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
"1id-abstracts.txt" listing contained in the Internet-Drafts Shadow
Directories on ftp.is.co.za (Africa), nic.nordu.net (Europe),
munnari.oz.au (Pacific Rim), ftp.ietf.org (US East Coast), or
ftp.isi.edu (US West Coast).

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.

TABLE OF CONTENTS

1.   INTRODUCTION

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

    - Provides the status of the finishing device.
    - Queries and controls the features and configuration of the
      finishing device.
    - Enables and disables the finishing processes.
    - Allows unsolicited status from the finishing device.

1.1 Scope

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

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

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

  - print-job-level finishing processes that are applied to a
    complete print job,

  - document-level finishing processes that are applied individually
    to each document in the print job,

  - document-level finishing processes that are applied to a selected
    document in the print job.

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

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

1.2 Rational

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

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

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

2. TERMINOLOGY
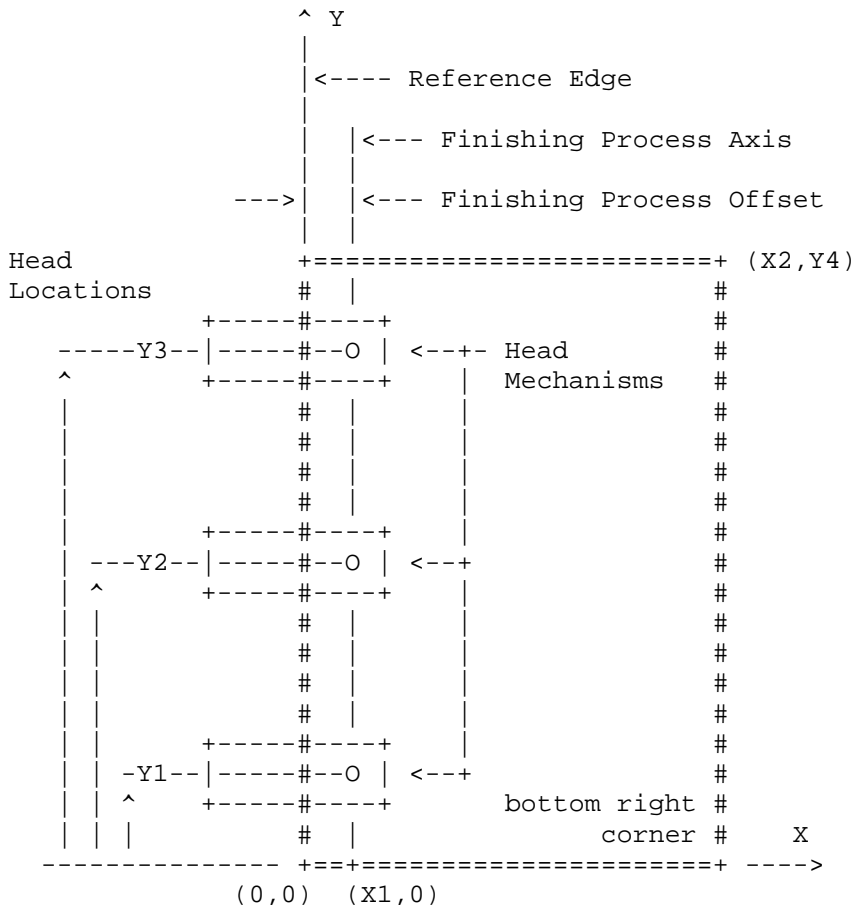
Where appropriate, the Printer Finishing MIB will conform to the
terminology, syntax, and semantics from the DMTF Large Mailing
Operations standard [LMO], the Internet Printing Protocol [IPP],
and/or the ISO Document Printing Application [DPA].

2.1 General Terminology

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

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

```
                            ^ Y
                            |
                            |<---- Reference Edge
                            |
                            |   |<--- Finishing Process Axis
                            |   |
                        --->|   |<--- Finishing Process Offset
                            |   |
       Head                 +========================+ (X2,Y4)
       Locations         #  |                        #
                  +-----#----+                       #
      -----Y3--|-----#--O |  <--+- Head             #
         ^       +-----#----+     |  Mechanisms       #
         |          #  |          |                   #
         |          #  |          |                   #
         |          #  |          |                   #
         |          #  |          |                   #
         |       +-----#----+     |                   #
         | ---Y2--|-----#--O |  <--+                  #
         | ^       +-----#----+     |                  #
         | |          #  |          |                  #
         | |          #  |          |                  #
         | |          #  |          |                  #
         | |          #  |          |                  #
         | |       +-----#----+     |                  #
         | | -Y1--|-----#--O |  <--+                  #
         | | ^       +-----#----+        bottom right #
         | | |          #  |                 corner #    X
        -------------- +==+====================+ ---->
                 (0,0)  (X1,0)
```

       Figure 1 - Finishing Process Axis Parallel to Y Axis

```
          ^ Y
          |            Head Locations
          |<---------------->|---X2
          |<---->|---X1       |
          |       |           |
          |     +-|-+       +-|-+
          |     | | |       | | |  (X3,Y2)
          +======|==========|======+
          #     | | |       | | |      # Finishing Process Axis
          #----|  O  |-------|  O  |----#----- Y1
          #     +---+       +---+   #  ^
          #      ^           ^      #  |
          #      |           |      #  |
          #      +-----------+      #  |
          #      |                  #  |
          #     Head               #  |
          #     Mechanisms         #  |
          #                        #  |
          #                        # Finishing Process Offset
          #                        #  |
          #                        #  |
          #                        #  |
          #                        #  |
          #                        #  |
          #                        #  |
          #            bottom right #  |
          #                  corner #  v   X
          +=========================+ ------>
        (0,0)                Reference Edge
```
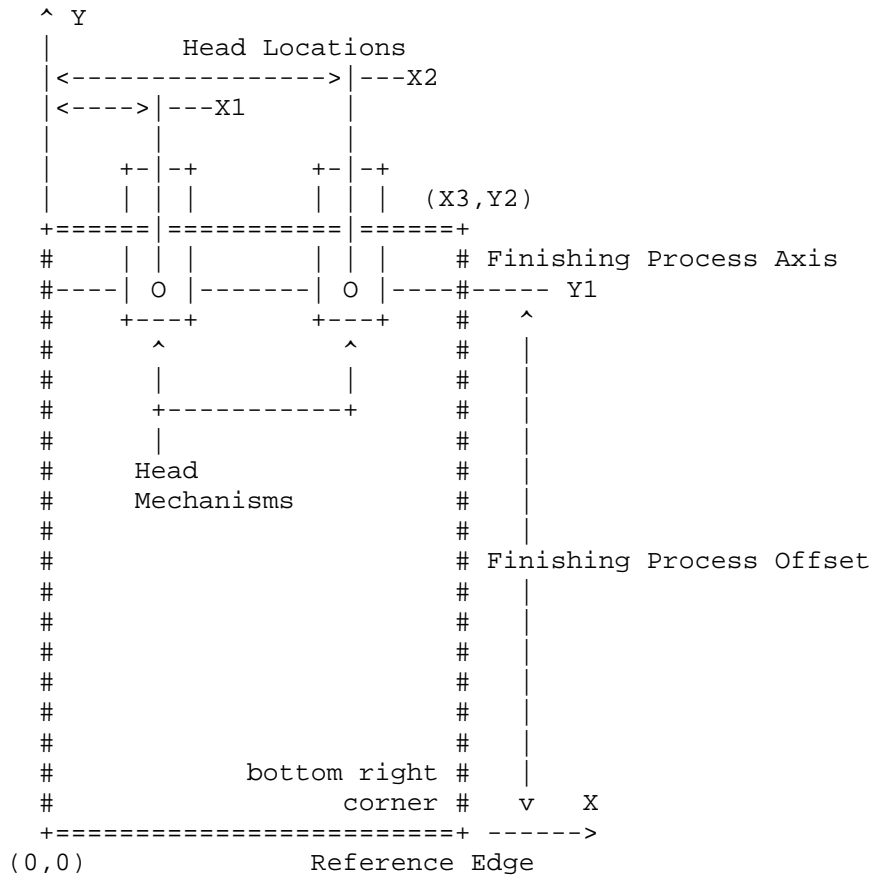
          Figure 2 - Finishing Process Axis Parallel to X Axis

   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

     document, and applying a binding to a document.

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

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

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

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

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

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

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

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

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

2.2 Process Specific Terminology

   FOLDING:

   Z Fold: A fold in which two folds are placed in the sheet in opposite

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

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

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

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

 BINDING:

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

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

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

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

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

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

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

   SLITTING/CUTTING/TRIMMING:

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

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

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

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

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

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

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

   STITCHING/STAPLING:

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

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

       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:

```
                      +----------+
                   +----------+ |
                   |  Marker  | |
                   | Supplies |-+
                   +----------+
                        \
   +-----+               \ +------+              +--------+   +------+
   |     |                \|      |              |        |   |      |
 +-----+ |    +-----+   +------+ | +------+  +--------+ | +------+ |
 |Input|-+ +------+|   |Marker|-+ +------+|  |Finisher|-+ |Output|-+
 |     |===>|     |+<==>|      |<==>|      |+==>|        |===>|      |
 +-----+    +-+ +-+    +------+   +-+ +-+  +--------+    +------+
    \          |  ||                  |   ||      \
     \         |  ||                  |   ||       \
      \        |  ||                  |   ||   +----------+
   +-------+   |  |+--------------------|   ||   | Finisher |-+
   |       |   |  +--------------------+   ||   | Supplies | |
 +-------+ |  |       Media Path       |+  +----------+ |
 | Media |-+  +------------------------+       |        |
 |(opt.) |                                     +----------+
 +-------+
```
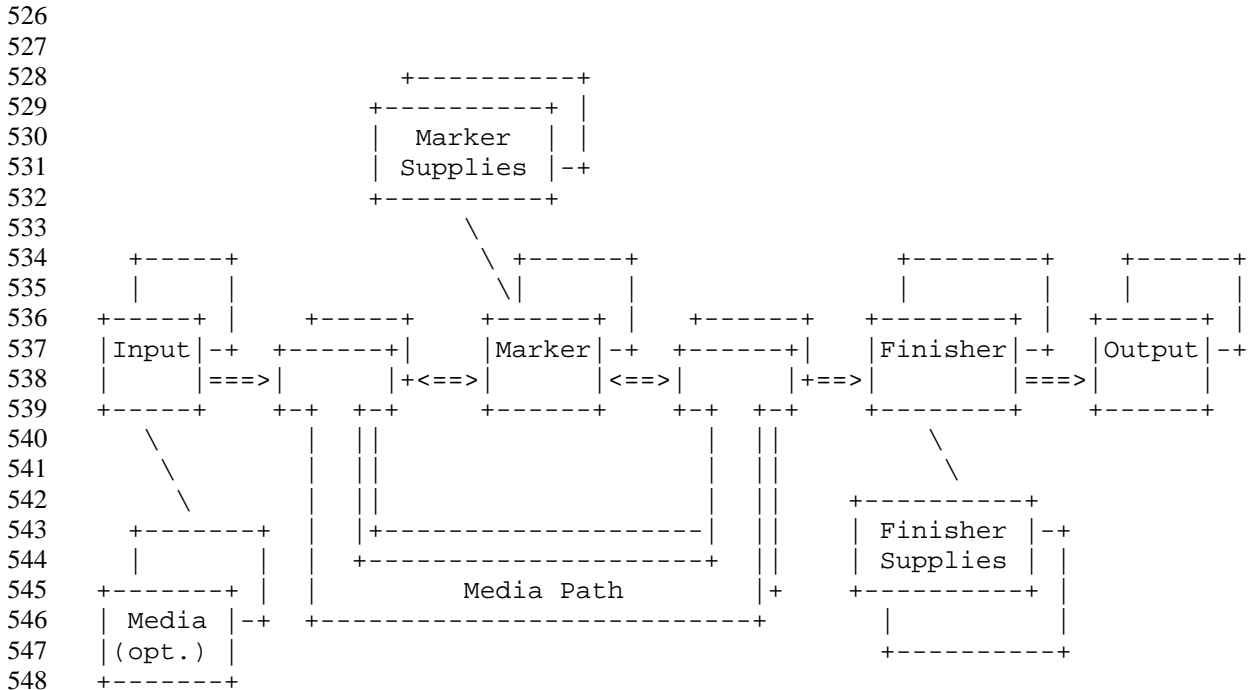
4.   FINISHING SPECIFICATIONS

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

   1. Finishing operations that can be performed on media that are
      associated with a specific printer media path and output subunit.

   2. Combinations of operations that cannot be performed.

   3. The location of the operation on the medium, if applicable.

   4. The physical characteristics of the result of the operation.
      For example, the size and shape of a punched hole, or if a fold
      operation creates a letter fold or a "Z" fold.

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

4.1 Multiple finDeviceTable Entries

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

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

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

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

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

4.2 Implicit Parameters

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

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

Notes:
   1. All hole to hole patterns are centered along the process edge.
   2. Offset is 0.18 inches to 0.51 inches.
   3. Reference edge is leftEdge(5) for letter and topEdge(3)

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

4.2.2 FinPunchHoleTypeTC, punchHoleSizeMaxDim, punchHoleSizeMinDim

   enum pattern | Hole Description
   -------------------+--
   twoHoleUSTop(4) | round(3), .2 - .32 inch diameter
   threeHoleUS(5) | round(3), .2 - .32 inch diameter
   twoHoleDIN(6) | round(3), 5 - 8 mm diameter
   fourHoleDIN(7) | round(3), 5 - 8 mm diameter
   twentyTwoHoleUS(8) | round(3), .2 - .32 inch diameter
   nineteenHoleUS(9) | rectang(6), .313 inches X .125 inches
   twoHoleMetric(10) | round(3), 5 - 8 mm diameter
   swedish4Hole(11) | round(3), 5 - 8 mm diameter
   twoHoleUSSide(12) | round(3), .2 - .32 inch diameter
   fiveHoleUS(13) | round(3), .2 - .32 inch diameter
   sevenHoleUS(14) | round(3), .2 - .32 inch diameter
   mixed7H4S(15) | round(3), 5 - 8 mm diameter
                        | rectang(6), 12 mm X 6 mm
   norweg6Hole(16) | round(3), 5 - 8 mm diameter
                        | rectang(6), 10 mm X 5.5 mm
   metric26Hole(17) | round(3), 5 - 8 mm
   metric30Hole(18) | round(3), 5 - 8 mm

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

5. THE ATTRIBUTE MECHANISM

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

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

     1. hrDeviceIndex - which device in the host

     2. finDeviceIndex - which finisher subunit in the printer device

     3. finDeviceAttributeTypeIndex - which attribute

     4. finDeviceAttributeInstanceIndex - which attribute instance for
        those attributes that can have multiple values per finishing
        subunit.

5.1 Conformance of Attribute Implementation

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

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

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

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

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

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

759
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
813   Bergman, Lewis   [page 14]

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   finDeviceAttributeAsOctets, 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
862   finAttributeTypeIndex   Data type
863   --------------------   ---------
864   other(1),   Integer32
865   AND/OR
866   OCTET STRING (SIZE(0..63))
867   INTEGER: and/or OCTETS: An attribute that is not currently
868   approved and registered.
869
870

874
875
876
877   A. Generic finisher subunit attributes that apply to all finisher
878   subunit types. (3..)
879
880   deviceName(3),   OCTET STRING (SIZE(0..63))
881   OCTETS: The name assigned to this finisher device subunit.
882
883   deviceVendorName(4),   OCTET STRING (SIZE(0..63))
884   OCTETS: The name of the vendor of this finisher device
885   subunit.
886
887   deviceModel(5),   OCTET STRING (SIZE(0..63))
888   OCTETS: The model name of this finisher device subunit.
889
890   deviceVersion(6),   OCTET STRING (SIZE(0..63))
891   OCTETS: The version string for this finisher device
892   subunit.
893
894   deviceSerialNumber(7),   OCTET STRING (SIZE(0..63))
895   OCTETS: The serial number assigned to this finisher device
896   subunit.
897
898   maximumSheets(8),   Integer32 (-2..32767)
899   INTEGER: Defines the maximum number of media sheets that a
900   finisher device is able to process.
901
902   finProcessOffsetUnits(9),   PrtMediaUnitTC
903   INTEGER: An enumeration which defines the units of measure
904   for the attributes finAxisOffset and finHeadLocation.
905
906   finReferenceEdge(10),   FinEdgeTC
907   INTEGER: An enumeration which defines which edge of the
908   form is the reference for this finishing process. The
909   Finishing Process Axis will be parallel to this axis.
910
911   finAxisOffset(11),   Integer32 (-2..2147483647)
912   INTEGER: Defines the parallel offset of the Finishing
913   Process Axis from the Reference Edge. For a value of
914   finEdgeTC equal to TopEdge or RightEdge, the value
915   given is to interpreted as a negative offset from the
916   reference edge. The units of measure are defined by the
917   attribute finReferenceEdgeTC.
918
919   finJogEdge(12),   FinEdgeTC
920   INTEGER: An enumeration which defines a second edge of the
921   document to which the media is aligned. The jog edge must
922   be perpendicular to the edge defined by finReferenceEdge.
923
924   finHeadLocation(13),   Integer32 (-2..2147483647)
925   INTEGER: MULTI-ROW: Defines the position of the Head
926   Mechanism relative to the axis, 'X' or 'Y', that is
927
928

                    perpendicular to the Process Axis. The units of measure
                    are defined by the attribute finProcessOffsetUnits.

          finOperationRestrictions(14),   Integer32 (0..65535)
             INTEGER: MULTI-ROW: Defines the finDeviceIndex of a
                finishing process which cannot be combined with the
                process defined by the finDeviceIndex for this
                finDeviceAttributeTable instance. When this condition
                occurs this attribute SHALL be presented in the
                attribute tables for both finishing processes that cannot
                be combined.

          finNumberOfPositions(15),   Integer32 (0..65535)
             INTEGER: Defines the total number of head positions for
                this finishing process. Each position many be realized by
                a unique head mechanism or a single head mechanism may be
                automatically moved to each position.

          namedConfiguration(16),   OCTET STRING (SIZE(0..63))
             OCTETS: Contains an administratively define name to define
                the finishing specification configured for this device.

          finMediaTypeRestriction(17),   OCTET STRING (SIZE(0..63))
             OCTETS: MULTI-ROW: Defines the media type which cannot be
                combined with the process defined by the finDeviceIndex
                for this finDeviceAttributeTable instance. Values are the
                same as defined for finSupplyMediaInputMediaName.

          finPrinterInputTraySupported(18), Integer32 (0..65535)
             INTEGER: MULTI-ROW: Defines the value of prtInputIndex
                corresponding to the printer input tray that can be used
                with the process defined by the finDeviceIndex for this
                finDeviceAttributeTable instance. If this attribute is
                not present, this process can be used with any input tray
                in the printer. For example, this attribute can indicate
                the current stapling capabilities for a stapler device
                for the input trays that depend upon the size and feed
                orientation. So if there were two letter trays, one with
                A size and the other with B size, a two position stapler
                might specify in one row: upper-left and upper-right for
                the input tray with A size, but only upper-left for the
                one with B size.

          finPreviousFinishingOperation(19),   Integer32 (0..65535)
             INTEGER: Defines the finDeviceIndex of the previous
                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 first in a

                    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.

```
     punchHoleSizeLongDim(81),         Integer32 (-2..2147483647)
        INTEGER:  Defines the size of the punched hole in the
          longest dimension.  This dimension is typically measured
          parallel to either the long edge or the short edge of the
          media and the longest dimension will always be measured 90
          degrees from the shortest dimension.  For a symmetrical
          hole, such as a round or square hole, the shortest and
          longest dimensions will be identical.

     punchHoleSizeShortDim(82),        Integer32 (-2..2147483647)
        INTEGER:  Defines the size of the punched hole in the
          shortest dimension.  This dimension is typically measured
          parallel to either the long edge or the short edge of the
          media and the shortest dimension will always be measured
          90 degrees from the longest dimension.  For a symmetrical
          hole, such as a round or square hole, the shortest and
          longest dimensions will be identical.

     punchPattern(83),                     FinPunchPatternTC
        INTEGER:  Defines the hole pattern produced by the punch
          process.

  H. Perforator type-specific attributes (90..)

  I. Slitter type-specific attributes (100..)

     slittingType(100),                    FinSlittingTypeTC
        INTEGER:  Provides additional information regarding the
          slitting process.

  J. Separation cutter type-specific attributes (110..)

  K. Imprinter type-specific attributes (120..)

  L. Wrapper type-specific attributes (130..)

     wrappingType(130),                    FinWrappingTypeTC
        INTEGER:  Provides additional information regarding the
          wrapping process.

  M. Bander type-specific attributes (140..)

  N. Make Envelopes type-specific attributes (150..)

  O. Stacker type-specific attributes (160..)

     stackOutputType(160)                  FinStackOutputTypeTC
        INTEGER:  Defines the job-to-job orientation produced by
          the stacker.
```

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

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

6.   ENUMERATIONS

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

6.1 Registering Additional Enumerated Values

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

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

   Some criteria that suggest using a type 1 enumeration are:

     a) the set of values in the enumeration is thought to be known,
          e.g., faceUp and faceDown

     b) the enumeration defines a set of units of measure which must
          be understood by a management application to be able to
          correctly display the value of an object that measurement unit
          controls;

   and

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

          to the OIDs for the related tables.

     There are no type 1 enums in the current draft.

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

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

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

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

7.   PRINTER FINISHING MIB SPECIFICATION

```
Finisher-MIB DEFINITIONS ::= BEGIN

IMPORTS
    mib-2                                      FROM RFC1213-MIB
    MODULE-IDENTITY, OBJECT-TYPE, Integer32,
    experimental                               FROM SNMPv2-SMI
    TEXTUAL-CONVENTION                         FROM SNMPv2-TC
    MODULE-COMPLIANCE, OBJECT-GROUP            FROM SNMPv2-CONF
    hrDeviceIndex                        FROM HOST-RESOURCES-MIB
    printmib, PrtMarkerSuppliesTypeTC,
    PrtSubUnitStatusTC, PrtInputTypeTC,
    PrtMarkerSuppliesSupplyUnitTC, PrtMediaUnitTC,
    PrtCapacityUnitTC, PrtMarkerSuppliesClassTC,
    PresentOnOff, prtMIBConformance           FROM Printer-MIB;

finisherMIB MODULE-IDENTITY
    LAST-UPDATED "9810090000Z"
    ORGANIZATION "IETF Printer MIB Working Group"
    CONTACT-INFO
        "Ron Bergman
         Dataproducts Corp.
         1757 Tapo Canyon Road
         Simi Valley, CA 91063-3394
         rbergma@dpc.com

        Send comments to the printmib WG using the Finisher MIB
        Project (FIN) Mailing List:  fin@pwg.org

        For further information, access the PWG web page under 'FIN':
              http://www.pwg.org/

        Implementers of this specification are encouraged to join the
        fin mailing list in order to participate in discussions on any
        clarifications needed and registration proposals being reviewed
        in order to achieve consensus."
    DESCRIPTION
        "The MIB module for management of printer finisher units.
         The Finisher MIB is an extension of the Printer MIB."
     ::= { experimental 54 }

  -- Note to RFC editor, experimental 54 was assigned to the Printer
  -- MIB.  A new arc in the experimental subtree must be assigned.


-- Textual conventions for this MIB module


FinDeviceTypeTC ::= TEXTUAL-CONVENTION
-- This is a type 2 enumeration.
    STATUS         current
    DESCRIPTION
```

```
        "The defined finishing device subunit process
         enumerations."
     SYNTAX        INTEGER {
         other(1),
         unknown(2),
         stitcher(3),
         folder(4),
         binder(5),
         trimmer(6),
         dieCutter(7),
         puncher(8),
         perforater(9),
         slitter(10),
         separationCutter(11),
         imprinter(12),
         wrapper(13),
         bander(14),
         makeEnvelope(15),
         stacker(16),
         sheetRotator(17)
            }

 FinAttributeTypeTC  ::= TEXTUAL-CONVENTION
 -- This is a type 3 enumeration.
     STATUS     current
     DESCRIPTION
         "This textual convention defines the set of enums for use in
          the finDeviceAttributeTable.  See section 5.7 for the complete
          specification of each attribute."
     SYNTAX        INTEGER {
         other(1),
         deviceName(3),
         deviceVendorName(4),
         deviceModel(5),
         deviceVersion(6),
         deviceSerialNumber(7),
         maximumSheets(8),
         finProcessOffsetUnits(9),
         finReferenceEdge(10),
         finAxisOffset(11),
         finJogEdge(12),
         finHeadLocation(13),
         finOperationRestrictions(14),
         finNumberOfPositions(15),
         namedConfiguration(16),
         finMediaTypeRestriction(17),
         finPrinterInputTraySupported(18),
         finPreviousFinishingOperation(19),
         finNextFinishingOperation(20),
         stitchingType(30),
         stitchingDirection(31),
```

```
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
```

          of the stack. This parameter can be used to determine what
          order the pages of a booklet are to be printed such that the
          staple clinch will be on the inside of the resulting booklet."
      SYNTAX   INTEGER {
          unknown(2),
          topDown(3),
          bottomUp(4)
          }

StitchingAngleTypeTC ::= TEXTUAL-CONVENTION
-- This is a type 2 enumeration.
      STATUS   current
      DESCRIPTION
        "This enumeration provides a description of the angular
        orientation of each stitch in a single or multiple stitching
        operation, relative to the 'X' axis. As with all finishing
        operations, the 'X' axis is always relative to the portrait
        orientation of the document regardless of the orientation
        of the printed image. This enum is primarily applicable to
        corner stitching operations."
      SYNTAX   INTEGER {
          unknown(2),
          horizontal(3),
          vertical(4),
          slanted(5)
          }

FinFoldingTypeTC ::= TEXTUAL-CONVENTION
--This is a type 2 enumeration.
      STATUS   current
      DESCRIPTION
        "The defined folding device process enumerations."
      SYNTAX   INTEGER {
          other(1),   -- More information in other attributes
          unknown(2),
          zFold(3),
          halfFold(4),
          letterFold(5)
          }

FinBindingTypeTC ::= TEXTUAL-CONVENTION
-- This is a type 2 enumeration.
      STATUS   current
      DESCRIPTION
        "The defined binding type enumerations."
      SYNTAX   INTEGER {
          other(1),   -- More information in other attributes
          unknown(2),
          tape(4),
          plastic(5),
          velo(6),

```
            perfect(7),
            spiral(8),
            adhesive(9),
            comb(10),
            padding(11)
            }

FinPunchHoleTypeTC ::= TEXTUAL-CONVENTION
--This is a type 2 enumeration.
   STATUS        current
   DESCRIPTION
     "The defined hole type punch process enumerations."
   SYNTAX        INTEGER {
         other(1),          -- More information in other attributes
         unknown(2),
         round(3),
         oblong(4),
         square(5),
         rectangular(6),
         star(7)
         }

FinPunchPatternTC ::= TEXTUAL-CONVENTION
--This is a type 2 enumeration.
   STATUS        current
   DESCRIPTION
     "The defined hole pattern punch process enumerations."
   SYNTAX        INTEGER {
         other(1),          --Pattern to be defined in other attributes
         unknown(2),
         twoHoleUSTop(4),    --Letter/legal, 8.5 inch edge
         threeHoleUS(5),     --Letter/ledger, 11 inch edge
         twoHoleDIN(6),      --A4/A3, 297 mm edge
         fourHoleDIN(7),     --A4/A3, 297 mm edge
         twentyTwoHoleUS(8), --Letter/ledger, 11 inch edge
         nineteenHoleUS(9),  --Letter/ledger, 11 inch edge
         twoHoleMetric(10),  --B5/B4, 257 mm edge
         swedish4Hole(11),   --A4/A3, 297 mm edge
         twoHoleUSSide(12),  --Letter/ledger, 11 inch edge
         fiveHoleUS(13),     --Letter/ledger, 11 inch edge
         sevenHoleUS(14),    --Letter/ledger, 11 inch edge
         mixed7H4S(15),      --A4/A3, 297 mm edge
         norweg6Hole(16),    --A4/A3, 297 mm edge
         metric26Hole(17),   --B5/B4, 257 mm edge
         metric30Hole(18)    --A4/A3, 297 mm edge
         }

FinSlittingTypeTC ::= TEXTUAL-CONVENTION
-- This is a type 2 enumeration.
    STATUS        current
    DESCRIPTION
```

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

```
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
```

          and whether the device subunit is enabled."
       DEFVAL   { notPresent }
       ::= { finDeviceEntry 3 }

   finDeviceCapacityUnit OBJECT-TYPE
       SYNTAX   PrtCapacityUnitTC
       MAX-ACCESS   read-only
       STATUS   current
       DESCRIPTION
         "The unit of measure for specifying the capacity of this
          finisher device subunit."
       ::= { finDeviceEntry 4 }

   finDeviceMaxCapacity OBJECT-TYPE
       SYNTAX   Integer32 (-2..2147483647)
       MAX-ACCESS   read-write
       STATUS   current
       DESCRIPTION
         "The maximum capacity of this finisher device subunit in
          finDeviceCapacityUnits. If the device can reliably sense
          this value, the value is sensed by the finisher device
          and is read-only: otherwise the value may be written by a
          management or control console application. The value (-1)
          means other and specifically indicates that the device
          places no restrictions on this parameter. The value (-2)
          means unknown."
       DEFVAL   { -2 }   -- unknown
       ::= { finDeviceEntry 5 }

   finDeviceCurrentCapacity OBJECT-TYPE
       SYNTAX   Integer32 (-2..2147483647)
       MAX-ACCESS   read-write
       STATUS   current
       DESCRIPTION
         "The current capacity of this finisher device subunit in
          finDeviceCapacityUnits. If the device can reliably sense
          this value, the value is sensed by the finisher and is
          read-only: otherwise the value may be written by a
          management or control console application. The value (-1)
          means other and specifically indicates that the device
          places no restrictions on this parameter. The value (-2)
          means unknown."
       DEFVAL   { -2 }   -- unknown
       ::= { finDeviceEntry 6 }

   finDeviceAssociatedMediaPaths OBJECT-TYPE
       SYNTAX   OCTET STRING
       MAX-ACCESS   read-only
       STATUS   current
       DESCRIPTION
         "Indicates the media paths which can supply media for this

         finisher device. The value of this object is a bit map in an
         octet string with each position representing the value of a
         prtMediaPathIndex. For a media path that can be a source
         for this finisher device subunit, the bit position equal
         to one less than the value of prtMediaPathIndex will be set.
         The bits are numbered starting with the most significant bit of
         the first byte being bit 0, the least significant bit of the
         first byte being bit 7, the most significant of the second byte
         being bit 8, and so on."
    ::= { finDeviceEntry 7 }

finDeviceAssociatedOutputs OBJECT-TYPE
    SYNTAX   OCTET STRING
    MAX-ACCESS read-only
    STATUS   current
    DESCRIPTION
      "Indicates the printer output subunits this finisher device
        subunit services. The value of this object is a bit map in an
        octet string with each position representing the value of a
        prtOutputIndex. For an output subunit that is serviced
        by this finisher device subunit, the bit position equal to
        to one less than the value of prtOutputIndex will be set.
        The bits are numbered starting with the most significant bit of
        the first byte being bit 0, the least significant bit of the
        first byte being bit 7, the most significant of the second byte
        being bit 8, and so on."
    ::= { finDeviceEntry 8 }

finDeviceStatus OBJECT-TYPE
    SYNTAX   PrtSubUnitStatusTC
    MAX-ACCESS read-only
    STATUS   current
    DESCRIPTION
      "Indicates the current status of this finisher device
        subunit."
    DEFVAL   { 5 }   -- unknown
    ::= { finDeviceEntry 9 }

finDeviceDescription OBJECT-TYPE
    SYNTAX   OCTET STRING (SIZE(0..255))
    MAX-ACCESS read-only
    STATUS   current
    DESCRIPTION
      "A free form text description of this device subunit in the
        localization specified by prtGeneralCurrentLocalization."
    ::= { finDeviceEntry 10 }

-- Finisher Supply Group (Mandatory)
--
-- A finisher device, but not all finisher devices, may have one or more

-- supplies associated with it. For example a finisher may use both
-- binding tape and stitching wire supplies. A finisher may also have
-- more than one source for a given type of supply e.g. multiple supply
-- sources of ink for imprinters.

finSupply OBJECT IDENTIFIER ::= { printmib 31 }

finSupplyTable OBJECT-TYPE
    SYNTAX   SEQUENCE OF FinSupplyEntry
    MAX-ACCESS not-accessible
    STATUS   current
    DESCRIPTION
      "Each unique source of supply is an entry in the finisher
       supply table. Each supply entry has its own
       characteristics associated with it such as colorant and
       current supply level."
    ::= { finSupply 1 }

finSupplyEntry OBJECT-TYPE
    SYNTAX   FinSupplyEntry
    MAX-ACCESS not-accessible
    STATUS   current
    DESCRIPTION
      "A list of finisher devices, with their associated
       supplies and supplies characteristics."
    INDEX { hrDeviceIndex, finSupplyIndex }
    ::= { finSupplyTable 1 }

FinSupplyEntry ::= SEQUENCE {
        finSupplyIndex   Integer32,
        finSupplyDeviceIndex   Integer32,
        finSupplyClass   PrtMarkerSuppliesClassTC,
        finSupplyType   PrtMarkerSuppliesTypeTC,
        finSupplyDescription   OCTET STRING,
        finSupplyUnit   PrtMarkerSuppliesSupplyUnitTC,
        finSupplyMaxCapacity   Integer32,
        finSupplyCurrentLevel   Integer32,
        finSupplyColorName   OCTET STRING
       }

finSupplyIndex OBJECT-TYPE
    SYNTAX   Integer32 (1..65535)
    MAX-ACCESS not-accessible
    STATUS   current
    DESCRIPTION
      "A unique value used by a finisher to identify this supply
       container/receptacle. Although these values may change
       due to a major reconfiguration of the finisher (e.g. the
       addition of new supply sources to the finisher), values
       are normally expected to remain stable across successive
       power cycles."

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

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

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

--   Capacity Attribute Relationships
--
--   MEDIA INPUT MEASUREMENT
--
--   _____ | |
-- | | | |
-- | | | | |
-- | | |_ _ _ _ _ _ _ _ _ _| _____ |direction
-- | | | | | v
-- MaxCapacity | | | |
-- | | | Sheets remaining | CurrentLevel
-- | | | | |
--   v | | | v
--   _____ +_____+ _____
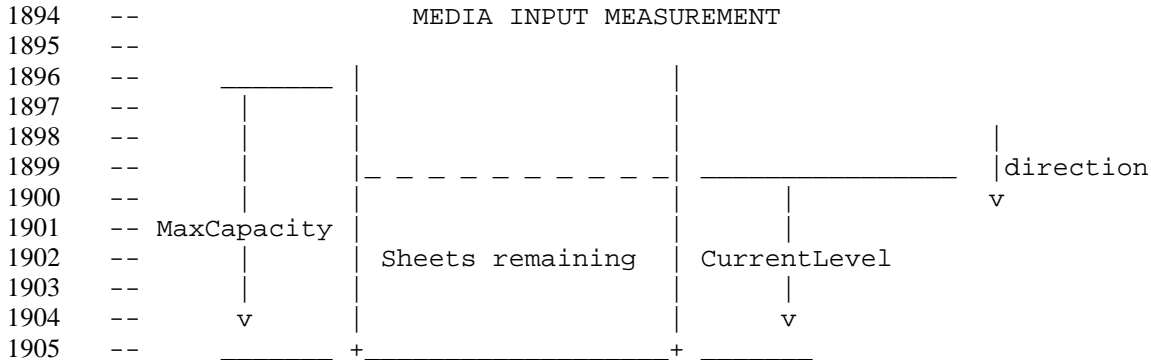
finSupplyColorName OBJECT-TYPE
        SYNTAX   OCTET STRING (SIZE(0..63))
        MAX-ACCESS read-only
        STATUS   current
        DESCRIPTION

          "The name of the color of this colorant using standardized
           string names from ISO 10175 (DPA) and ISO 10180 (SPDL)
           which are: other, unknown, white, red, green, blue, cyan,
           magenta, yellow and black. Implementors may add
           additional string values. The naming conventions in ISO
           9070 are recommended in order to avoid potential name
           clashes."
       ::= { finSupplyEntry 9 }

-- Finisher Supply, Media Input Group (Conditionally Mandatory)
--
-- A finisher device may have one or more associated supply media
-- inputs. Each entry in this table defines an input for a
-- supply media type such as inserts, covers, etc.
--
-- This group is mandatory only if the printer system contains a
-- finisher device that requires a media supply used exclusively by a
-- finishing process. Examples are inserts or covers that are not
-- supplied by an input subunit that provides media to the marker.

finSupplyMediaInput OBJECT IDENTIFIER ::= { printmib 32 }

finSupplyMediaInputTable OBJECT-TYPE
    SYNTAX   SEQUENCE OF FinSupplyMediaInputEntry
    MAX-ACCESS not-accessible
    STATUS   current
    DESCRIPTION
      "The input subunits associated with a finisher supply media
       are each represented by an entry in this table."
 ::= { finSupplyMediaInput 1 }

finSupplyMediaInputEntry OBJECT-TYPE
    SYNTAX   FinSupplyMediaInputEntry
    MAX-ACCESS not-accessible
    STATUS   current
    DESCRIPTION
      "A list of finisher supply media input subunit features and
       characteristics."
    INDEX { hrDeviceIndex, finSupplyMediaInputIndex }
    ::= { finSupplyMediaInputTable 1 }

FinSupplyMediaInputEntry ::= SEQUENCE {
    finSupplyMediaInputIndex   Integer32,
    finSupplyMediaInputDeviceIndex   Integer32,
    finSupplyMediaInputSupplyIndex   Integer32,
    finSupplyMediaInputType   PrtInputTypeTC,
    finSupplyMediaInputDimUnit   PrtMediaUnitTC,
    finSupplyMediaInputMediaDimFeedDir Integer32,
    finSupplyMediaInputMediaDimXFeedDir Integer32,
    finSupplyMediaInputStatus   PrtSubUnitStatusTC,

```
        finSupplyMediaInputMediaName        OCTET STRING,
        finSupplyMediaInputName             OCTET STRING,
        finSupplyMediaInputDescription      OCTET STRING,
        finSupplyMediaInputSecurity         PresentOnOff,
        finSupplyMediaInputMediaWeight      Integer32,
        finSupplyMediaInputMediaThickness   Integer32,
        finSupplyMediaInputMediaType        OCTET STRING
      }

finSupplyMediaInputIndex OBJECT-TYPE
    SYNTAX       Integer32 (1..65535)
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
      "A unique value used by a finisher to identify this supply
       media input subunit.  Although these values may change
       due to a major reconfiguration of the finisher (e.g. the
       addition of new supply media input sources to the
       finisher), values are normally expected to remain stable
       across successive power cycles."
    ::= { finSupplyMediaInputEntry 1 }

finSupplyMediaInputDeviceIndex OBJECT-TYPE
    SYNTAX       Integer32 (0..65535)
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
      "The value of finDeviceIndex corresponding to the finishing
       device subunit with which this finisher media supply is
       associated."
    ::= { finSupplyMediaInputEntry 2 }

finSupplyMediaInputSupplyIndex OBJECT-TYPE
    SYNTAX       Integer32 (0..65535)
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
      "The value of finSupplyIndex corresponding to the finishing
       supply subunit with which this finisher media supply is
       associated."
    ::= { finSupplyMediaInputEntry 3 }

finSupplyMediaInputType OBJECT-TYPE
    SYNTAX       PrtInputTypeTC
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
      "The type of technology (discriminated primarily according
       to the feeder mechanism type) employed by the input
       subunit."
    ::= { finSupplyMediaInputEntry 4 }
```

finSupplyMediaInputDimUnit OBJECT-TYPE
    SYNTAX   PrtMediaUnitTC
    MAX-ACCESS   read-only
    STATUS   current
    DESCRIPTION
      "The unit of measure for specifying dimensional values for
        this input device."
    ::= { finSupplyMediaInputEntry 5 }

finSupplyMediaInputMediaDimFeedDir OBJECT-TYPE
    SYNTAX   Integer32 (-2..2147483647)
    MAX-ACCESS   read-write
    STATUS   current
    DESCRIPTION
      "This object provides the value of the dimension in the
        feed direction of the media that is placed or will be
        placed in this input device. Feed dimension measurements
        are taken parallel to the feed direction of the device and
        measured in finSupplyMediaInputDimUnits. If this input
        device can reliably sense this value, the value is sensed
        and is read-only access. Otherwise the value is read-write
        access and may be written by management or control panel
        applications. The value (-1) means other and specifically
        indicates that this device places no restrictions on this
        parameter. The value (-2) indicates unknown. "
    ::= { finSupplyMediaInputEntry 6 }

finSupplyMediaInputMediaDimXFeedDir OBJECT-TYPE
    SYNTAX   Integer32 (-2..2147483647)
    MAX-ACCESS   read-write
    STATUS   current
    DESCRIPTION
      "This object provides the value of the dimension across the
        feed direction of the media that is placed or will be
        placed in this input device. The cross feed direction is
        ninety degrees relative to the feed direction on this
        device and measured in finSupplyMediaInputDimUnits. If
        this input device can reliably sense this value, the value
        is sensed and is read-only access. Otherwise the value is
        read-write access and may be written by management or
        control panel applications. The value (-1) means other and
        specifically indicates that this device places no
        restrictions on this parameter. The value (-2) indicates
        unknown. "
    ::= { finSupplyMediaInputEntry 7 }

finSupplyMediaInputStatus OBJECT-TYPE
    SYNTAX   PrtSubUnitStatusTC
    MAX-ACCESS   read-only
    STATUS   current

        DESCRIPTION
          "This value indicates the current status of this input
           device."
        DEFVAL   { 5 }   -- unknown
        ::= { finSupplyMediaInputEntry 8 }

    finSupplyMediaInputMediaName OBJECT-TYPE
        SYNTAX   OCTET STRING (SIZE(0..63))
        MAX-ACCESS read-write
        STATUS   current
        DESCRIPTION
          "The name of the current media contained in this input
           device. Examples are Engineering Manual Cover, Section A Tab
           Divider or any ISO standard names."
        ::= { finSupplyMediaInputEntry 9 }

    finSupplyMediaInputName OBJECT-TYPE
        SYNTAX   OCTET STRING (SIZE(0..63))
        MAX-ACCESS read-write
        STATUS   current
        DESCRIPTION
          "The name assigned to this input subunit."
        ::= { finSupplyMediaInputEntry 10 }

    finSupplyMediaInputDescription OBJECT-TYPE
        SYNTAX   OCTET STRING (SIZE(0..255))
        MAX-ACCESS read-only
        STATUS   current
        DESCRIPTION
          "A free form text description of this input subunit in the
           localization specified by prtGeneralCurrentLocalization."
        ::= { finSupplyMediaInputEntry 11 }

    finSupplyMediaInputSecurity OBJECT-TYPE
        SYNTAX   PresentOnOff
        MAX-ACCESS read-write
        STATUS   current
        DESCRIPTION
          "Indicates if this subunit has some security associated
           with it."
        ::= { finSupplyMediaInputEntry 12 }

    finSupplyMediaInputMediaWeight OBJECT-TYPE
        SYNTAX   Integer32
        MAX-ACCESS read-write
        STATUS   current
        DESCRIPTION
          "The weight of the media associated with this Input device
           in grams per meter squared. The value (-1) means other
           and specifically indicates that the device places no
           restriction on this parameter. The value (-2) means

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

```
     STATUS     current
     DESCRIPTION
       "The attribute table defines special parameters that are
        applicable only to a minority of the finisher devices.
        An attribute table entry is used, rather than unique
        objects, to minimize the number of MIB objects and to
        allow for expansion without the addition of MIB objects.
        Each finisher device is represented by a separate row
        in the device subunit attribute table."
     ::= { finDeviceAttribute 1 }

finDeviceAttributeEntry OBJECT-TYPE
     SYNTAX     FinDeviceAttributeEntry
     MAX-ACCESS not-accessible
     STATUS     current
     DESCRIPTION
       "Each entry defines a finisher function parameter that
        cannot be represented by an object in the finisher
        device subunit table."
     INDEX  { hrDeviceIndex, finDeviceIndex,
              finDeviceAttributeTypeIndex,
              finDeviceAttributeInstanceIndex }
     ::= { finDeviceAttributeTable 1 }

FinDeviceAttributeEntry ::= SEQUENCE {
     finDeviceAttributeTypeIndex       FinAttributeTypeTC,
     finDeviceAttributeInstanceIndex   Integer32,
     finDeviceAttributeValueAsInteger  Integer32,
     finDeviceAttributeValueAsOctets   OCTET STRING
     }

finDeviceAttributeTypeIndex OBJECT-TYPE
     SYNTAX     FinAttributeTypeTC
     MAX-ACCESS  not-accessible
     STATUS     current
     DESCRIPTION
       "Defines the attribute type represented by this row."
     ::= { finDeviceAttributeEntry 1 }

finDeviceAttributeInstanceIndex OBJECT-TYPE
     SYNTAX     Integer32 (1..65535)
     MAX-ACCESS  not-accessible
     STATUS     current
     DESCRIPTION
       "An index that allows the discrimination of an attribute
        instance when the same attribute occurs multiple times for
        a specific instance of a finisher function.  The value of
        this index shall be 1 if only a single instance of the
        attribute occurs for the specific finisher function."
     ::= { finDeviceAttributeEntry 2 }
```

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

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]

```
        DESCRIPTION
            "It is conformant to implement this object as read-only."

        OBJECT     finSupplyMediaInputMediaThickness
        MIN-ACCESS read-only
        DESCRIPTION
            "It is conformant to implement this object as read-only."

        OBJECT     finSupplyMediaInputMediaType
        MIN-ACCESS read-only
        DESCRIPTION
            "It is conformant to implement this object as read-only."

        OBJECT     finDeviceAttributeValueAsInteger
        MIN-ACCESS read-only
        DESCRIPTION
            "It is conformant to implement this object as read-only."

        OBJECT     finDeviceAttributeValueAsOctets
        MIN-ACCESS read-only
        DESCRIPTION
            "It is conformant to implement this object as read-only."

        GROUP      finSupplyMediaInputGroup
        DESCRIPTION
            "This group is conditionally mandatory and must be included
             if a finisher device requires a media supply that is used
             exclusively by a finishing process."

    ::= { prtMIBConformance 3 }


finMIBGroups      OBJECT IDENTIFIER ::= { prtMIBConformance 4 }

finDeviceGroup OBJECT-GROUP
    OBJECTS { finDeviceType, finDevicePresentOnOff,
              finDeviceCapacityUnit, finDeviceMaxCapacity,
              finDeviceCurrentCapacity, finDeviceAssociatedMediaPaths,
              finDeviceAssociatedOutputs, finDeviceStatus,
              finDeviceDescription }
    STATUS  current
    DESCRIPTION
        "The finisher device group."
    ::= { finMIBGroups 1 }

finSupplyGroup OBJECT-GROUP
    OBJECTS { finSupplyDeviceIndex, finSupplyClass, finSupplyType,
              finSupplyDescription, finSupplyUnit, finSupplyMaxCapacity,
              finSupplyCurrentLevel, finSupplyColorName }
    STATUS  current
    DESCRIPTION
```

        "The finisher supply group."
     ::= { finMIBGroups 2 }

finSupplyMediaInputGroup OBJECT-GROUP
    OBJECTS { finSupplyMediaInputDeviceIndex,
                finSupplyMediaInputSupplyIndex, finSupplyMediaInputType,
                finSupplyMediaInputDimUnit,
                finSupplyMediaInputMediaDimFeedDir,
                finSupplyMediaInputMediaDimXFeedDir,
                finSupplyMediaInputStatus, finSupplyMediaInputMediaName,
                finSupplyMediaInputName, finSupplyMediaInputDescription,
                finSupplyMediaInputSecurity,
                finSupplyMediaInputMediaWeight,
                finSupplyMediaInputMediaThickness,
                finSupplyMediaInputMediaType }
    STATUS current
    DESCRIPTION
        "The finisher supply, media input group."
     ::= { finMIBGroups 3 }

finDeviceAttributeGroup OBJECT-GROUP
    OBJECTS { finDeviceAttributeValueAsInteger,
                finDeviceAttributeValueAsOctets }
    STATUS current
    DESCRIPTION
        "The finisher device attribute group."
     ::= { finMIBGroups 4 }

END

8. REFERENCES

   [PRTMIB] The Printer MIB, RFC XXXX, IETF standards track document.

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
        Requirement Levels", BCP 14, RFC 2119, March 1997.

   [LMO] Large Mailing Operations Specification, DMTF. See
        http://www.dmtf.org/tech/apps.html

   [DPA] ISO/IEC 10175 Document Printing Application (DPA). See
        ftp://ftp.pwg.org/pub/pwg/dpa/

   [IPP] Internet Printing Protocol/1.0: Model and Semantics, RFC 2566.

9. SECURITY CONSIDERATIONS

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

10. FULL COPYRIGHT STATEMENT

11. AUTHORS

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

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

    Phone: 805-578-4421
    Fax: 805-578-4001
    Email: rbergma@hitachi-hkis.com

    Harry Lewis (chairman)
    IBM Corporation
    6300 Diagonal Hwy
    Boulder, CO 80301

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

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

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

    For further information, access the PWG web page under "FIN":
    http://www.pwg.org/

    Other Participants:

    Chuck Adams - Tektronix
    Carlos Becerra - HP
    Andy Davidson - Tektronix
    Mabry Dozier - QMS
    Lee Farrell - Canon
    Jennifer Gattis - Duplo USA
    Paul Gloger - Xerox
    Richard Hart - Digital
    Tom Hastings - Xerox
    Scott Isaacson - Novell
    David Kellerman - Northlake Software

        Henrik Holst - i-data International
        Rick Landau - Digital
        Jay Martin - Underscore
        Gary Padlipski - Xerox
        Kevin Palmer - Duplo USA
        Bob Pentecost - HP
        Stuart Rowley - Kyocera
        Yuki Sacchi - Japan Computer Industry
        Philip Thambidunai - Okidata
        William Wagner - DPI/Osicom
        Chris Wellens - Interworking Labs
        Don Wright - Lexmark
        Lloyd Young - Lexmark