

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59

INTERNET-DRAFT

Ron Bergman
Hitachi Koki
Harry Lewis
IBM Corp.
May 9, 2000

Printer Finishing MIB

<draft-ietf-printmib-finishing-09.txt>

Expires November 9, 2000

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of section 10 of RFC2026.

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 "lid-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), ftp.ietf.org (US East Coast), or ftp.isi.edu (US West Coast).

Copyright Notice

Copyright (C) The Internet Society (1998). All Rights Reserved.

Abstract

This document defines a printer industry standard SNMP MIB for the management of printer finishing device subunits. The finishing device subunits applicable to this MIB are an integral part of the Printer System. This MIB does not apply to a Finisher Device that is not connected to a Printer System.

The Finisher MIB is defined as an extension of the Printer MIB [PrtMIB] and it is expected that the information defined in this document will be incorporated into a future update of the Printer MIB.

65 TABLE OF CONTENTS

66
67 1. INTRODUCTION.....2
68 1.1 Scope.....2
69 1.2 Rational.....3
70 2. TERMINOLOGY.....3
71 2.1 General Terminology.....4
72 2.2 Process Specific Terminology.....6
73 3. FINISHER SUBUNITS INTEGRATED INTO THE PRINTER MODEL.....9
74 4. FINISHING SPECIFICATIONS.....10
75 4.1 Multiple finDeviceTable Entries.....10
76 4.2.1 FinPunchPatternTC.....11
77 4.2.2 FinPunchHoleTypeTC, punchHoleSizeMaxDim, punchHoleSizeMinDim...12
78 5. THE ATTRIBUTE MECHANISM.....12
79 5.1 Conformance of Attribute Implementation.....13
80 5.2 Useful, 'Unknown', and 'Other' Values for Objects and Attributes.13
81 5.3 Data Sub-types and Attribute Naming Conventions.....14
82 5.4 Single-Value (Row) Versus Multi-Value (MULTI-ROW) Attributes.....14
83 5.6 Index Value Attributes.....15
84 5.7 Attribute Specifications.....15
85 6. ENUMERATIONS.....20
86 6.1 Registering Additional Enumerated Values.....20
87 7. PRINTER FINISHING MIB SPECIFICATION.....21
88 8. REFERENCES.....43
89 9. SECURITY CONSIDERATIONS.....44
90 10. FULL COPYRIGHT STATEMENT.....44
91 11. AUTHORS.....45

92
93
94 1. INTRODUCTION

95
96 This document describes an SNMP Management Information Base (MIB) to
97 provide for the management of in-line post-processing in a fashion
98 that is currently provided for printers, using the Printer MIB
99 [PrtMIB]. The Printer Finishing MIB includes the following features:

- 100
101 - Provides the status of the finishing device.
102 - Queries and controls the features and configuration of the
103 finishing device.
104 - Enables and disables the finishing processes.
105 - Allows unsolicited status from the finishing device.
106

107
108 1.1 Scope

109
110 This document provides a robust set of finishing devices, features,
111 and functions, based upon today's state of the art of in-line
112 finishing. Since finishing typically accompanies higher speed
113 network printers and copiers, in contrast to simple desktop devices,
114 no attempt is made to limit the scope to "bare minimum". On the
115

121
122 other hand, the Printer Finishing MIB does not duplicate the
123 production mail preparation, custom insertion, franking, and reprints
124 that are covered by the DMTF Large Mailing Operations standard [LMO].
125

126 Information supplied by the Printer Finishing MIB may be utilized by
127 printer and finisher management applications engaged in monitoring
128 status and managing configuration, and also used by print and
129 finishing submission applications which are engaged in:

- 130
131 - print-job-level finishing processes that are applied to a
132 complete print job,
133
- 134 - document-level finishing processes that are applied individually
135 to each document in the print job,
136
- 137 - document-level finishing processes that are applied to a selected
138 document in the print job.
139

140 Note that not all combinations of finishing processes are
141 permitted. Compatible combinations of finishing processes are
142 implementation specific. The MIB allows invalid combinations to be
143 identified.
144

145 The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
146 "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
147 document are to be interpreted as described in [RFC2119].
148
149

150 1.2 Rational

151
152 The Printer MIB [PrtMIB] is now successfully deployed in a large
153 segment of the network printer market. SNMP and/or HTTP enabled
154 printers and software management applications are growing in numbers.
155

156 There is an increase in the availability of network printers and
157 copiers that include in-line finishing processes. Thus a well
158 defined and ordered set of finishing objects is now necessary for
159 printer management.
160

161 The printer model defined in the Printer MIB includes finishing
162 processes and the MIB was designed to later incorporate finisher
163 objects or to be referenced by a future Finisher MIB.
164

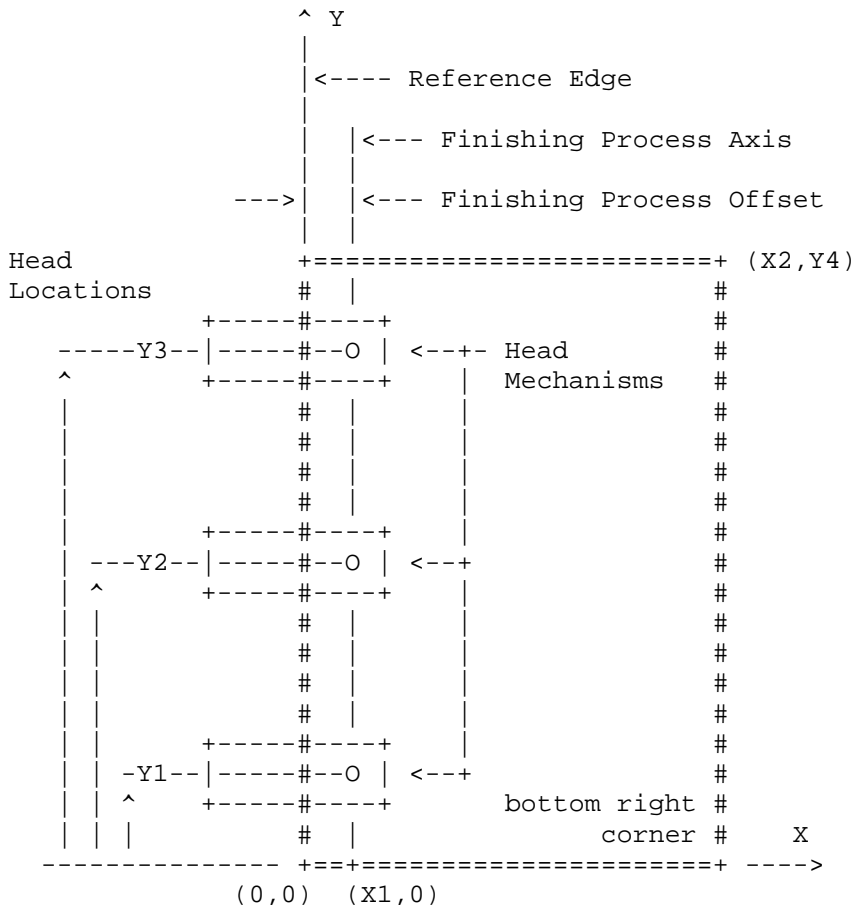
165 166 167 2. TERMINOLOGY

168
169 Where appropriate, the Printer Finishing MIB will conform to the
170 terminology, syntax, and semantics from the DMTF Large Mailing
171 Operations standard [LMO], the Internet Printing Protocol [IPP],
172 and/or the ISO Document Printing Application [DPA].
173
174

180
 181 2.1 General Terminology
 182

183 Finisher Input: An input tray on the finisher and not otherwise
 184 associated with the printer. An example of a finisher input is a
 185 tray that holds finishing "inserts".
 186

187 Finisher Output: The output of the finisher. Because processing is
 188 in-line, the finisher outputs are a direct extension of the set of
 189 printer outputs.
 190



221
 222 Figure 1 - Finishing Process Axis Parallel to Y Axis
 223
 224
 225
 226
 227
 228
 229
 230
 231
 232

234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291

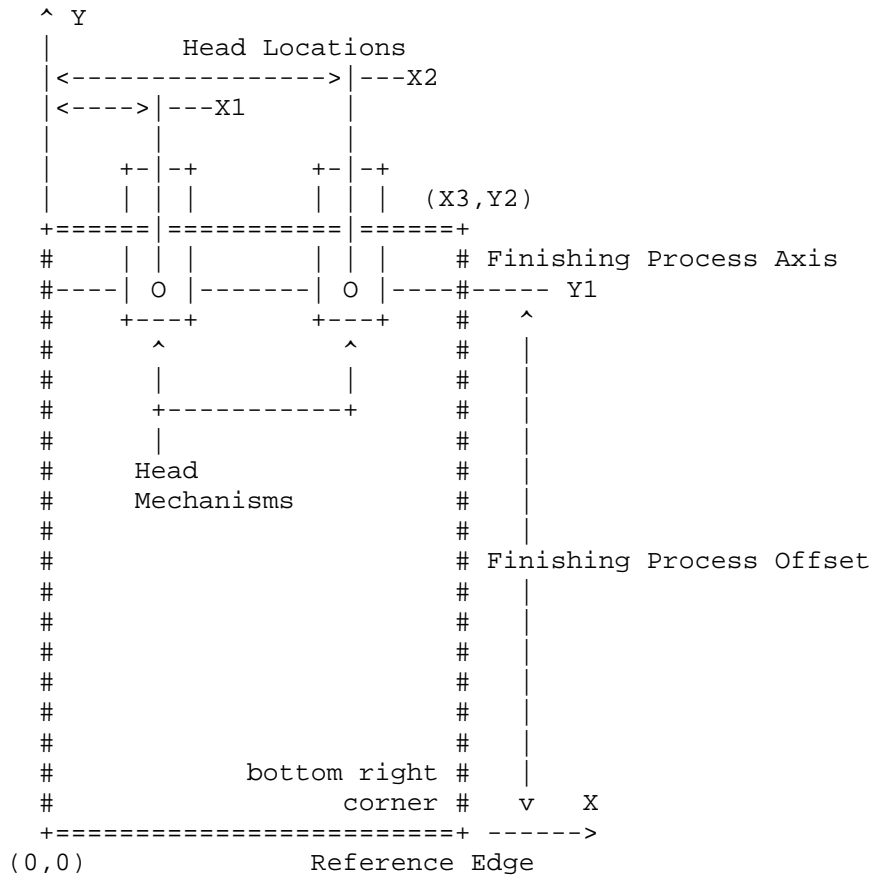


Figure 2 - Finishing Process Axis Parallel to X Axis

Media Orientation: All Finishing Processes are defined relative to a portrait orientation of the medium, regardless of the orientation of the printed image or the direction of feed. The 'X' and 'Y' axis, therefore, will always reference the medium as shown in figures 1 and 2, with the 'X' axis always along the short edge of the medium. All edges and corners are also defined with the medium orientation as shown using the syntax top, bottom, left, and right. Thus the bottom edge of the medium is at Y = 0, the left edge is at X = 0, and the bottom right corner is at (X2,0) as shown in the figure 1 and at (X3,0) as shown in figure 2.

Finishing: Defined by DPA as an operation on a document following the completion of the image process. Finishing processes defined within this document are those applied to one or more instances of rectangular paper sheet media.

Finishing Process: Defined by DPA as an operation applied by a machine such as trimming a document, folding the sheets in a

296 document, and applying a binding to a document.

297
298 Finishing Specification: Defined by DPA as the specific sequence of
299 operations for a serial combination of finishing processes. The
300 exact sequential order of the processes, in many cases, is
301 critical to the obtaining the desired result. For example, a
302 folding operation followed by trimming could provide a very
303 different result than if the trimming was followed by the folding.
304

305 Finishing Process Parameters: This parameter set is used to create a
306 detailed definition of the finishing process. Generic Finishing
307 Process Parameters are applicable to any Finishing Specification.
308

309 - Head Mechanism: Defined by DPA as the physical mechanism that is
310 used to perform a finishing process. The head position may be
311 fixed or variable depending upon the capabilities of the device.
312

313 - Reference Edge: Defined by DPA as the edge of the document
314 relative to the axis to which the finishing process is applied.
315 The edge of the medium defined to be the Reference Edge may be
316 either the 'X' or the 'Y' axis, depending upon the finishing
317 process to be performed.
318

319 Note that the Reference Edge may change from one finishing process
320 to another for one of two reasons. First, a subsequent process may
321 require a different Reference Edge. Second, the actual dimensions
322 of the document may change, for example as a result of a folding or
323 a trimming operation.
324

325 - Jog Edge: Defined by DPA as one of the two edges that is
326 perpendicular to the Reference Edge. Specifying the Jog Edge
327 parameter indicates the edges of all sheets which correspond to
328 the Jog Edge are aligned.
329

330 - Finishing Process Axis: Defined by DPA as the axis to which some
331 finishing processes are applied to or referenced from by the Head
332 Mechanism. Examples are the axis for a fold process or the axis
333 for a punch process.
334

335 - Head Locations: Defined by DPA as the position of the Heads on the
336 Finishing Process Axis.
337

338 - Finishing Process Offset: The offset from the Reference Edge to
339 the Finishing Process Axis at which the finishing process takes
340 place or is applied.
341

342 2.2 Process Specific Terminology

343 FOLDING:

344
345
346 Z Fold: A fold in which two folds are placed in the sheet in opposite
347
348

354 directions. The first fold is located at 25% of the sheet length,
355 and the second is located at 50% of the sheet length (i.e., the
356 center of the sheet). Z Folding is often used on 11x17 inch or A3
357 size sheets, when they are included in sets containing 8.5x11 inch
358 or A4 size sheets.

359
360 Half Fold: To fold a sheet in half so that one of the resulting
361 dimensions are exactly half the original sheet. Often used for
362 signatures or booklets.

363
364 Letter Fold: Folding a sheet roughly in thirds. Usually performed on
365 8.5x11 inch or A4 size sheets for insertion into an envelope.

366
367 Signature: The process by which images are placed on a large sheet of
368 paper in correct panel areas and in the proper orientation such that
369 when the sheet is folded it will produce a booklet with each page in
370 the proper order and orientation.

371
372
373 BINDING:

374
375 Adhesive Binding: A method of attaching sheets together to form a
376 book or booklet using glue or adhesive. Some adhesive binding
377 methods apply the glue to sheets individually, before merging them
378 together for form a book, but most methods involve the application
379 of adhesive to an entire book of sheets.

380
381 Comb Binding: A method of binding in which a series of small
382 rectangular holes is placed along the bind edge the sheets. The
383 sheets are then held together using a tube shaped plastic binding
384 strip with comb like fingers that fit through the holes in the
385 sheets.

386
387 Spiral Binding: Sometimes referred to as wire binding, this binding
388 method is a mechanical bind in which the individual leaves are held
389 together by a wire or plastic spiral that is fed through small holes
390 in the paper binding edge.

391
392 Padding: Applying a non-penetrating adhesive to the edge of a stack
393 of sheets such that the sheets can be easily peeled off one at a
394 time. Frequently used for forms.

395
396 Velo Binding: A bind formed by punching holes into the edge of the
397 sheets, placing a two piece plastic strip (one side formed with
398 plastic pins that pass through the holes) along the edge and then
399 staking the two pieces together.

400
401 Perfect Binding: A method of binding in which all pages are cut and
402 roughed up at the back or binding edge and held together by an
403 adhesive.

412 Tape Binding: The act of placing tape over the bind edge of a set.
413 Sometimes contains adhesive to provide a functional bind to the set,
414 and sometimes done for decorative purposes on a set that has been
415 edge stapled.

416
417
418 SLITTING/CUTTING/TRIMMING:
419

420 Trim: To cut the edges of a sheet or set of sheets.
421

422 Face Trim: To cut the edges of a set of sheets on a booklet of sheets
423 that have been folded to eliminate the "creep" or edge shingling
424 that results from the folding process.
425

426 Gutter Trim: To cut a larger sheet into smaller sheets eliminating
427 the gutter between adjacent images. This operation requires a
428 minimum of two cuts for each gutter.
429

430 Tab Cutting: The act of cutting the edge of a sheet to form an index
431 tab, thereby allowing quick identification and access. The external
432 tabs are sequentially placed along the book edge for visibility and
433 ease of grasping.
434

435 Perforating: The act of cutting a series of very small, closely
436 spaced holes or slots into a sheet to allow for ease of separation
437 of a portion of the sheet. Sometimes also used to ease
438 bending/hinging of heavy weight papers.
439

440 Scoring: A means of applying small linear grooves or impressions
441 along a sheet to allow easy folding. Often used on heavy weight
442 sheets and book covers.
443

444 Slitting: The action of cutting apart a large sheet to form smaller
445 sheets. Usually done using a sharp circular roll system.
446
447

448 STITCHING/STAPLING:
449

450 Staple: The process of binding a set of sheets together using a 'U'
451 shaped piece of metal wire that is punched through the set. The ends
452 of the metal staple are then bent over, or 'clinched' to hold the
453 staple in place. Technically the term 'stapler' refers to devices
454 that use pre-cut metal staples, but the term is also commonly used
455 to refer to devices that use wire spools and then cut/form the
456 staple. (see the definition of Stitch)
457

458 Stitch: The process of binding a set of sheets together using a 'U'
459 shaped piece of metal wire that is punched through the set. The
460 wire used to form the staple is cut and formed into a 'U' shape in
461 the stitcher head, and the staple 'leg' length is often varied
462 depending on the number of sheets to be bound together. The ends of
463
464

466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523

the metal staple are bent over, or 'clinched' to hold the staple in place.

Stitching can also refer to the process of sewing the edges of the signatures of a book together.

Saddle Stitch: The process of stapling a set along its center line as part of a booklet making process. Usually 2 or 3 staples are used.

Dual Stapling: The process of placing 2 staples along the bind edge of a set. The staples are typically located at 25% and 75% of the length of the bind edge. Although dual stapling is often performed on the long edge of a set, legal documents are frequently dual stapled along the top, or short edge of the set.

Triple Stapling: Same as above, but using 3 staples along the bind edge, and usually applies to the long edge only.

WRAPPING:

Shrink Wrap: A wrap of thin plastic which when heated will shrink and wrap tightly around the stack thus preparing it for shipment.

BANDING:

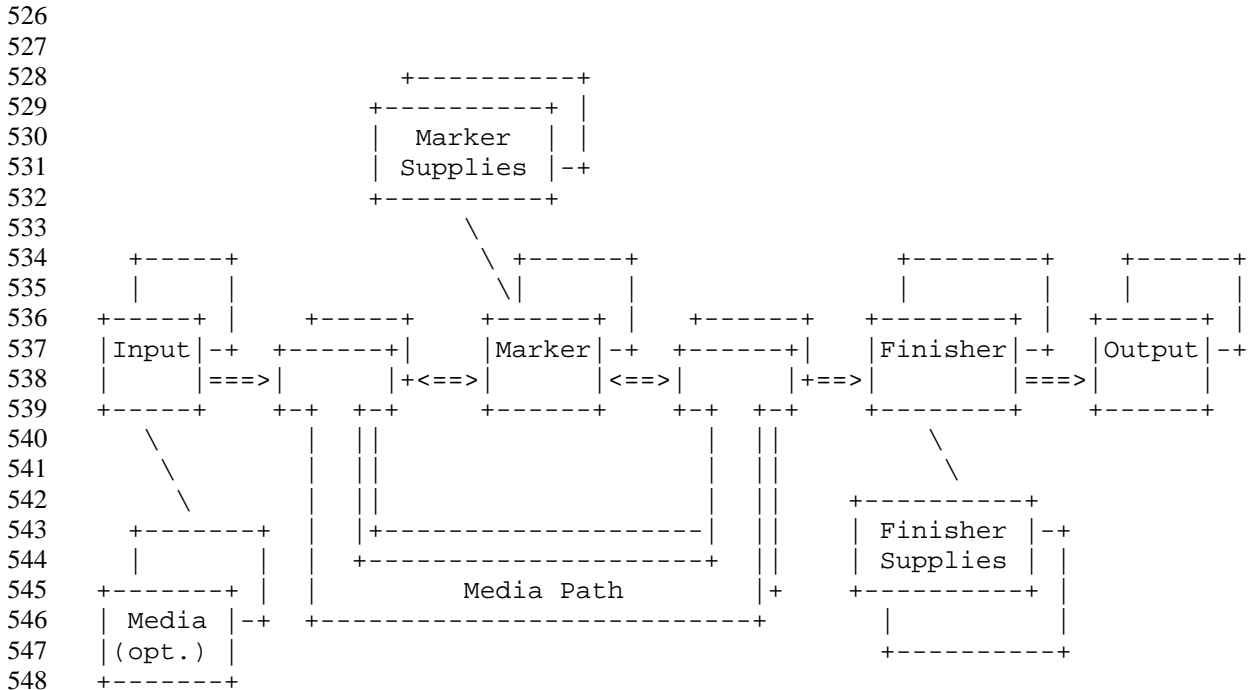
Band Wrap: Bundling a finished stack to prepare for shipment. Also known as Strap Wrap.

ROTATING:

Sheet Rotator: A device that rotates each sheet as received from the Media Path to the proper orientation for the finisher processing.

3. FINISHER SUBUNITS INTEGRATED INTO THE PRINTER MODEL

The Printer Finisher Device subunits receive media from one or more Printer Media Path subunits and deliver the media to one or more Printer Output subunits after the completion of the finishing processes. The Printer Model, as described in the Printer MIB [PRTMIB], is modified adding the finisher subunit(s) and finisher supplies between the media path and output subunits as follows:



553 4. FINISHING SPECIFICATIONS

554 The Finisher MIB is able to provide most of the information that is
 555 required to generate a Finishing Specification. This includes;

- 558 1. Finishing operations that can be performed on media that are
 559 associated with a specific printer media path and output subunit.
- 561 2. Combinations of operations that cannot be performed.
- 563 3. The location of the operation on the medium, if applicable.
- 565 4. The physical characteristics of the result of the operation.
 566 For example, the size and shape of a punched hole, or if a fold
 567 operation creates a letter fold or a "Z" fold.

569 The Finisher MIB permits an agent to describe the order that
 570 operations can be performed.

572 4.1 Multiple finDeviceTable Entries

574 Each finishing operation supported by the printer is represented by
 575 one or more entries in the finDeviceTable. Each entry in this table
 576 defines a "logical" finishing device, since the function of several
 577 table entries may be performed by a single finisher mechanism.
 578 Multiple entries may also exist in the table as a result of the

586 existence of multiple finisher mechanisms that perform the same type
 587 of operation.

588
 589 One example of possible multiple entries for a single finisher
 590 device, is a hole punch operation that creates more than one hole.
 591 This could be performed using a single die punch that moves to each
 592 required position or a multi-die punch that simultaneously creates
 593 all holes. In either case, each defined hole position may be defined
 594 as a separate table entry.

595
 596 In both cases, if the punch positions can be individually selected,
 597 a table entry for each position would be necessary.

598
 599 For the multi-die punch, each head mechanism may have a different
 600 hole pattern or size. If these differences are to be properly
 601 disclosed, a table entry for each head mechanism would be required.

602
 603 4.2 Implicit Parameters

604
 605 Finishing operations that are specified by an enum define a
 606 standard operation and in many cases an implicit set of physical
 607 characteristics is to be included when specifying the enum. If
 608 explicit values for these characteristics are not provided in the
 609 attributes table, the values defined in this section are to be
 610 implied.

611
 612 4.2.1 FinPunchPatternTC

enum pattern	Reference Edge	Reference Axis Offset	Hole spacing (see note 1)
twoHoleUSTop(4)	topEdge	note 2	2.75 inches
threeHoleUS(5)	note 3	note 2	4.25 inches
twoHoleDIN(6)	note 4	note 5	80 mm
fourHoleDIN(7)	note 4	note 5	80 mm
twentyTwoHoleUS(8)	note 3	note 2	.5 inches
nineteenHoleUS(9)	note 3	note 9	.5625 inches
twoHoleMetric(10)	note 6	note 8	80 mm
swedish4Hole(11)	note 4	note 5	21, 70, 21 mm
twoHoleUSSide(12)	note 3	note 2	2.75 inches
fiveHoleUS(13)	note 3	note 2	2, 2.25, 2.25, 2 in
sevenHoleUS(14)	note 3	note 2	1, 1, 2.25, 2.25, 1, 1 in
mixed7H4S(15)	note 4	note 5	note 7
norweg6Hole(16)	note 4	note 5	note 8
metric26Hole(17)	note 6	note 5	9.5 mm
metric30Hole(18)	note 4	note 5	9.5 mm

- 613
 614
 615
 616
 617
 618
 619
 620
 621
 622
 623
 624
 625
 626
 627
 628
 629
 630
 631
 632
 633 Notes:
- 634 1. All hole to hole patterns are centered along the process edge.
 - 635 2. Offset is 0.18 inches to 0.51 inches.
 - 636 3. Reference edge is leftEdge(5) for letter and topEdge(3)

644 for ledger.

- 645 4. Reference edge is leftEdge(5) for A4 and topEdge(3) for A3.
- 646 5. Offset is 4.5 mm to 13 mm.
- 647 6. Reference edge is leftEdge(5) for B5 and topEdge(3) for B4.
- 648 7. 7 holes and 4 slots are punched in a H-S-H-H-S-H-S-H-H-S-H
- 649 pattern with 15, 25, 23, 20, 37, 37, 20, 23, 25, 15 mm spacing.
- 650 8. 4 holes and 2 slots are punched in a H-H-S-S-H-H pattern with
- 651 a 64, 18.5, 75, 18.5, 64 mm spacing.
- 652 9. Offset is .188 inches.

653
 654
 655 4.2.2 FinPunchHoleTypeTC, punchHoleSizeMaxDim, punchHoleSizeMinDim
 656

657 enum pattern	658 Hole Description
659 twoHoleUSTop(4)	659 round(3), .2 - .32 inch diameter
660 threeHoleUS(5)	660 round(3), .2 - .32 inch diameter
661 twoHoleDIN(6)	661 round(3), 5 - 8 mm diameter
662 fourHoleDIN(7)	662 round(3), 5 - 8 mm diameter
663 twentyTwoHoleUS(8)	663 round(3), .2 - .32 inch diameter
664 nineteenHoleUS(9)	664 rectang(6), .313 inches X .125 inches
665 twoHoleMetric(10)	665 round(3), 5 - 8 mm diameter
666 swedish4Hole(11)	666 round(3), 5 - 8 mm diameter
667 twoHoleUSSide(12)	667 round(3), .2 - .32 inch diameter
668 fiveHoleUS(13)	668 round(3), .2 - .32 inch diameter
669 sevenHoleUS(14)	669 round(3), .2 - .32 inch diameter
670 mixed7H4S(15)	670 round(3), 5 - 8 mm diameter
671	671 rectang(6), 12 mm X 6 mm
672 norweg6Hole(16)	672 round(3), 5 - 8 mm diameter
673	673 rectang(6), 10 mm X 5.5 mm
674 metric26Hole(17)	674 round(3), 5 - 8 mm
675 metric30Hole(18)	675 round(3), 5 - 8 mm

676
 677 Note: Hole size ranges are typical and are provided as a reference
 678 only. Exact tolerances should be site defined.
 679

680
 681
 682 5. THE ATTRIBUTE MECHANISM
 683

684 Attributes provide a function similar to information objects, except
 685 that attributes are identified by an enum, instead of an OID. Thus
 686 new attributes may be registered without requiring a change to the
 687 MIB. In addition, an implementation that does not have the
 688 functionality represented by the attribute can omit the attribute
 689 entirely, rather than having to return a distinguished value. The
 690 agent is free to materialize an attribute in the Attribute Table as
 691 soon as the agent is aware of the value of the attribute.
 692

693 The agent materializes finishing subunit attributes in a four-indexed
 694 finDeviceAttributeTable:
 695

- 701
- 702
- 703 1. hrDeviceIndex - which device in the host
- 704
- 705 2. finDeviceIndex - which finisher subunit in the printer device
- 706
- 707 3. finDeviceAttributeTypeIndex - which attribute
- 708
- 709 4. finDeviceAttributeInstanceIndex - which attribute instance for
- 710 those attributes that can have multiple values per finishing
- 711 subunit.
- 712

713

714 5.1 Conformance of Attribute Implementation

715

716 An agent SHALL implement any attribute if (1) the device supports the
717 functionality represented by the attribute and (2) the information is
718 available to the agent. The agent MAY create the attribute row in
719 the finDeviceAttributeTable when the information is available or MAY
720 create the row earlier with the designated 'unknown' value
721 appropriate for that attribute. See next section.

722

723 If the device does not implement or does not provide access to the
724 information about an attribute, the agent SHOULD NOT create the
725 corresponding row in the finDeviceAttributeTable.

726

727

728 5.2 Useful, 'Unknown', and 'Other' Values for Objects and Attributes

729

730 Some attributes have a 'useful' Integer32 value, some have a 'useful'
731 OCTET STRING value, some MAY have either or both depending on
732 implementation, and some MUST have both. See the
733 finDeviceAttributeTypeTC textual convention for the specification of
734 each attribute.

735

736 SNMP requires that if an object cannot be implemented because its
737 values cannot be accessed, then a compliant agent SHALL return an
738 SNMP error in SNMPv1 or an exception value in SNMPv2. However, this
739 MIB has been designed so that 'all' objects can and SHALL be
740 implemented by an agent, so that neither the SNMPv1 error nor the
741 SNMPv2 exception value SHALL be generated by the agent. This MIB has
742 also been designed so that when an agent materializes an attribute,
743 the agent SHALL materialize a row consisting of both the
744 finDeviceAttributeValueAsInteger and
745 finDeviceAttributeValueAsOctets objects.

746

747 In general, values for objects and attributes have been chosen so
748 that a management application will be able to determine whether a
749 'useful', 'unknown', or 'other' value is available. When a useful
750 value is not available for an object that agent SHALL return a zero-
751 length string for octet strings, the value 'unknown(2)' for enums, a
752 '0' value for an object that represents an index in another table,

760 and a value '-2' for counting integers.

761
762 Since each attribute is represented by a row consisting of both the
763 finDeviceAttributeValueAsInteger and finDeviceAttributeValueAsOctets
764 MANDATORY objects, SNMP requires that the agent SHALL always create
765 an attribute row with both objects specified. However, for most
766 attributes the agent SHALL return a "useful" value for one of the
767 objects and SHALL return the 'other' value for the other object. For
768 integer only attributes, the agent SHALL always return a zero-length
769 string value for the finDeviceAttributeValueAsOctets object. For
770 octet string only attributes, the agent SHALL always return a '-1'
771 value for the finDeviceAttributeValueAsInteger object.
772
773

774 5.3 Data Sub-types and Attribute Naming Conventions

775
776 Many attributes are sub-typed to give a more specific data type than
777 Integer32 or OCTET STRING. The data sub-type of each attribute is
778 indicated on the first line(s) of the description. Some attributes
779 have several different data sub-type representations. When an
780 attribute has both an Integer32 data sub-type and an OCTET STRING
781 data sub-type, the attribute can be represented in a single row in
782 the finDeviceAttributeTable. In this case, the data sub-type name is
783 not included as the last part of the name of the attribute. When the
784 data sub-types cannot be represented by a single row in the
785 finDeviceAttributeTable, each such representation is considered a
786 separate attribute and is assigned a separate name and enum value.
787 For these attributes, the name of the data sub-type is the last part
788 of the name of the attribute.
789
790

791 5.4 Single-Value (Row) Versus Multi-Value (MULTI-ROW) Attributes

792
793 Most attributes SHALL have only one row per finishing subunit.
794 However, a few attributes can have multiple values per finishing
795 subunit, where each value is a separate row in the
796 finDeviceAttributeTable. Unless indicated with 'MULTI-ROW:' in the
797 finDeviceAttributeTypeTC description, an agent SHALL ensure that each
798 attribute occurs only once in the finDeviceAttributeTable for a
799 finishing subunit. Most of the 'MULTI-ROW' attributes do not allow
800 duplicate values, i.e., the agent SHALL ensure that each value occurs
801 only once for a finishing subunit. Only if the specification of the
802 'MULTI-ROW' attribute also says "There is no restriction on the same
803 xxx occurring in multiple rows" can the agent allow duplicate values
804 to occur for a single finishing subunit.
805
806

807 5.5 Linked MUTI-ROW Values

808
809 Some MULTI-ROW attributes are intended to go together. Thus a set
810 of value instances represent a single instance. For example, the
811
812

817
818 puncher attributes indicate the location, maximum size, minimum size
819 and shape of the various holes that the puncher can produce. So the
820 first set of values could represent one kind of hole, and the second
821 set another kind of hole, etc.
822

823
824 5.6 Index Value Attributes

825
826 A number of attributes are indexes in other tables. Such attribute
827 names end with the word 'Index'. If the agent has not (yet) assigned
828 an index value for a particular index attribute for a finishing
829 subunit, the agent SHALL either: (1) return the value 0 or (2) not
830 add this attribute to the finDeviceAttributeTable until the index
831 value is assigned. In the interests of brevity, the semantics for 0
832 is specified once here and is not repeated for each index attribute
833 specification and a DEFVAL of 0 is indicated.
834

835
836 5.7 Attribute Specifications

837
838 This section specifies the set of attributes that are enumerated
839 in finAttributeTypeTC. The data type tag definitions 'INTEGER:'
840 or 'OCTETS', indicate if the attribute can be represented using
841 the object finDeviceAttributeAsInteger or the object
842 finDeviceAttributeAsOoctets, respectively. In some cases, a choice
843 between the two data types is possible and for a few attributes both
844 objects may be required at the same time to properly present the
845 value.
846

847 NOTE - The enum assignments are grouped logically with values
848 assigned in groups of 10, so that additional values may be
849 registered in the future and assigned a value that is part of
850 their logical grouping.
851

852 Values in the range 2**30 to 2**31-1 are reserved for private or
853 experimental usage. This range corresponds to the same range
854 reserved in IPP. Implementers are warned that use of such values
855 may conflict with other implementations. Implementers are encouraged
856 to request registration of enum values following the procedures in
857 Section 6.1.
858

859 The attribute types defined at the time of completion of this
860 specification are:

861

finAttributeTypeIndex	Data type
-----	-----
other(1),	Integer32
	AND/OR
	OCTET STRING (SIZE(0..63))
INTEGER: and/or OCTETS:	An attribute that is not currently
approved and registered.	

867
868
869
870

872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929

A. Generic finisher subunit attributes that apply to all finisher subunit types. (3..)

deviceName(3), OCTET STRING (SIZE(0..63))
OCTETS: The name assigned to this finisher device subunit.

deviceVendorName(4), OCTET STRING (SIZE(0..63))
OCTETS: The name of the vendor of this finisher device subunit.

deviceModel(5), OCTET STRING (SIZE(0..63))
OCTETS: The model name of this finisher device subunit.

deviceVersion(6), OCTET STRING (SIZE(0..63))
OCTETS: The version string for this finisher device subunit.

deviceSerialNumber(7), OCTET STRING (SIZE(0..63))
OCTETS: The serial number assigned to this finisher device subunit.

maximumSheets(8), Integer32 (-2..32767)
INTEGER: Defines the maximum number of media sheets that a finisher device is able to process.

finProcessOffsetUnits(9), PrtMediaUnitTC
INTEGER: An enumeration which defines the units of measure for the attributes finAxisOffset and finHeadLocation.

finReferenceEdge(10), FinEdgeTC
INTEGER: An enumeration which defines which edge of the form is the reference for this finishing process. The Finishing Process Axis will be parallel to this axis.

finAxisOffset(11), Integer32 (-2..2147483647)
INTEGER: Defines the parallel offset of the Finishing Process Axis from the Reference Edge. For a value of finEdgeTC equal to TopEdge or RightEdge, the value given is to interpreted as a negative offset from the reference edge. The units of measure are defined by the attribute finReferenceEdgeTC.

finJogEdge(12), FinEdgeTC
INTEGER: An enumeration which defines a second edge of the document to which the media is aligned. The jog edge must be perpendicular to the edge defined by finReferenceEdge.

finHeadLocation(13), Integer32 (-2..2147483647)
INTEGER: MULTI-ROW: Defines the position of the Head Mechanism relative to the axis, 'X' or 'Y', that is

930
931 INTERNET-DRAFT Printer Finishing MIB May 9, 2000
932
933
934 perpendicular to the Process Axis. The units of measure
935 are defined by the attribute finProcessOffsetUnits.
936
937 finOperationRestrictions(14), Integer32 (0..65535)
938 INTEGER: MULTI-ROW: Defines the finDeviceIndex of a
939 finishing process which cannot be combined with the
940 process defined by the finDeviceIndex for this
941 finDeviceAttributeTable instance. When this condition
942 occurs this attribute SHALL be presented in the
943 attribute tables for both finishing processes that cannot
944 be combined.
945
946 finNumberOfPositions(15), Integer32 (0..65535)
947 INTEGER: Defines the total number of head positions for
948 this finishing process. Each position may be realized by
949 a unique head mechanism or a single head mechanism may be
950 automatically moved to each position.
951
952 namedConfiguration(16), OCTET STRING (SIZE(0..63))
953 OCTETS: Contains an administratively define name to define
954 the finishing specification configured for this device.
955
956 finMediaTypeRestriction(17), OCTET STRING (SIZE(0..63))
957 OCTETS: MULTI-ROW: Defines the media type which cannot be
958 combined with the process defined by the finDeviceIndex
959 for this finDeviceAttributeTable instance. Values are the
960 same as defined for finSupplyMediaInputMediaName.
961
962 finPrinterInputTraySupported(18), Integer32 (0..65535)
963 INTEGER: MULTI-ROW: Defines the value of prtInputIndex
964 corresponding to the printer input tray that can be used
965 with the process defined by the finDeviceIndex for this
966 finDeviceAttributeTable instance. If this attribute is
967 not present, this process can be used with any input tray
968 in the printer. For example, this attribute can indicate
969 the current stapling capabilities for a stapler device
970 for the input trays that depend upon the size and feed
971 orientation. So if there were two letter trays, one with
972 A size and the other with B size, a two position stapler
973 might specify in one row: upper-left and upper-right for
974 the input tray with A size, but only upper-left for the
975 one with B size.
976
977 finPreviousFinishingOperation(19), Integer32 (0..65535)
978 INTEGER: Defines the finDeviceIndex of the previous
979 finishing process for implementations in which the
980 finishing processes are performed in a prescribed order.
981 Each finishing process in the fixed sequence is either
982 performed or not performed according to the finishing
983 instructions submitted with the job. A value of 0
984 indicates that this finishing process is the first in a
985
986
987

988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045

sequence. Finishing processes which are not part of a fixed sequence SHALL NOT have this attribute.

finNextFinishingOperation(20), Integer32 (0..65535)
INTEGER: Defines the finDeviceIndex of the next finishing process for implementations in which the finishing processes are performed in a prescribed order. Each finishing process in the fixed sequence is either performed or not performed according to the finishing instructions submitted with the job. A value of 0 indicates that this finishing process is the last in a sequence. Finishing processes which are not part of a fixed sequence SHALL NOT have this attribute.

B. Stitcher type-specific attributes (30..)

stitchingType(30), FinStitchingTypeTC
INTEGER: MULTI-ROW: Provides additional information regarding the stitching operation.

stitchingDirection(31), StitchingDirTypeTC
INTEGER: Defines the orientation of the stitching process.

stitchingAngle(32), StitchingAngleTypeTC
INTEGER: Defines enumerations that describe the angular orientation of the stitching process relative to the 'X' axis.

C. Folder type-specific attributes (40..)

foldingType(40), FinFoldingTypeTC
INTEGER: Provides additional information regarding the folding process.

D. Binder type-specific attributes (50..)

bindingType(50), FinBindingTypeTC
INTEGER: Provides additional information regarding the binding process.

E. Trimmer type-specific attributes (60..)

F. Die cutter type-specific attributes (70..)

G. Puncher type-specific attributes (80..)

punchHoleType(80), FinPunchHoleTypeTC
INTEGER: Provides information regarding the shape of the punched hole.

1046
1047 INTERNET-DRAFT Printer Finishing MIB May 9, 2000
1048
1049
1050 punchHoleSizeLongDim(81), Integer32 (-2..2147483647)
1051 INTEGER: Defines the size of the punched hole in the
1052 longest dimension. This dimension is typically measured
1053 parallel to either the long edge or the short edge of the
1054 media and the longest dimension will always be measured 90
1055 degrees from the shortest dimension. For a symmetrical
1056 hole, such as a round or square hole, the shortest and
1057 longest dimensions will be identical.
1058
1059 punchHoleSizeShortDim(82), Integer32 (-2..2147483647)
1060 INTEGER: Defines the size of the punched hole in the
1061 shortest dimension. This dimension is typically measured
1062 parallel to either the long edge or the short edge of the
1063 media and the shortest dimension will always be measured
1064 90 degrees from the longest dimension. For a symmetrical
1065 hole, such as a round or square hole, the shortest and
1066 longest dimensions will be identical.
1067
1068 punchPattern(83), FinPunchPatternTC
1069 INTEGER: Defines the hole pattern produced by the punch
1070 process.
1071
1072 H. Perforator type-specific attributes (90..)
1073
1074 I. Slitter type-specific attributes (100..)
1075
1076 slittingType(100), FinSlittingTypeTC
1077 INTEGER: Provides additional information regarding the
1078 slitting process.
1079
1080 J. Separation cutter type-specific attributes (110..)
1081
1082 K. Imprinter type-specific attributes (120..)
1083
1084 L. Wrapper type-specific attributes (130..)
1085
1086 wrappingType(130), FinWrappingTypeTC
1087 INTEGER: Provides additional information regarding the
1088 wrapping process.
1089
1090 M. Bander type-specific attributes (140..)
1091
1092 N. Make Envelopes type-specific attributes (150..)
1093
1094 O. Stacker type-specific attributes (160..)
1095
1096 stackOutputType(160) FinStackOutputTypeTC
1097 INTEGER: Defines the job-to-job orientation produced by
1098 the stacker.
1099
1100
1101
1102
1103 Bergman, Lewis [page 19]

1107
1108 stackOffset(161) Integer32 (-2..2147483647)
1109 INTEGER: Defines the copy-to-copy output stack offset as
1110 a positive offset distance. The units of measure are
1111 defined by finProcessOffsetUnits.

1112
1113 stackRotation(162) Integer32 (-2..180)
1114 INTEGER: Defines the copy-to-copy output stack rotation
1115 measured in degrees. The value is the positive
1116 copy-to-copy rotation."

1117
1118
1119
1120 6. ENUMERATIONS

1121
1122 Enumerations (enums) are sets of symbolic values defined for use
1123 with one or more objects. Some common enumeration sets are
1124 assigned a symbolic data type name (textual convention). These
1125 enumerations are listed at the beginning of this specification.

1126
1127
1128 6.1 Registering Additional Enumerated Values

1129
1130 The Printer Working Group (PWG) has defined several type of
1131 enumerations. These enumerations differ in the method employed
1132 to control the addition of new enumerations. Throughout this
1133 document, references to "enumeration (n)", where n can be 1, 2 or
1134 3 can be found in the various tables. The definitions of these
1135 types of enumerations are:

1136
1137 enumeration (1) All the values are defined in the Printer Finisher
1138 MIB specification (RFC for the Printer Finisher MIB). Additional
1139 enumerated values require a new RFC. Type 1 enumerations are
1140 typically used where changes to the enumeration are either unlikely
1141 or will have a significant impact on the structure of the MIB or
1142 implementation of the MIB in management applications.

1143
1144 Some criteria that suggest using a type 1 enumeration are:

- 1145
1146 a) the set of values in the enumeration is thought to be known,
1147 e.g., faceUp and faceDown
1148
1149 b) the enumeration defines a set of units of measure which must
1150 be understood by a management application to be able to
1151 correctly display the value of an object that measurement unit
1152 controls;

1153
1154 and

- 1155
1156 c) the enumeration is tied to the structure of the MIB or the
1157 model on which the MIB is based, i.e., the enumeration is tied

1166 to the OIDs for the related tables.
1167

1168 There are no type 1 enums in the current draft.
1169
1170

1171 enumeration (2) An initial set of values are defined in the
1172 Printer Finisher MIB specification. Additional enumerated values
1173 are registered after review by this working group. The initial
1174 versions of the MIB will contain the values registered so far.
1175

1176 After the MIB is approved, additional values will be registered
1177 through IANA after approval by this working group. The current
1178 set of approved values should always be obtained from the IANA
1179 registry. Type 2 enumerations are typically used where it is
1180 important to ensure consistent usage of the enumeration values;
1181 that is, to ensure that the same entity does not get two
1182 different enumerations values, or two different entities do not
1183 get the same enum value.
1184
1185

1186 enumeration (3) An initial set of values are defined in the
1187 Printer Finisher MIB specification. Additional enumerated values
1188 are registered without working group review. The initial versions
1189 of the MIB will contain the values registered so far. After the MIB
1190 is approved, additional values will be registered through IANA
1191 without approval by this working group. The current set of
1192 approved values should always be obtained from the IANA registry.
1193

1194 Type 3 enumerations are used for enumerations that can be extended
1195 without any controls; an example is the prtMarkerSuppliesType, which
1196 can be extended as needed by any manufacturer to describe the
1197 supplies required by a new printer.
1198
1199

1200 7. PRINTER FINISHING MIB SPECIFICATION
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219

```

1220
1221 INTERNET-DRAFT          Printer Finishing MIB          May 9, 2000
1222
1223
1224 Finisher-MIB DEFINITIONS ::= BEGIN
1225
1226 IMPORTS
1227     mib-2                                FROM RFC1213-MIB
1228     MODULE-IDENTITY, OBJECT-TYPE, Integer32,
1229     experimental                          FROM SNMPv2-SMI
1230     TEXTUAL-CONVENTION                    FROM SNMPv2-TC
1231     MODULE-COMPLIANCE, OBJECT-GROUP      FROM SNMPv2-CONF
1232     hrDeviceIndex                        FROM HOST-RESOURCES-MIB
1233     printmib, PrtMarkerSuppliesTypeTC,
1234     PrtSubUnitStatusTC, PrtInputTypeTC,
1235     PrtMarkerSuppliesSupplyUnitTC, PrtMediaUnitTC,
1236     PrtCapacityUnitTC, PrtMarkerSuppliesClassTC,
1237     PresentOnOff, prtMIBConformance     FROM Printer-MIB;
1238
1239 finisherMIB MODULE-IDENTITY
1240     LAST-UPDATED "9810090000Z"
1241     ORGANIZATION "IETF Printer MIB Working Group"
1242     CONTACT-INFO
1243         "Ron Bergman
1244         Dataproducts Corp.
1245         1757 Tapo Canyon Road
1246         Simi Valley, CA 91063-3394
1247         rbergma@dpc.com
1248
1249         Send comments to the printmib WG using the Finisher MIB
1250         Project (FIN) Mailing List:  fin@pwg.org
1251
1252         For further information, access the PWG web page under 'FIN':
1253         http://www.pwg.org/
1254
1255         Implementers of this specification are encouraged to join the
1256         fin mailing list in order to participate in discussions on any
1257         clarifications needed and registration proposals being reviewed
1258         in order to achieve consensus."
1259     DESCRIPTION
1260         "The MIB module for management of printer finisher units.
1261         The Finisher MIB is an extension of the Printer MIB."
1262     ::= { experimental 54 }
1263
1264     -- Note to RFC editor, experimental 54 was assigned to the Printer
1265     -- MIB.  A new arc in the experimental subtree must be assigned.
1266
1267
1268     -- Textual conventions for this MIB module
1269
1270
1271     FinDeviceTypeTC ::= TEXTUAL-CONVENTION
1272     -- This is a type 2 enumeration.
1273         STATUS      current
1274         DESCRIPTION
1275
1276
1277     Bergman, Lewis

```

```

1278
1279 INTERNET-DRAFT          Printer Finishing MIB          May 9, 2000
1280
1281
1282     "The defined finishing device subunit process
1283     enumerations."
1284     SYNTAX          INTEGER {
1285         other(1),
1286         unknown(2),
1287         stitcher(3),
1288         folder(4),
1289         binder(5),
1290         trimmer(6),
1291         dieCutter(7),
1292         puncher(8),
1293         perforater(9),
1294         slitter(10),
1295         separationCutter(11),
1296         imprinter(12),
1297         wrapper(13),
1298         bander(14),
1299         makeEnvelope(15),
1300         stacker(16),
1301         sheetRotator(17)
1302     }
1303
1304 FinAttributeTypeTC ::= TEXTUAL-CONVENTION
1305 -- This is a type 3 enumeration.
1306     STATUS          current
1307     DESCRIPTION
1308         "This textual convention defines the set of enums for use in
1309         the finDeviceAttributeTable. See section 5.7 for the complete
1310         specification of each attribute."
1311     SYNTAX          INTEGER {
1312         other(1),
1313         deviceName(3),
1314         deviceVendorName(4),
1315         deviceModel(5),
1316         deviceVersion(6),
1317         deviceSerialNumber(7),
1318         maximumSheets(8),
1319         finProcessOffsetUnits(9),
1320         finReferenceEdge(10),
1321         finAxisOffset(11),
1322         finJogEdge(12),
1323         finHeadLocation(13),
1324         finOperationRestrictions(14),
1325         finNumberOfPositions(15),
1326         namedConfiguration(16),
1327         finMediaTypeRestriction(17),
1328         finPrinterInputTraySupported(18),
1329         finPreviousFinishingOperation(19),
1330         finNextFinishingOperation(20),
1331         stitchingType(30),
1332         stitchingDirection(31),
1333
1334
1335 Bergman, Lewis

```

```

1336
1337 INTERNET-DRAFT          Printer Finishing MIB          May 9, 2000
1338
1339
1340         foldingType(40),
1341         bindingType(50),
1342         punchHoleType(80),
1343         punchHoleSizeLongDim(81),
1344         punchHoleSizeShortDim(82),
1345         punchPattern(83),
1346         slittingType(100),
1347         wrappingType(130),
1348         stackOutputType(160),
1349         stackOffset(161),
1350         stackRotation(162)
1351     }
1352
1353 FinEdgeTC ::= TEXTUAL-CONVENTION
1354 -- This is a type 2 enumeration.
1355     STATUS          current
1356     DESCRIPTION
1357         "Specifies an edge for a Finishing Process."
1358     SYNTAX          INTEGER {
1359         topEdge(3),
1360         bottomEdge(4),
1361         leftEdge(5),
1362         rightEdge(6)
1363     }
1364
1365 FinStitchingTypeTC ::= TEXTUAL-CONVENTION
1366 -- This is a type 2 enumeration.
1367     STATUS          current
1368     DESCRIPTION
1369         "The defined stitching type enumerations.  For the edgeStitch and
1370         stapleDual enums, the finReferenceEdge attribute is recommended
1371         to define the edge to which the operation applies."
1372     SYNTAX          INTEGER {
1373         other(1),          -- More information in other attributes
1374         unknown(2),
1375         stapleTopLeft(4),
1376         stapleBottomLeft(5),
1377         stapleTopRight(6),
1378         stapleBottomRight(7),
1379         saddleStitch(8),
1380         edgeStitch(9),
1381         stapleDual(10)
1382     }
1383
1384 StitchingDirTypeTC ::= TEXTUAL-CONVENTION
1385 -- This is a type 2 enumeration.
1386     STATUS          current
1387     DESCRIPTION
1388         "Defines the direction, relative to the top sheet in the output
1389         subunit, that the stitching operation was performed.  For a
1390         topDown(3) process, the staple will be clinched on the bottom
1391
1392
1393 Bergman, Lewis

```



```

1394
1395 INTERNET-DRAFT          Printer Finishing MIB          May 9, 2000
1396
1397
1398         of the stack.  This parameter can be used to determine what
1399         order the pages of a booklet are to be printed such that the
1400         staple clinch will be on the inside of the resulting booklet."
1401     SYNTAX          INTEGER {
1402         unknown(2),
1403         topDown(3),
1404         bottomUp(4)
1405     }
1406
1407     StitchingAngleTypeTC ::= TEXTUAL-CONVENTION
1408     -- This is a type 2 enumeration.
1409     STATUS          current
1410     DESCRIPTION
1411     "This enumeration provides a description of the angular
1412     orientation of each stitch in a single or multiple stitching
1413     operation, relative to the 'X' axis.  As with all finishing
1414     operations, the 'X' axis is always relative to the portrait
1415     orientation of the document regardless of the orientation
1416     of the printed image.  This enum is primarily applicable to
1417     corner stitching operations."
1418     SYNTAX          INTEGER {
1419         unknown(2),
1420         horizontal(3),
1421         vertical(4),
1422         slanted(5)
1423     }
1424
1425     FinFoldingTypeTC ::= TEXTUAL-CONVENTION
1426     --This is a type 2 enumeration.
1427     STATUS          current
1428     DESCRIPTION
1429     "The defined folding device process enumerations."
1430     SYNTAX          INTEGER {
1431         other(1),          -- More information in other attributes
1432         unknown(2),
1433         zFold(3),
1434         halfFold(4),
1435         letterFold(5)
1436     }
1437
1438     FinBindingTypeTC ::= TEXTUAL-CONVENTION
1439     -- This is a type 2 enumeration.
1440     STATUS          current
1441     DESCRIPTION
1442     "The defined binding type enumerations."
1443     SYNTAX          INTEGER {
1444         other(1),          -- More information in other attributes
1445         unknown(2),
1446         tape(4),
1447         plastic(5),
1448         velo(6),
1449
1450
1451     Bergman, Lewis

```

```

1452
1453 INTERNET-DRAFT          Printer Finishing MIB          May 9, 2000
1454
1455
1456         perfect(7),
1457         spiral(8),
1458         adhesive(9),
1459         comb(10),
1460         padding(11)
1461     }
1462
1463 FinPunchHoleTypeTC ::= TEXTUAL-CONVENTION
1464 --This is a type 2 enumeration.
1465     STATUS          current
1466     DESCRIPTION
1467     "The defined hole type punch process enumerations."
1468     SYNTAX          INTEGER {
1469         other(1),          -- More information in other attributes
1470         unknown(2),
1471         round(3),
1472         oblong(4),
1473         square(5),
1474         rectangular(6),
1475         star(7)
1476     }
1477
1478 FinPunchPatternTC ::= TEXTUAL-CONVENTION
1479 --This is a type 2 enumeration.
1480     STATUS          current
1481     DESCRIPTION
1482     "The defined hole pattern punch process enumerations."
1483     SYNTAX          INTEGER {
1484         other(1),          --Pattern to be defined in other attributes
1485         unknown(2),
1486         twoHoleUSTop(4),   --Letter/legal, 8.5 inch edge
1487         threeHoleUS(5),    --Letter/ledger, 11 inch edge
1488         twoHoleDIN(6),     --A4/A3, 297 mm edge
1489         fourHoleDIN(7),    --A4/A3, 297 mm edge
1490         twentyTwoHoleUS(8), --Letter/ledger, 11 inch edge
1491         nineteenHoleUS(9), --Letter/ledger, 11 inch edge
1492         twoHoleMetric(10), --B5/B4, 257 mm edge
1493         swedish4Hole(11),  --A4/A3, 297 mm edge
1494         twoHoleUSSide(12), --Letter/ledger, 11 inch edge
1495         fiveHoleUS(13),    --Letter/ledger, 11 inch edge
1496         sevenHoleUS(14),   --Letter/ledger, 11 inch edge
1497         mixed7H4S(15),     --A4/A3, 297 mm edge
1498         norweg6Hole(16),   --A4/A3, 297 mm edge
1499         metric26Hole(17),  --B5/B4, 257 mm edge
1500         metric30Hole(18),  --A4/A3, 297 mm edge
1501     }
1502
1503 FinSlittingTypeTC ::= TEXTUAL-CONVENTION
1504 -- This is a type 2 enumeration.
1505     STATUS          current
1506     DESCRIPTION
1507
1508
1509 Bergman, Lewis

```

```

1510
1511 INTERNET-DRAFT          Printer Finishing MIB          May 9, 2000
1512
1513
1514     "The defined slitting type enumerations."
1515     SYNTAX      INTEGER {
1516         other(1),          -- More information in other attributes
1517         unknown(2),
1518         slitAndSeparate(4),
1519         slitAndMerge(5)
1520     }
1521
1522 FinWrappingTypeTC ::= TEXTUAL-CONVENTION
1523 --This is a type 2 enumeration.
1524     STATUS      current
1525     DESCRIPTION
1526     "The defined wrapping device process enumerations."
1527     SYNTAX      INTEGER {
1528         other(1),          -- More information in other attributes
1529         unknown(2),
1530         shrinkWrap(4),
1531         paperWrap(5)
1532     }
1533
1534 FinStackOutputTypeTC ::= TEXTUAL-CONVENTION
1535 --This is a type 2 enumeration.
1536     STATUS      current
1537     DESCRIPTION
1538     "The defined stack output type enumerations."
1539     SYNTAX      INTEGER {
1540         other(1),          -- More information in other attributes
1541         unknown(2),
1542         straight(4),       -- No offset, one on top of another
1543         offset(5),
1544         crissCross(6)     -- Rotated
1545     }
1546
1547
1548 -- Finisher Device Group (Mandatory)
1549 --
1550 -- A printer may support zero or more finishing subunits.  A
1551 -- finishing device subunit may be associated with one or more
1552 -- output subunits and one or more media path subunits.
1553
1554 finDevice OBJECT IDENTIFIER ::= { printmib 30 }
1555
1556 finDeviceTable OBJECT-TYPE
1557     SYNTAX      SEQUENCE OF FinDeviceEntry
1558     MAX-ACCESS  not-accessible
1559     STATUS      current
1560     DESCRIPTION
1561     "This table defines the finishing device subunits,
1562     including information regarding possible configuration
1563     options and the status for each finisher device subunit."
1564     ::= { finDevice 1 }
1565
1566
1567 Bergman, Lewis          [page 27]

```

```

1568
1569 INTERNET-DRAFT          Printer Finishing MIB          May 9, 2000
1570
1571
1572
1573 finDeviceEntry OBJECT-TYPE
1574     SYNTAX      FinDeviceEntry
1575     MAX-ACCESS  not-accessible
1576     STATUS      current
1577     DESCRIPTION
1578         "There is an entry in the finishing device table for each
1579         possible finisher process."
1580     INDEX { hrDeviceIndex, finDeviceIndex }
1581     ::= { finDeviceTable 1 }
1582
1583 FinDeviceEntry ::= SEQUENCE {
1584     finDeviceIndex      Integer32,
1585     finDeviceType       FinDeviceTypeTC,
1586     finDevicePresentOnOff PresentOnOff,
1587     finDeviceCapacityUnit PrtCapacityUnitTC,
1588     finDeviceMaxCapacity Integer32,
1589     finDeviceCurrentCapacity Integer32,
1590     finDeviceAssociatedMediaPaths OCTET STRING,
1591     finDeviceAssociatedOutputs  OCTET STRING,
1592     finDeviceStatus       PrtSubUnitStatusTC,
1593     finDeviceDescription  OCTET STRING
1594 }
1595
1596 finDeviceIndex OBJECT-TYPE
1597     SYNTAX      Integer32 (1..65535)
1598     MAX-ACCESS  not-accessible
1599     STATUS      current
1600     DESCRIPTION
1601         "A unique value used to identify a finisher process.
1602         Although these values may change due to a major
1603         reconfiguration of the printer system (e.g. the addition
1604         of new finishing processes), the values are normally
1605         expected to remain stable across successive power cycles."
1606     ::= { finDeviceEntry 1 }
1607
1608 finDeviceType OBJECT-TYPE
1609     SYNTAX      FinDeviceTypeTC
1610     MAX-ACCESS  read-only
1611     STATUS      current
1612     DESCRIPTION
1613         "Defines the type of finishing process associated with this
1614         table row entry."
1615     ::= { finDeviceEntry 2 }
1616
1617 finDevicePresentOnOff OBJECT-TYPE
1618     SYNTAX      PresentOnOff
1619     MAX-ACCESS  read-write
1620     STATUS      current
1621     DESCRIPTION
1622         "Indicates if this finishing device subunit is available
1623
1624
1625 Bergman, Lewis

```

```

1626
1627 INTERNET-DRAFT          Printer Finishing MIB          May 9, 2000
1628
1629
1630     and whether the device subunit is enabled."
1631     DEFVAL      { notPresent }
1632     ::= { finDeviceEntry 3 }
1633
1634 finDeviceCapacityUnit OBJECT-TYPE
1635     SYNTAX      PrtCapacityUnitTC
1636     MAX-ACCESS  read-only
1637     STATUS      current
1638     DESCRIPTION
1639         "The unit of measure for specifying the capacity of this
1640         finisher device subunit."
1641     ::= { finDeviceEntry 4 }
1642
1643 finDeviceMaxCapacity OBJECT-TYPE
1644     SYNTAX      Integer32 (-2..2147483647)
1645     MAX-ACCESS  read-write
1646     STATUS      current
1647     DESCRIPTION
1648         "The maximum capacity of this finisher device subunit in
1649         finDeviceCapacityUnits.  If the device can reliably sense
1650         this value, the value is sensed by the finisher device
1651         and is read-only: otherwise the value may be written by a
1652         management or control console application.  The value (-1)
1653         means other and specifically indicates that the device
1654         places no restrictions on this parameter.  The value (-2)
1655         means unknown."
1656     DEFVAL      { -2 }          -- unknown
1657     ::= { finDeviceEntry 5 }
1658
1659 finDeviceCurrentCapacity OBJECT-TYPE
1660     SYNTAX      Integer32 (-2..2147483647)
1661     MAX-ACCESS  read-write
1662     STATUS      current
1663     DESCRIPTION
1664         "The current capacity of this finisher device subunit in
1665         finDeviceCapacityUnits.  If the device can reliably sense
1666         this value, the value is sensed by the finisher and is
1667         read-only: otherwise the value may be written by a
1668         management or control console application.  The value (-1)
1669         means other and specifically indicates that the device
1670         places no restrictions on this parameter.  The value (-2)
1671         means unknown."
1672     DEFVAL      { -2 }          -- unknown
1673     ::= { finDeviceEntry 6 }
1674
1675 finDeviceAssociatedMediaPaths OBJECT-TYPE
1676     SYNTAX      OCTET STRING
1677     MAX-ACCESS  read-only
1678     STATUS      current
1679     DESCRIPTION
1680         "Indicates the media paths which can supply media for this
1681
1682
1683 Bergman, Lewis

```

```

1684
1685 INTERNET-DRAFT          Printer Finishing MIB          May 9, 2000
1686
1687
1688     finisher device.  The value of this object is a bit map in an
1689     octet string with each position representing the value of a
1690     prtMediaPathIndex.  For a media path that can be a source
1691     for this finisher device subunit, the bit position equal
1692     to one less than the value of prtMediaPathIndex will be set.
1693     The bits are numbered starting with the most significant bit of
1694     the first byte being bit 0, the least significant bit of the
1695     first byte being bit 7, the most significant of the second byte
1696     being bit 8, and so on."
1697 ::= { finDeviceEntry 7 }
1698
1699 finDeviceAssociatedOutputs OBJECT-TYPE
1700     SYNTAX          OCTET STRING
1701     MAX-ACCESS      read-only
1702     STATUS          current
1703     DESCRIPTION
1704         "Indicates the printer output subunits this finisher device
1705         subunit services.  The value of this object is a bit map in an
1706         octet string with each position representing the value of a
1707         prtOutputIndex.  For an output subunit that is serviced
1708         by this finisher device subunit, the bit position equal to
1709         to one less than the value of prtOutputIndex will be set.
1710         The bits are numbered starting with the most significant bit of
1711         the first byte being bit 0, the least significant bit of the
1712         first byte being bit 7, the most significant of the second byte
1713         being bit 8, and so on."
1714     ::= { finDeviceEntry 8 }
1715
1716 finDeviceStatus OBJECT-TYPE
1717     SYNTAX          PrtSubUnitStatusTC
1718     MAX-ACCESS      read-only
1719     STATUS          current
1720     DESCRIPTION
1721         "Indicates the current status of this finisher device
1722         subunit."
1723     DEFVAL          { 5 }          -- unknown
1724     ::= { finDeviceEntry 9 }
1725
1726 finDeviceDescription OBJECT-TYPE
1727     SYNTAX          OCTET STRING (SIZE(0..255))
1728     MAX-ACCESS      read-only
1729     STATUS          current
1730     DESCRIPTION
1731         "A free form text description of this device subunit in the
1732         localization specified by prtGeneralCurrentLocalization."
1733     ::= { finDeviceEntry 10 }
1734
1735
1736 -- Finisher Supply Group (Mandatory)
1737 --
1738 -- A finisher device, but not all finisher devices, may have one or more
1739
1740
1741 Bergman, Lewis

```

```

1742
1743 INTERNET-DRAFT          Printer Finishing MIB          May 9, 2000
1744
1745
1746 -- supplies associated with it.  For example a finisher may use both
1747 -- binding tape and stitching wire supplies.  A finisher may also have
1748 -- more than one source for a given type of supply e.g. multiple supply
1749 -- sources of ink for imprinters.
1750
1751 finSupply OBJECT IDENTIFIER ::= { printmib 31 }
1752
1753 finSupplyTable OBJECT-TYPE
1754     SYNTAX      SEQUENCE OF FinSupplyEntry
1755     MAX-ACCESS  not-accessible
1756     STATUS      current
1757     DESCRIPTION
1758         "Each unique source of supply is an entry in the finisher
1759         supply table.  Each supply entry has its own
1760         characteristics associated with it such as colorant and
1761         current supply level."
1762     ::= { finSupply 1 }
1763
1764 finSupplyEntry OBJECT-TYPE
1765     SYNTAX      FinSupplyEntry
1766     MAX-ACCESS  not-accessible
1767     STATUS      current
1768     DESCRIPTION
1769         "A list of finisher devices, with their associated
1770         supplies and supplies characteristics."
1771     INDEX      { hrDeviceIndex, finSupplyIndex }
1772     ::= { finSupplyTable 1 }
1773
1774 FinSupplyEntry ::= SEQUENCE {
1775     finSupplyIndex          Integer32,
1776     finSupplyDeviceIndex   Integer32,
1777     finSupplyClass         PrtMarkerSuppliesClassTC,
1778     finSupplyType          PrtMarkerSuppliesTypeTC,
1779     finSupplyDescription   OCTET STRING,
1780     finSupplyUnit          PrtMarkerSuppliesSupplyUnitTC,
1781     finSupplyMaxCapacity   Integer32,
1782     finSupplyCurrentLevel  Integer32,
1783     finSupplyColorName    OCTET STRING
1784 }
1785
1786 finSupplyIndex OBJECT-TYPE
1787     SYNTAX      Integer32 (1..65535)
1788     MAX-ACCESS  not-accessible
1789     STATUS      current
1790     DESCRIPTION
1791         "A unique value used by a finisher to identify this supply
1792         container/receptacle.  Although these values may change
1793         due to a major reconfiguration of the finisher (e.g. the
1794         addition of new supply sources to the finisher), values
1795         are normally expected to remain stable across successive
1796         power cycles."
1797
1798
1799 Bergman, Lewis

```

```

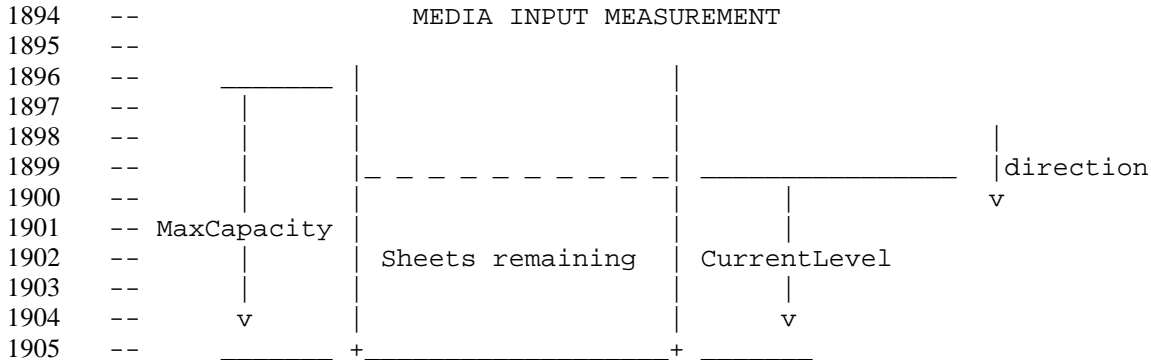
1800
1801 INTERNET-DRAFT          Printer Finishing MIB          May 9, 2000
1802
1803
1804 ::= { finSupplyEntry 1 }
1805
1806 finSupplyDeviceIndex OBJECT-TYPE
1807     SYNTAX      Integer32 (0..65535)
1808     MAX-ACCESS  read-only
1809     STATUS      current
1810     DESCRIPTION
1811         "The value of finDeviceIndex corresponding to the finishing
1812         device subunit with which this finisher supply is associated."
1813     ::= { finSupplyEntry 2 }
1814
1815 finSupplyClass OBJECT-TYPE
1816     SYNTAX      PrtMarkerSuppliesClassTC
1817     MAX-ACCESS  read-only
1818     STATUS      current
1819     DESCRIPTION
1820         "This value indicates whether this supply entity
1821         represents a supply that is consumed or a container that
1822         is filled."
1823     ::= { finSupplyEntry 3 }
1824
1825 finSupplyType OBJECT-TYPE
1826     SYNTAX      PrtMarkerSuppliesTypeTC
1827     MAX-ACCESS  read-only
1828     STATUS      current
1829     DESCRIPTION
1830         "The type of this supply."
1831     ::= { finSupplyEntry 4 }
1832
1833 finSupplyDescription OBJECT-TYPE
1834     SYNTAX      OCTET STRING (SIZE(0..255))
1835     MAX-ACCESS  read-only
1836     STATUS      current
1837     DESCRIPTION
1838         "The description of this supply/receptacle in text useful
1839         for operators and management applications and in the
1840         localization specified by prtGeneralCurrentLocalization."
1841     ::= { finSupplyEntry 5 }
1842
1843 finSupplyUnit OBJECT-TYPE
1844     SYNTAX      PrtMarkerSuppliesSupplyUnitTC
1845     MAX-ACCESS  read-only
1846     STATUS      current
1847     DESCRIPTION
1848         "Unit of measure of this finisher supply container or
1849         receptacle."
1850     ::= { finSupplyEntry 6 }
1851
1852 finSupplyMaxCapacity OBJECT-TYPE
1853     SYNTAX      Integer32 (-2..2147483647)
1854     MAX-ACCESS  read-write
1855
1856
1857 Bergman, Lewis

```


1861
 1862 STATUS current
 1863 DESCRIPTION
 1864 "The maximum capacity of this supply container/receptacle
 1865 expressed in Supply Units. If this supply container/
 1866 receptacle can reliably sense this value, the value is
 1867 sensed and is read-only; otherwise the value may be
 1868 written by a control panel or management application. The
 1869 value (-1) means other and places no restrictions on this
 1870 parameter. The value (-2) means unknown."
 1871 DEFVAL { -2 } -- unknown
 1872 ::= { finSupplyEntry 7 }

1873
 1874 finSupplyCurrentLevel OBJECT-TYPE
 1875 SYNTAX Integer32
 1876 MAX-ACCESS read-write
 1877 STATUS current
 1878 DESCRIPTION
 1879 "The current level if this supply is a container; the
 1880 remaining space if this supply is a receptacle. If this
 1881 supply container/receptacle can reliably sense this value,
 1882 the value is sensed and is read-only; otherwise the value
 1883 may be written by a control panel or management
 1884 application. The value (-1) means other and places no
 1885 restrictions on this parameter. The value (-2) means
 1886 unknown. A value of (-3) means that the printer knows there
 1887 is some supply or remaining space."
 1888 DEFVAL { -2 } -- unknown
 1889 ::= { finSupplyEntry 8 }

1891
 1892 -- Capacity Attribute Relationships



1906
 1907
 1908 finSupplyColorName OBJECT-TYPE
 1909 SYNTAX OCTET STRING (SIZE(0..63))
 1910 MAX-ACCESS read-only
 1911 STATUS current
 1912 DESCRIPTION

```

1916
1917 INTERNET-DRAFT          Printer Finishing MIB          May 9, 2000
1918
1919
1920     "The name of the color of this colorant using standardized
1921     string names from ISO 10175 (DPA) and ISO 10180 (SPDL)
1922     which are: other, unknown, white, red, green, blue, cyan,
1923     magenta, yellow and black. Implementors may add
1924     additional string values. The naming conventions in ISO
1925     9070 are recommended in order to avoid potential name
1926     clashes."
1927 ::= { finSupplyEntry 9 }
1928
1929
1930 -- Finisher Supply, Media Input Group (Conditionally Mandatory)
1931 --
1932 -- A finisher device may have one or more associated supply media
1933 -- inputs. Each entry in this table defines an input for a
1934 -- supply media type such as inserts, covers, etc.
1935 --
1936 -- This group is mandatory only if the printer system contains a
1937 -- finisher device that requires a media supply used exclusively by a
1938 -- finishing process. Examples are inserts or covers that are not
1939 -- supplied by an input subunit that provides media to the marker.
1940
1941 finSupplyMediaInput OBJECT IDENTIFIER ::= { printmib 32 }
1942
1943 finSupplyMediaInputTable OBJECT-TYPE
1944     SYNTAX      SEQUENCE OF FinSupplyMediaInputEntry
1945     MAX-ACCESS  not-accessible
1946     STATUS      current
1947     DESCRIPTION
1948         "The input subunits associated with a finisher supply media
1949         are each represented by an entry in this table."
1950 ::= { finSupplyMediaInput 1 }
1951
1952 finSupplyMediaInputEntry OBJECT-TYPE
1953     SYNTAX      FinSupplyMediaInputEntry
1954     MAX-ACCESS  not-accessible
1955     STATUS      current
1956     DESCRIPTION
1957         "A list of finisher supply media input subunit features and
1958         characteristics."
1959     INDEX      { hrDeviceIndex, finSupplyMediaInputIndex }
1960 ::= { finSupplyMediaInputTable 1 }
1961
1962 FinSupplyMediaInputEntry ::= SEQUENCE {
1963     finSupplyMediaInputIndex      Integer32,
1964     finSupplyMediaInputDeviceIndex Integer32,
1965     finSupplyMediaInputSupplyIndex Integer32,
1966     finSupplyMediaInputType       PrtInputTypeTC,
1967     finSupplyMediaInputDimUnit     PrtMediaUnitTC,
1968     finSupplyMediaInputMediaDimFeedDir Integer32,
1969     finSupplyMediaInputMediaDimXFeedDir Integer32,
1970     finSupplyMediaInputStatus     PrtSubUnitStatusTC,
1971
1972
1973 Bergman, Lewis

```

```

1974
1975 INTERNET-DRAFT          Printer Finishing MIB          May 9, 2000
1976
1977
1978     finSupplyMediaInputMediaName      OCTET STRING,
1979     finSupplyMediaInputName           OCTET STRING,
1980     finSupplyMediaInputDescription    OCTET STRING,
1981     finSupplyMediaInputSecurity       PresentOnOff,
1982     finSupplyMediaInputMediaWeight    Integer32,
1983     finSupplyMediaInputMediaThickness Integer32,
1984     finSupplyMediaInputMediaType      OCTET STRING
1985 }
1986
1987 finSupplyMediaInputIndex OBJECT-TYPE
1988     SYNTAX      Integer32 (1..65535)
1989     MAX-ACCESS  not-accessible
1990     STATUS      current
1991     DESCRIPTION
1992         "A unique value used by a finisher to identify this supply
1993         media input subunit.  Although these values may change
1994         due to a major reconfiguration of the finisher (e.g. the
1995         addition of new supply media input sources to the
1996         finisher), values are normally expected to remain stable
1997         across successive power cycles."
1998     ::= { finSupplyMediaInputEntry 1 }
1999
2000 finSupplyMediaInputDeviceIndex OBJECT-TYPE
2001     SYNTAX      Integer32 (0..65535)
2002     MAX-ACCESS  read-only
2003     STATUS      current
2004     DESCRIPTION
2005         "The value of finDeviceIndex corresponding to the finishing
2006         device subunit with which this finisher media supply is
2007         associated."
2008     ::= { finSupplyMediaInputEntry 2 }
2009
2010 finSupplyMediaInputSupplyIndex OBJECT-TYPE
2011     SYNTAX      Integer32 (0..65535)
2012     MAX-ACCESS  read-only
2013     STATUS      current
2014     DESCRIPTION
2015         "The value of finSupplyIndex corresponding to the finishing
2016         supply subunit with which this finisher media supply is
2017         associated."
2018     ::= { finSupplyMediaInputEntry 3 }
2019
2020 finSupplyMediaInputType OBJECT-TYPE
2021     SYNTAX      PrtInputTypeTC
2022     MAX-ACCESS  read-only
2023     STATUS      current
2024     DESCRIPTION
2025         "The type of technology (discriminated primarily according
2026         to the feeder mechanism type) employed by the input
2027         subunit."
2028     ::= { finSupplyMediaInputEntry 4 }
2029
2030
2031 Bergman, Lewis

```

```

2032
2033 INTERNET-DRAFT          Printer Finishing MIB          May 9, 2000
2034
2035
2036
2037 finSupplyMediaInputDimUnit OBJECT-TYPE
2038     SYNTAX          PrtMediaUnitTC
2039     MAX-ACCESS      read-only
2040     STATUS          current
2041     DESCRIPTION
2042         "The unit of measure for specifying dimensional values for
2043         this input device."
2044     ::= { finSupplyMediaInputEntry 5 }
2045
2046 finSupplyMediaInputMediaDimFeedDir OBJECT-TYPE
2047     SYNTAX          Integer32 (-2..2147483647)
2048     MAX-ACCESS      read-write
2049     STATUS          current
2050     DESCRIPTION
2051         "This object provides the value of the dimension in the
2052         feed direction of the media that is placed or will be
2053         placed in this input device.  Feed dimension measurements
2054         are taken parallel to the feed direction of the device and
2055         measured in finSupplyMediaInputDimUnits.  If this input
2056         device can reliably sense this value, the value is sensed
2057         and is read-only access.  Otherwise the value is read-write
2058         access and may be written by management or control panel
2059         applications.  The value (-1) means other and specifically
2060         indicates that this device places no restrictions on this
2061         parameter.  The value (-2) indicates unknown. "
2062     ::= { finSupplyMediaInputEntry 6 }
2063
2064 finSupplyMediaInputMediaDimXFeedDir OBJECT-TYPE
2065     SYNTAX          Integer32 (-2..2147483647)
2066     MAX-ACCESS      read-write
2067     STATUS          current
2068     DESCRIPTION
2069         "This object provides the value of the dimension across the
2070         feed direction of the media that is placed or will be
2071         placed in this input device.  The cross feed direction is
2072         ninety degrees relative to the feed direction on this
2073         device and measured in finSupplyMediaInputDimUnits.  If
2074         this input device can reliably sense this value, the value
2075         is sensed and is read-only access.  Otherwise the value is
2076         read-write access and may be written by management or
2077         control panel applications.  The value (-1) means other and
2078         specifically indicates that this device places no
2079         restrictions on this parameter.  The value (-2) indicates
2080         unknown. "
2081     ::= { finSupplyMediaInputEntry 7 }
2082
2083 finSupplyMediaInputStatus OBJECT-TYPE
2084     SYNTAX          PrtSubUnitStatusTC
2085     MAX-ACCESS      read-only
2086     STATUS          current
2087
2088
2089 Bergman, Lewis

```

```

2090
2091 INTERNET-DRAFT          Printer Finishing MIB          May 9, 2000
2092
2093
2094     DESCRIPTION
2095     "This value indicates the current status of this input
2096     device."
2097     DEFVAL      { 5 }      -- unknown
2098     ::= { finSupplyMediaInputEntry 8 }
2099
2100 finSupplyMediaInputMediaName OBJECT-TYPE
2101     SYNTAX      OCTET STRING (SIZE(0..63))
2102     MAX-ACCESS  read-write
2103     STATUS      current
2104     DESCRIPTION
2105     "The name of the current media contained in this input
2106     device. Examples are Engineering Manual Cover, Section A Tab
2107     Divider or any ISO standard names."
2108     ::= { finSupplyMediaInputEntry 9 }
2109
2110 finSupplyMediaInputName OBJECT-TYPE
2111     SYNTAX      OCTET STRING (SIZE(0..63))
2112     MAX-ACCESS  read-write
2113     STATUS      current
2114     DESCRIPTION
2115     "The name assigned to this input subunit."
2116     ::= { finSupplyMediaInputEntry 10 }
2117
2118 finSupplyMediaInputDescription OBJECT-TYPE
2119     SYNTAX      OCTET STRING (SIZE(0..255))
2120     MAX-ACCESS  read-only
2121     STATUS      current
2122     DESCRIPTION
2123     "A free form text description of this input subunit in the
2124     localization specified by prtGeneralCurrentLocalization."
2125     ::= { finSupplyMediaInputEntry 11 }
2126
2127 finSupplyMediaInputSecurity OBJECT-TYPE
2128     SYNTAX      PresentOnOff
2129     MAX-ACCESS  read-write
2130     STATUS      current
2131     DESCRIPTION
2132     "Indicates if this subunit has some security associated
2133     with it."
2134     ::= { finSupplyMediaInputEntry 12 }
2135
2136 finSupplyMediaInputMediaWeight OBJECT-TYPE
2137     SYNTAX      Integer32
2138     MAX-ACCESS  read-write
2139     STATUS      current
2140     DESCRIPTION
2141     "The weight of the media associated with this Input device
2142     in grams per meter squared. The value (-1) means other
2143     and specifically indicates that the device places no
2144     restriction on this parameter. The value (-2) means
2145
2146
2147 Bergman, Lewis

```

```

2148
2149 INTERNET-DRAFT          Printer Finishing MIB          May 9, 2000
2150
2151
2152     unknown.  This object can be used to calculate the weight
2153     of individual pages processed by the document finisher.
2154     This value, when multiplied by the number of pages in a
2155     finished set, can be used to calculate the weight of a set
2156     before it is inserted into a mailing envelope."
2157 ::= { finSupplyMediaInputEntry 13 }
2158
2159 finSupplyMediaInputMediaThickness OBJECT-TYPE
2160     SYNTAX      Integer32 (-2..2147483647)
2161     MAX-ACCESS  read-write
2162     STATUS      current
2163     DESCRIPTION
2164         "This object identifies the thickness of the input media
2165         processed by this document input subunit measured in
2166         micrometers.  This value may be used by devices (or
2167         operators) to set up proper machine tolerances for the
2168         feeder operation.  The value (-2) indicates that the media
2169         thickness is unknown or not used in the setup for this
2170         input subunit."
2171 ::= { finSupplyMediaInputEntry 14 }
2172
2173 finSupplyMediaInputMediaType OBJECT-TYPE
2174     SYNTAX      OCTET STRING (SIZE(0..63))
2175     MAX-ACCESS  read-write
2176     STATUS      current
2177     DESCRIPTION
2178         "The name of the type of medium associated with this input
2179         subunit.  Valid values are standardized strings from ISO 10175
2180         (DPA) and ISO 10180 (SPDL) which are: stationary, transparency,
2181         envelope, envelope-plain, envelope window, continuous-long,
2182         continuous-short, tab-stock, labels, multi-layer."
2183 ::= { finSupplyMediaInputEntry 15 }
2184
2185
2186 -- Finisher Device Attribute Group (Mandatory)
2187 --
2188 -- A finisher device subunit may have one or more parameters that
2189 -- cannot be specified by any other objects in the MIB.  The
2190 -- Device Attribute group facilitates the definition of these
2191 -- parameters.  The objects which define the attributes are
2192 -- read-write, to allow both Set and Get operations.
2193 --
2194 -- At least one table entry must exist for each finisher device defined
2195 -- by the MIB.  If no other entry is possible for a finisher device, the
2196 -- deviceName(3) attribute MUST be returned.
2197
2198 finDeviceAttribute OBJECT IDENTIFIER ::= { printmib 33 }
2199
2200 finDeviceAttributeTable OBJECT-TYPE
2201     SYNTAX      SEQUENCE OF FinDeviceAttributeEntry
2202     MAX-ACCESS  not-accessible
2203
2204
2205 Bergman, Lewis

```

```

2206
2207 INTERNET-DRAFT          Printer Finishing MIB          May 9, 2000
2208
2209
2210     STATUS      current
2211     DESCRIPTION
2212         "The attribute table defines special parameters that are
2213         applicable only to a minority of the finisher devices.
2214         An attribute table entry is used, rather than unique
2215         objects, to minimize the number of MIB objects and to
2216         allow for expansion without the addition of MIB objects.
2217         Each finisher device is represented by a separate row
2218         in the device subunit attribute table."
2219 ::= { finDeviceAttribute 1 }
2220
2221 finDeviceAttributeEntry OBJECT-TYPE
2222     SYNTAX      FinDeviceAttributeEntry
2223     MAX-ACCESS  not-accessible
2224     STATUS      current
2225     DESCRIPTION
2226         "Each entry defines a finisher function parameter that
2227         cannot be represented by an object in the finisher
2228         device subunit table."
2229     INDEX      { hrDeviceIndex, finDeviceIndex,
2230                finDeviceAttributeTypeIndex,
2231                finDeviceAttributeInstanceIndex }
2232 ::= { finDeviceAttributeTable 1 }
2233
2234 FinDeviceAttributeEntry ::= SEQUENCE {
2235     finDeviceAttributeTypeIndex      FinAttributeTypeTC,
2236     finDeviceAttributeInstanceIndex  Integer32,
2237     finDeviceAttributeValueAsInteger Integer32,
2238     finDeviceAttributeValueAsOctets  OCTET STRING
2239 }
2240
2241 finDeviceAttributeTypeIndex OBJECT-TYPE
2242     SYNTAX      FinAttributeTypeTC
2243     MAX-ACCESS  not-accessible
2244     STATUS      current
2245     DESCRIPTION
2246         "Defines the attribute type represented by this row."
2247 ::= { finDeviceAttributeEntry 1 }
2248
2249 finDeviceAttributeInstanceIndex OBJECT-TYPE
2250     SYNTAX      Integer32 (1..65535)
2251     MAX-ACCESS  not-accessible
2252     STATUS      current
2253     DESCRIPTION
2254         "An index that allows the discrimination of an attribute
2255         instance when the same attribute occurs multiple times for
2256         a specific instance of a finisher function.  The value of
2257         this index shall be 1 if only a single instance of the
2258         attribute occurs for the specific finisher function."
2259 ::= { finDeviceAttributeEntry 2 }
2260
2261
2262
2263 Bergman, Lewis

```

```

2264
2265 INTERNET-DRAFT          Printer Finishing MIB          May 9, 2000
2266
2267
2268 finDeviceAttributeValueAsInteger OBJECT-TYPE
2269     SYNTAX      Integer32 (-2..2147483647)
2270     MAX-ACCESS  read-write
2271     STATUS      current
2272     DESCRIPTION
2273         "Defines the integer value of the attribute.  The value of
2274         the attribute is represented as an integer if the
2275         finAttributeTypeTC description for the attribute has the
2276         tag 'INTEGER:'.
2277
2278         Depending upon the attribute enum definition, this object
2279         may be either an integer, a counter, an index, or an enum.
2280         Attributes for which the concept of an integer value is
2281         not meaningful SHALL return a value of -1 for this
2282         attribute."
2283     DEFVAL      { -2 }          -- unknown
2284     ::= { finDeviceAttributeEntry 3 }
2285
2286 finDeviceAttributeValueAsOctets OBJECT-TYPE
2287     SYNTAX      OCTET STRING (SIZE(0..63))
2288     MAX-ACCESS  read-write
2289     STATUS      current
2290     DESCRIPTION
2291         "Contains the octet string value of the attribute.  The
2292         value of the attribute is represented as a string if the
2293         finAttributeTypeTC description for the attribute has the
2294         tag 'OCTETS:'.
2295
2296         Depending upon the attribute enum definition, this object
2297         may be either a coded character set string (text) or a
2298         binary octet string.  Attributes for which the concept of
2299         an octet string value is not meaningful SHALL contain a
2300         zero length string."
2301     DEFVAL      { 'H' }        -- empty string
2302     ::= { finDeviceAttributeEntry 4 }
2303
2304
2305 -- Conformance Information
2306
2307 -- compliance statements
2308
2309 finMIBCompliance MODULE-COMPLIANCE
2310     STATUS      current
2311     DESCRIPTION
2312         "The compliance statement for agents that implement the
2313         finisher MIB."
2314     MODULE -- this module
2315     MANDATORY-GROUPS { finDeviceGroup, finSupplyGroup,
2316                       finDeviceAttributeGroup }
2317
2318     OBJECT      finDevicePresentOnOff
2319
2320
2321 Bergman, Lewis

```


2322
2323 INTERNET-DRAFT Printer Finishing MIB May 9, 2000
2324
2325
2326 MIN-ACCESS read-only
2327 DESCRIPTION
2328 "It is conformant to implement this object as read-only."
2329
2330 OBJECT finDeviceMaxCapacity
2331 MIN-ACCESS read-only
2332 DESCRIPTION
2333 "It is conformant to implement this object as read-only."
2334
2335 OBJECT finDeviceCurrentCapacity
2336 MIN-ACCESS read-only
2337 DESCRIPTION
2338 "It is conformant to implement this object as read-only."
2339
2340 OBJECT finSupplyMaxCapacity
2341 MIN-ACCESS read-only
2342 DESCRIPTION
2343 "It is conformant to implement this object as read-only."
2344
2345 OBJECT finSupplyCurrentLevel
2346 MIN-ACCESS read-only
2347 DESCRIPTION
2348 "It is conformant to implement this object as read-only."
2349
2350 OBJECT finSupplyMediaInputMediaDimFeedDir
2351 MIN-ACCESS read-only
2352 DESCRIPTION
2353 "It is conformant to implement this object as read-only."
2354
2355 OBJECT finSupplyMediaInputMediaDimXFeedDir
2356 MIN-ACCESS read-only
2357 DESCRIPTION
2358 "It is conformant to implement this object as read-only."
2359
2360 OBJECT finSupplyMediaInputMediaName
2361 MIN-ACCESS read-only
2362 DESCRIPTION
2363 "It is conformant to implement this object as read-only."
2364
2365 OBJECT finSupplyMediaInputName
2366 MIN-ACCESS read-only
2367 DESCRIPTION
2368 "It is conformant to implement this object as read-only."
2369
2370 OBJECT finSupplyMediaInputSecurity
2371 MIN-ACCESS read-only
2372 DESCRIPTION
2373 "It is conformant to implement this object as read-only."
2374
2375 OBJECT finSupplyMediaInputMediaWeight
2376 MIN-ACCESS read-only
2377
2378
2379 Bergman, Lewis [page 41]

```

2380
2381 INTERNET-DRAFT          Printer Finishing MIB          May 9, 2000
2382
2383
2384     DESCRIPTION
2385         "It is conformant to implement this object as read-only."
2386
2387     OBJECT      finSupplyMediaInputMediaThickness
2388     MIN-ACCESS  read-only
2389     DESCRIPTION
2390         "It is conformant to implement this object as read-only."
2391
2392     OBJECT      finSupplyMediaInputMediaType
2393     MIN-ACCESS  read-only
2394     DESCRIPTION
2395         "It is conformant to implement this object as read-only."
2396
2397     OBJECT      finDeviceAttributeValueAsInteger
2398     MIN-ACCESS  read-only
2399     DESCRIPTION
2400         "It is conformant to implement this object as read-only."
2401
2402     OBJECT      finDeviceAttributeValueAsOctets
2403     MIN-ACCESS  read-only
2404     DESCRIPTION
2405         "It is conformant to implement this object as read-only."
2406
2407     GROUP      finSupplyMediaInputGroup
2408     DESCRIPTION
2409         "This group is conditionally mandatory and must be included
2410         if a finisher device requires a media supply that is used
2411         exclusively by a finishing process."
2412
2413     ::= { prtMIBConformance 3 }
2414
2415
2416 finMIBGroups      OBJECT IDENTIFIER ::= { prtMIBConformance 4 }
2417
2418 finDeviceGroup OBJECT-GROUP
2419     OBJECTS { finDeviceType, finDevicePresentOnOff,
2420             finDeviceCapacityUnit, finDeviceMaxCapacity,
2421             finDeviceCurrentCapacity, finDeviceAssociatedMediaPaths,
2422             finDeviceAssociatedOutputs, finDeviceStatus,
2423             finDeviceDescription }
2424     STATUS current
2425     DESCRIPTION
2426         "The finisher device group."
2427     ::= { finMIBGroups 1 }
2428
2429 finSupplyGroup OBJECT-GROUP
2430     OBJECTS { finSupplyDeviceIndex, finSupplyClass, finSupplyType,
2431             finSupplyDescription, finSupplyUnit, finSupplyMaxCapacity,
2432             finSupplyCurrentLevel, finSupplyColorName }
2433     STATUS current
2434     DESCRIPTION
2435
2436
2437 Bergman, Lewis

```

```

2438
2439 INTERNET-DRAFT          Printer Finishing MIB          May 9, 2000
2440
2441
2442     "The finisher supply group."
2443     ::= { finMIBGroups 2 }
2444
2445 finSupplyMediaInputGroup OBJECT-GROUP
2446     OBJECTS { finSupplyMediaInputDeviceIndex,
2447               finSupplyMediaInputSupplyIndex, finSupplyMediaInputType,
2448               finSupplyMediaInputDimUnit,
2449               finSupplyMediaInputMediaDimFeedDir,
2450               finSupplyMediaInputMediaDimXFeedDir,
2451               finSupplyMediaInputStatus, finSupplyMediaInputMediaName,
2452               finSupplyMediaInputName, finSupplyMediaInputDescription,
2453               finSupplyMediaInputSecurity,
2454               finSupplyMediaInputMediaWeight,
2455               finSupplyMediaInputMediaThickness,
2456               finSupplyMediaInputMediaType }
2457     STATUS current
2458     DESCRIPTION
2459         "The finisher supply, media input group."
2460     ::= { finMIBGroups 3 }
2461
2462 finDeviceAttributeGroup OBJECT-GROUP
2463     OBJECTS { finDeviceAttributeValueAsInteger,
2464               finDeviceAttributeValueAsOctets }
2465     STATUS current
2466     DESCRIPTION
2467         "The finisher device attribute group."
2468     ::= { finMIBGroups 4 }
2469
2470 END
2471
2472
2473
2474
2475 8. REFERENCES
2476
2477 [PRTMIB] The Printer MIB, RFC XXXX, IETF standards track document.
2478
2479 [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
2480     Requirement Levels", BCP 14, RFC 2119, March 1997.
2481
2482 [LMO] Large Mailing Operations Specification, DMTF. See
2483     http://www.dmtf.org/tech/apps.html
2484
2485 [DPA] ISO/IEC 10175 Document Printing Application (DPA). See
2486     ftp://ftp.pwg.org/pub/pwg/dpa/
2487
2488 [IPP] Internet Printing Protocol/1.0: Model and Semantics, RFC 2566.
2489
2490
2491
2492
2493
2494
2495 Bergman, Lewis

```

2500 9. SECURITY CONSIDERATIONS

2501
2502 The Finisher MIB specifies a database and not necessarily a protocol
2503 for accessing the database. With regards to the security of the
2504 information within the database, it is anticipated that the primary
2505 vehicle for accessing this data will be through the use of the Simple
2506 Network Management Protocol (SNMP). SNMP Version 1 has known security
2507 vulnerabilities due to the fact that only a "clear text" community
2508 name is used for accessing MIB objects within a particular entity.
2509 SNMPv2 attempted to provide additional security features but is still
2510 not considered to be an adequate solution. SNMP Version 3 provides
2511 much more robust security mechanisms with regards to with support
2512 authentication, as well as privacy.
2513

2514
2515
2516 10. FULL COPYRIGHT STATEMENT

2517
2518 Copyright (C) The Internet Society (1998). All Rights Reserved.
2519

2520 This document and translations of it may be copied and furnished to
2521 others, and derivative works that comment on or otherwise explain it
2522 or assist in its implementation may be prepared, copied, published
2523 and distributed, in whole or in part, without restriction of any
2524 kind, provided that the above copyright notice and this paragraph are
2525 included on all such copies and derivative works. However, this
2526 document itself may not be modified in any way, such as by removing
2527 the copyright notice or references to the Internet Society or other
2528 Internet organizations, except as needed for the purpose of
2529 developing Internet standards in which case the procedures for
2530 copyrights defined in the Internet Standards process must be
2531 followed, or as required to translate it into languages other than
2532 English.
2533

2534 The limited permissions granted above are perpetual and will not be
2535 revoked by the Internet Society or its successors or assigns.
2536

2537 This document and the information contained herein is provided on an
2538 "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING
2539 TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING
2540 BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION
2541 HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF
2542 MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.
2543

2558 11. AUTHORS

2559
2560 This document was created with significant contributions from the
2561 following individuals.

2562
2563 Ron Bergman (Editor)
2564 Hitachi Koki Imaging Solutions
2565 1757 Tapo Canyon Road
2566 Simi Valley, CA 93063-3394

2567
2568 Phone: 805-578-4421
2569 Fax: 805-578-4001
2570 Email: rbergma@hitachi-hkis.com

2571
2572
2573 Harry Lewis (chairman)
2574 IBM Corporation
2575 6300 Diagonal Hwy
2576 Boulder, CO 80301

2577
2578 Phone: (303) 924-5337
2579 Fax: (303) 924-4662
2580 Email: harryl@us.ibm.com

2581
2582
2583
2584 Send comments to the Printer Working Group (PWG) using the Finisher
2585 MIB Project (FIN) Mailing List: fin@pwg.org

2586
2587 Implementers of this specification are encouraged to join this email
2588 distribution list in order to participate in any discussions of
2589 clarification issues and review registration proposals for
2590 additional attributes and enum values.

2591
2592 For further information, access the PWG web page under "FIN":
2593 <http://www.pwg.org/>

2594
2595
2596 Other Participants:

2597
2598 Chuck Adams - Tektronix
2599 Carlos Becerra - HP
2600 Andy Davidson - Tektronix
2601 Mabry Dozier - QMS
2602 Lee Farrell - Canon
2603 Jennifer Gattis - Duplo USA
2604 Paul Gloger - Xerox
2605 Richard Hart - Digital
2606 Tom Hastings - Xerox
2607 Scott Isaacson - Novell
2608 David Kellerman - Northlake Software

2612
2613 INTERNET-DRAFT Printer Finishing MIB May 9, 2000
2614
2615
2616 Henrik Holst - i-data International
2617 Rick Landau - Digital
2618 Jay Martin - Underscore
2619 Gary Padlipski - Xerox
2620 Kevin Palmer - Duplo USA
2621 Bob Pentecost - HP
2622 Stuart Rowley - Kyocera
2623 Yuki Sacchi - Japan Computer Industry
2624 Philip Thambidunai - Okidata
2625 William Wagner - DPI/Osicom
2626 Chris Wellens - Interworking Labs
2627 Don Wright - Lexmark
2628 Lloyd Young - Lexmark
2629
2630
2631
2632
2633
2634
2635
2636
2637
2638
2639
2640
2641
2642
2643
2644
2645
2646
2647
2648
2649
2650
2651
2652
2653
2654
2655
2656
2657
2658
2659
2660
2661
2662
2663
2664
2665
2666
2667
2668
2669 Bergman, Lewis [page 46]