object
   attributes that represent an Output Device capable of rendering a
   document in visible form. Examples include electronic and electro-
   mechanical printers such as laser printers, ink-jet printers, and
   various kinds of impact printers, but may include other types of
   output devices such as microfiche imagers and plotters as well.

   A Printer Object in a Print Server may supply queuing, spooling, and
   scheduling for an Output device that does not queue or spool.

   A Print Server, in the most common case, controls exactly one
   downstream Output Device. The Print Server's Printer object has
   attributes whose values are the same as those of the Printer object
   in the downstream Output Device.

   A Printer Object in a Print Server may contain a set of printer
   object attributes that are the union of the Printer objects in the
   downstream Output Devices. This object extends the capabilities of
   an Output Device. For example, an administrator might define a

1556   single Print Server to represent all of the Output Devices of the
1557   same type and capability in a single location, associated with a
1558   particular server. A end user would normally send a print job to a
1559   Print Server, and allow the Print Server to assign the job to a
1560   particular Output Device based on the relative load and availability
1561   of the printers under its control, thus providing a load balancing
1562   service. However, nothing precludes an administrator from
1563   configuring a print system so that an end user can send a print job
1564   directly to an Output Device.

1565   The attributes defined in this section provide information about a
1566   particular Printer.

1567 2.3.1 printer-name (name)

1568   This attribute uniquely identifies the printer on its host.

1569 2.3.1 printer location (string)

1570   This attribute identifies the location of this printer.

1571 2.3.1 printer-model (string)

1572   This attribute identifies the make and model of the printer.

1573 2.3.1 printer type (type2Enum)

1574   This attribute identifies the marking technology of the printer.

1575   The standard values for this attribute are the descriptive names
1576   specified by ISO DPA which have corresponding enum symbolic and
1577   numeric values assigned by the Printer MIB (RFC 1759).. These
1578   standard values are:

other	Other than the standard values
unknown	Unknown printer type
electrophotographic-LED	electrophotographic LED
electrophotographic-laser	electrophotographic laser
electrophotographic-other	other electrophotographic
impact-moving-head-dot-matrix-9-pin	9-pin impact moving head dot matrix
impact moving head dot-matrix 24 pin	24 pin impact moving head dot matrix
impact-moving-head-dot-matrix-other	neither 9-pin nor 24-pin moving head dot matrix
impact-moving-head-fully formed	fully formed impact moving head
impact band	impact band
impact-other	impact other
inkjet-aqueous	aqueous inkjet

```
            inkjet-solid              solid inkjet
            inkjet-other              other inkjet
            pen                       pen
            thermal-transfer          thermal transfer
            thermal-sensitive         thermal sensitive
            thermal-diffusion         thermal diffusion
            thermal-other             other thermal
            electro erosion           electro erosion
            electro static            electro static
            photographic-microfiche   photographic microfiche
            photographic-             photographic imagesetter
            imagesetter
            photographic other        other photographic
            ion deposition            ion deposition
            E-beam                    E-beam
            typesetter                typesetter
```

1579

1580   2.3.1 printer state (type1Enum)

1581   This attribute identifies the current state of the printer and shall
1582   be set by the Printer. The protocol support all values for printer
1583   states, however a Printer shall only generate the printer states
1584   which are appropriate for the particular implementation.

1585   The following standard values are defined:

```
            unknown         The printer state is not known, or is
                            indeterminate, or is not returned by the
                            operation
            idle            The printer is ready to accept jobs, but
                            none have been scheduled on it.
            printing        The printer is currently printing a job
            needs-          The printer needs human attention (no
            attention       special skills required).  This state
                            typically includes adding paper, clearing a
                            jam, changing the medium, etc.
            paused          The operator has (temporarily) paused the
                            printer, by means outside the scope of IPP
                            V1.0.
            shutdown        The printer has been taken out of service,
                            (for a long time), whether for repairs or
                            others reasons.  The printer's message
                            generic attribute may be used to record a
                            reason and estimated time for return to
                            service
            job start-      The currently processing job was started
            wait            with the job start wait attribute set, and
                            is awaiting operator intervention or time-
                            out.
```

job-end-wait	The currently processing job was started with the job-end-wait attribute set, and is awaiting operator intervention or time out.
job password wait	The currently processing job was started with the job-password attribute set, and is awaiting the operator or user to enter the password supplied by the job-password attribute.
needs key operator	The printer needs the attention of a key operator. Key operator functions are printer-specific, but typically include adding toner or developer, or attending to a hardware fault.
connecting to-printer	The server has scheduled a job on the printer and is in the process of connecting to a shared network printer (and may not be able to actually start printing the job for an arbitrarily long time depending on the usage of the printer by other servers).
timed-out	The server was able to connect to the printer (or is always connected), but was unable to get a response from the printer in the time specified by the printer's printer timeout period attribute.

1586

1587   2.3.1 printer-state-message (string)

1588   This attributes specifies a message that gives further information
1589   about the current printer state and shall be set by the Printer.

1590   2.3.1 message (string)

1591   This attribute provides a message from an operator, system
1592   administrator or "intelligent" process to indicate to the end user
1593   information or status of the printer, such as why it is unavailable
1594   or when it is expected to be available.

1595   2.3.1 printer job templates (1#urlDefault)

1596   This attribute identifies the URL of each of the Job Templates that
1597   this Printer is associated with and the one Job Template this Printer
1598   uses as its default for supply job attributes that the client omits.
1599   There shall be only one value with the default qualifier. Other
1600   Printers can be associated with the same Job Templates.

1601   The syntax is:

1602   url [":" default]

2.3.1 locale (type3Locale)

This attribute specifies the locale that the Printer operates in.

The standard values are defined in the section on the job locale
attribute.

2.3.1 notification-events (1#type2Enum)

This attribute specifies the events on whose occurrence the Printer
should notify those addresses specified by the notification-addresses
attribute.

If the attribute is unspecified, the Printer does not perform
notification, though the Printer still checks the job's notification-
events attribute.

In this attribute, job-problem and printer-problem have the same
meaning.

The standard values are defined in the section on the job's
notification-events attribute.

NOTE - This attribute is intended to notify operators, not end-users.

2.3.1 notification-addresses (1#url)

This attribute specifies the method and addresses to which the
Printer should send messages when events specified by the
notification-events attribute occur.

If the attribute is unspecified, the Printer does not perform
notification, though the Printer still checks the job's notification-
events attribute.

NOTE - This attribute is intended to notify operators, not end users.

2.3.1 end-user-acl (1#name)

This attribute specifies the end users who are allowed to print on
the Printer.

If the attribute is unspecified, the Printer allows anyone to print.

2.3.1 maximum printer speed (positiveIntegerUnits)

This attribute indicates the maximum printer speed of the Printer in
units of pages per minute, impressions per minute, lines per minute,
and characters per minute. A job cannot control a Printer's speed,
but a Printer Browser can use printer speed as a criteria.

1636   The standard units are a type2Enum and are: ppm, ipm, spm, lpm, cps.

1637   2.3.1 fonts-substitutions (1#stringPair)

1638   This attribute specifies an appropriate substitute for a font that is
1639   advertised as supported in the fonts supported attribute, even though
1640   the Printer doesn't actually have the font available.

1641   This attribute consists of a set of font pairs: a font name and the
1642   font to use instead.

1643   If this attribute is unspecified, the Printer does not perform any
1644   font substitutions.

1645   2.3.1 fonts-supported (1#stringState)

1646   This attribute identifies the font resources supported by this
1647   printer and indicates the state of readiness for each font.

1648   The standard names are defined in the section on default font.

1649   Each item in the list contains the pair consisting of a font name and
1650   a state indicating the font's readiness state.

1651   2.3.1 media-supported (1#nameState)

1652   This attribute identifies the media, media-sizes, input trays, and
1653   electronic forms supported by this printer, and indicates the state
1654   of readiness for each medium resource.

1655   The standard names are defined in the section on the section on the
1656   medium-select.

1657   Standard states are: not-ready, on-order, and special-order. The
1658   omission of a state shall indicate that the medium is ready, i.e.,
1659   can be used without human intervention..

1660   2.3.1 document-formats-supported (1#type2FormatState)

1661   This attribute identifies the document-formats, including the
1662   document-format-versions, supported by the Printer. This set includes
1663   both the formats that are native to the Printer and those formats
1664   that the Printer can translate to one that is native to the Printer.
1665   From the client's point of view, this set contains all formats in
1666   which documents can be submitted to this Printer.

1667   Proprietary document format identifiers, and versions are assigned by
1668   the owners of those formats.

1669   The state of readiness for each format is also included, though all
1670   formats should normally always be ready.

2.3.1 numbers-up-supported (1#type3EnumState)

   This attribute identifies the number-up values supported by this
   printer..

   The state of readiness for each number up value is also included,
   though all number-up conversions should always be ready.

2.3.1 finishings-supported (1#type2EnumState)

   This attribute identifies the finishing operations supported by this
   Printer and states of readiness for each finishing.

   The standard finishing objects are defined in the section on the
   finishing Job attribute.

2.3.1 sides-supported (1#type2EnumState)

   This attribute indicates the values of the sides attribute supported
   by this printer and the states of readiness of each value.

   The standard values are defined in the section on the sides
   attribute.

2.3.1 print-qualities-supported (1#type2EnumState)

   This attribute indicates the values of the printer-quality attribute
   supported by this printer and the states of readiness for each print
   quality value.

   The standard values are defined in the printer-quality attribute.

2.3.1 printer-resolutions-supported (1#positiveIntegerCrossState)

   This attribute indicates the values of the printer-resolution-select
   attribute supported by this printer and their states of readiness.

   The state of readiness for each printer resolution is also included,
   though normally all printer-resolutions should always be ready.

   The syntax is discussed in the section on the printer-resolution-
   select attribute.

2.3.1 code-sets-supported (1#type3EnumState)

   This attribute indicates the values of the default code set attribute
   supported by this printer and the states of readiness for each code
   set.

   The standard values are defined in the default-code-set attribute.

1703   2.3.1 off-peak-times-supported (1#type3EnumState)

1704   This attribute indicates the values of the job-print-off-peak
1705   attribute supported by this printer and the states of readiness for
1706   each value.

1707   If this attribute is unspecified, then the Printer has no off-peak
1708   periods.

1709   The standard values are defined in the section on the job-print-off-
1710   peak Job attribute.

1711   Note: this document does not define how an administrator associates
1712   the off-peak names with actual time periods.

1713   2.3.1 events-supported (1#type2EnumState)

1714   This attribute indicates the values of the job and printer
1715   notification events attribute supported by this Printer and the
1716   states of readiness for each value.

1717   If this attribute is unspecified, then the Printer does not support
1718   notification.

1719   The standard values are defined in the section on the notification-
1720   events attribute.

1721   2.3.1 locales supported (1#type3LocaleState)

1722   This attribute indicates the values of the job-locale attribute
1723   supported by this Printer and the states of readiness for each value.

1724   The standard values are defined in the section on the job-locale
1725   attribute.

1726   2.3.1 job sheets supported (1#type3EnumState)

1727   This attribute identifies the job-sheet values supported by this
1728   printer, and the state of readiness for each job-sheet.

1729   To allow no job sheets, the system administrator shall include the
1730   value "none" as a value for this attribute. The client specifies that
1731   there are no job sheets by using the value "none" as the value of the
1732   job-sheets attribute.

1733   If the job-sheets attribute is not specified or contains a value
1734   which the Printer does not support, then the server shall select from
1735   among the values of this attribute. The server shall not select the
1736   value "none" unless it is the only value specified for the job-
1737   sheets-supported attribute.

NOTE - When the client supplies a value other than "none", it is
preferable for the server to produce some job jobsheet, even if not
the desired one, rather than produce none at all or reject the job.

2.3.1 maximum copies (positiveInteger)

This attribute indicates the maximum number of copies of a document
that can be rendered by this printer in a single print-job.

If the attribute is unspecified, there is no limit on the maximum
number of copies for this Printer.

2.3.1 maximum job octets (positiveInteger)

This attribute indicates that the Printer shall accept a job only if
its size in octets is less that the value specified by this
attribute.

If the attribute is unspecified, there is no limit on the size of a
job in octets.

2.3.1 maximum-impressions (positiveInteger)

This attribute indicates that the Printer shall accept a job only if
its size in impression is less that the value specified by this
attribute.

If the attribute is unspecified, there is no limit on the size of a
job in impressions.

2.3.1 maximum-media-sheets (positiveInteger)

This attribute indicates that the Printer shall accept a job only if
its size in media sheets is less that the value specified by this
attribute.

If the attribute is unspecified, there is no limit on the size of a
job in media-sheets.

2.3.1 maximum-job-retention-period (deltaTime)

This attribute indicates that when the Printer accepts a job, the
retention period must not exceed the value of this attribute.
Otherwise, the Printer sets the job's retention-period to the value
of this attribute.

If this attribute is unspecified, then the Printer places no limit on
the retention time.

1771   2.3.1 maximum-end-user-priority (type1Enum)

1772   This attribute indicates that when the Printer accepts a job, the
1773   job priority must not exceed the value of this attribute. Otherwise,
1774   the Printer sets the job's job priority to the value of this
1775   attribute.

1776   If this attribute is unspecified, then the Printer places no limit on
1777   the job-priority.

1778   The standard values are defined in the section on the job priority
1779   attribute.

1780   2.3.1 queued-job-count (cardinal)

1781   This attribute contains a count of the number of jobs that are either
1782   pending and/or processing and shall be set by the Printer.

1783   2.3.1 scheduling algorithm (type3Enum)

1784   This attribute indicates the current scheduling algorithm for this
1785   Printer. Standard values are: "none", "smallest-job-first", "time-
1786   received".

1787   2.4 Job Templates

1788   The attributes for a Job Template can be any of the Job object
1789   attributes defined in the sections:

1790   ──────   Job Sheet Attributes
1791   ──────   Notification Attributes
1792   ──────   Job Scheduling Attributes
1793   ──────   (except job print after)
1794   ──────   Job Production Attributes
1795   ──────   (except page-select)
1796   ──────   Attributes for Conversion of Text and HTML Files
1797
1798   2.4 Conformance

1799   A conforming implementation shall implement all operations, objects
1800   and attributes defined in this document.

1801   Also, for the core set of attributes listed in this specification, it
1802   is not required that a conforming server support all (standard)
1803   values of all supported attributes. For example, it is not required
1804   that a printer implement all finishing methods indicated by the
1805   standard values.

1806   The explicit requirement of the term "supported", with respect to one
1807   of the attributes that deal with printer functions or resources, is
1808   that the server shall recognize the attribute and those values that

1809   are supported, and shall be able to respond to a query about which
1810   values that printer does, in fact, support.

1811   IPP is explicitly designed to be extensible. Additional attributes
1812   can be proposed to be registered by going through the type 2 enum
1813   process which will register their specification after approval with
1814   IANA. In addition specific implementation instances may support not
1815   only the basic protocol as defined in this specification, but may add
1816   vendor specific private extensions by prefixing attribute names with
1817   their company name registered with IANA for use in domains. See
1818   attribute syntax section. However, such private extensions shall not
1819   duplicate attribute semantics already in this specification.

1820 3. Security Considerations

1821   This protocol does not identify any new authentication mechanisms.
1822   The authentication mechanisms built into HTTP (such as SSL and SHTTP)
1823   are recommended.

1824   This protocol does define a simple authorization mechanism by
1825   introducing the "end-user-acl" attribute as part of the Printer
1826   object. This ACL attribute is a multi-valued list of all of the
1827   authenticated names of end-users. This protocol does not specify
1828   what the domain is for names in this ACL attribute.

1829   Issue: Will it always be possible for a Printer to obtain a
1830   meaningful authenticated name that the Printer can match against the
1831   end-user-acl, or will some other mechanism be necessary, such as a
1832   password?

1833 4. References

1834   [1]   Smith, R., Wright, F., Hastings, T., Zilles, S., and
1835   Gyllenskog, J., "Printer MIB", RFC 1759, March 1995.
1836
1837   [2]   Berners-Lee, T, Fielding, R., and Nielsen, H., "Hypertext
1838   Transfer Protocol - HTTP/1.0", RFC 1945, August 1995.
1839
1840   [3]   Crocker, D., "Standard for the Format of ARPA Internet Text
1841   Messages", RFC 822, August 1982.
1842
1843   [4]   Postel, J., "Instructions to RFC Authors", RFC 1543, October
1844   1993.
1845
1846   [5]   ISO/IEC 10175 Document Printing Application (DPA), Final, June
1847   1996.
1848
1849   [6]   Herriot, R. (editor), X/Open A Printing System Interoperability
1850   Specification (PSIS), August 1995.
1851
1852   [7]   Kirk, M. (editor), POSIX System Administration - Part 4:
1853   Printing Interfaces, POSIX 1387.4 D8, 1994.

1854
1855   [8]   Borenstein, N., and Freed, N., "MIME (Multi-purpose Internet
1856   Mail Extensions) Part One: Mechanism for Specifying and
1857   Describing the Format of Internet Message Bodies", RFC 1521,
1858   September, 1993.
1859
1860   [9]   Braden, S., "Requirements for Internet Hosts - Application and
1861   Support", RFC 1123, October, 1989,
1862
1863   [10] McLaughlin, L. III, (editor), "Line Printer Daemon Protocol"
1864   RFC 1179, August 1990.
1865
1866   [11] Berners-Lee, T., Masinter, L., McCahill, M. , "Uniform Resource
1867   Locators (URL)", RFC 1738, December, 1994.
1868
1869   5. Author's Address

1870   Scott A. Isaacson
1871   Novell, Inc.
1872   122 E 1700 S
1873   Provo, UT   84606
1874
1875   Phone: 801-861-7366
1876   Fax: 801 861 4025
1877   EMail: scott_isaacson@novell.com
1878
1879   Tom Hastings
1880   Xerox Corporation
1881   701 S. Aviation Blvd.
1882   El Segundo, CA   90245
1883
1884   Phone: 310-333-6413
1885   Fax: 310-333-5514
1886   EMail: hastings@cp10.es.xerox.com
1887
1888   Robert Herriot
1889   Sun Microsystems Inc.
1890   2550 Garcia Ave., MPK-17
1891   Mountain View, CA 94043
1892
1893   Phone: 415-786-8995
1894   Fax: 415-786-7077
1895   Email: robert.herriot@eng.sun.com
1896
1897   Roger deBry
1898   HUC/003G
1899   IBM Corporation
1900   P.O. Box 1900
1901   Boulder, CO 80301-9191
1902
1903   Phone: (303) 924-4080
1904   Fax: (303) 924-9889
1905   Email: debry@vnet.ibm.com

Other Participants:

Devon Taylor, Novell, Inc.
Mike MacKay, Novell, Inc.
Peter Zehler, Xerox, Corp.
Keith Carter, IBM Corporation
Carl-Uno Manros, Xerox, Corp.
Don Wright - Lexmark
Steve Gebert - IBM
Ray Lutz - Cognisys
Mabry Dozier - QMS
Lee Ferrel - Canon
Hiro Sato - Canon
Pat Nogay - IBM
Jim Walker - Dazel
Jay Martin - Underscore
Bill Wagner - DPI
Stan McConnell - Xerox
Bob Setterbo - Adobe
Randy Turner - Sharp
Rob Whittle - Novell
Ron Bergman - Data Products
Lloyd Young - Lexmark
Andy Davidson - Tektronix
Rick Landau - Digital
David Kellerman - Northlake Software
David Roach - Unisys
Mike Timperman - Lexmark
Chris Wellens - Interworking Labs
Pete Loya - HP
Bob Pentecost - HP
Harry Lewis - IBM
William Wagner - Digital Products
Atsushi Yuki - Kyocera
Rob Rhoads - Intel
Jeff Barnett - IBM
Jim Walker - Dazel
Jeff Copeland - QMS
Chuck Adams - Tektronix

1948 6. Appendix A: Sample IPP Operations

1949   The following examples illustrate typical flows using the IPP
1950   protocol. In these examples, the IPP Printer object named "printer 1"
1951   is located at the node identified by the DNS name "some.domain.com".
1952   A Job Template has been defined for printer-1 which establishes the
1953   print defaults.

1954   For brevity in the following flows, none of the HTTP headers are
1955   shown. CRLF sequences are not shown.

1956 6.1 Querying the printer

1957   Client   some.domain.com
1958
1959   -->
1960   Post http://some.domain.com/printer 1   http/1.0
1961   Get Attributes IPP/1.0
1962   printer-state :
1963   sides-supported :
1964   media-supported :
1965   document formats supported :
1966
1967   <--
1968   http/1.0 201 "Created" (a response)
1969   IPP/1.0 xxx "attribute list returned"
1970   printer state : idle
1971   sides supported : 1 sided
1972   media-supported : iso-a4-white, iso-b4-white
1973   document-formats-supported : Postscript/2.0
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991 6.1 Print Operation   with print data included

1992   Client   some.domain.com
1993
1994   -->

```
    Post http://some.domain.com/printer-1  http/1.0
      Print IPP/1.0
      Print Job Object Header
         job name : My Job
         medium : iso-a4-white
         notification-events : Job-completion
         notification-address : joe@pc.domain.com
      Document Header
         document name : Letter to Mom
      Document-Content Header (content type = Postscript/2.0)
         <Document in Postscript level 2 format>



    <─────────────────────────────────
    http/1.0  200 "accepted"
      IPP/1.0 xxx "print job accepted and queued"
         job-identifier : some.domain.com/printer-1/0037
         current job state : pending
         printer state : needs sttention

6.1 Print Operation - with no data included

    Client───────────────────────────── some.domain.com


    ───────────────────────────────────>
    Post http://some.domain.com/printer-1  http/1.0
      Print IPP/1.0
      Print-Job-Object Header
         job name : My Job
         medium : iso a4 white
         notification-events : Job-completion
         notification-address : joe@some.domain.com
      Document Header
         document name : Letter to Mom
         document URL : joe@pc.domain.com/Docs/To mom.ps

  <─────────────────────────────────
    http/1.0  200 "accepted"
      IPP/1.0 xxx "print job accepted and queued"
         job identifier : some.domain.com/printer 1/0037
         current-job-state : pending
         printer-state : processing
6.1 Querying the state of the job

    In this example, no attributes are specified, so all job attributes
    are returned.

    Client───────────────────────────── some.domain.com


    ───────────────────────────────────>
    Post http://some.domain.com/printer-1/0037  http/1.0
      Get Attributes  IPP/1.0
```

```
2043
2044
2045        <──────────────────────────────────────────
2046        http/1.0 201 "Created" (a response)
2047          IPP/1.0  xxx "atribute list returned"
2048          job Name : My Job
2049          job-Originator : Joe@some.domain.com
2050          job originating host : pc.domain.com
2051          notification address : joe@pc.domain.com
2052          job-locale : xx:xx:xx
2053          current-job-status : printing
2054          submission-time : 1996 Nov 22 1214
2055          media sheets completed : 2
2056
2057
2058    6.1 Canceling a Job

2059        Client ─────────────────────────────── some.domain.com

2060        ──────────────────────────────────────────>
2061        Post: http://some.domain.com/printer 1/0037
2062          Cancel-Job  IPP/1.0
2063
2064
2065
2066        <──────────────────────────────────────────
2067        http/1.0  200 "okay"
2068        Current-job-state : terminating
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079    6.1 Listing jobs on a Printer

2080        List jobs on printer 1, only return job sizes. Jobs are returned in
2081        the order they are scheduled for printing. A Job identifier attribute
2082        precedes the attributes returned for each job to delimit job
2083        boundaries.

2084        Client ─────────────────────────────── some.domain.com

2085        ──────────────────────────────────────────>
2086        Post http/1.0 some.domain.com/printer 1
2087          Get-Jobs  IPP/1.0
2088            total-job-octets :
2089
```

```
2090        <----------------------------------------
2091     http/1.0  201 "Created" (a response)
2092       -- IPP/1.0 xxx "created an attribute list"
2093       -- job-identifier : 0033
2094       -- total-job-octets : 4567
2095       -- job-identifier : 0034
2096       -- total-job-octets : 12345
2097       -- job-identifier : 0035
2098       -- total-job-octets : 12356

2099
```