Network Working Group
Request for Comments: 3806
Category: Informational

R. Bergman
Hitachi Printing Solutions
H. Lewis
IBM Corporation
I. McDonald
High North Inc.

June 2004

Printer Finishing MIB

Status of this Memo

This memo provides information for the Internet community. It does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

Copyright Notice

Abstract

This document defines a MIB module 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 applies only to a Finisher Device that is connected to a Printer System.

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 [RFC3805]. 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.

The Finisher MIB is defined as an extension of the Printer MIB [RFC3805] and it is expected that the information defined in this document will be incorporated into a future update of the Printer MIB.

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 [RFC3805] 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.

1.3. The Internet-Standard Management Framework

For a detailed overview of the documents that describe the current Internet-Standard Management Framework, please refer to section 7 of RFC 3410 [RFC3410].

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. MIB objects are generally accessed through the Simple Network Management Protocol (SNMP). Objects in the MIB are defined using the mechanisms defined in the Structure of Management Information (SMI). This memo specifies a MIB module that is compliant to the SMIv2, which is described in STD 58, RFC 2578 [RFC2578], STD 58, RFC 2579 [RFC2579] and STD 58, RFC 2580 [RFC2580].

1.4. Read-Write Objects

Some objects in the Finisher MIB reflect the existence or amount of a given resource within the finisher. Some examples of such resources are the size and number of sheets in an inserter tray or the existence of certain finisher options. Some finishers have automatic sensors for these resources. Most finishers lack sensors for every property of every resource. The management application is allowed to write into objects that hold descriptive or existence values for finishers that cannot sense these values. The ability to change the value of a read-write object may depend on the implementation of the agent. Many objects in the MIB are given read-write access, but an implementation might only permit a management application to change

the value if the finisher can not sense the value itself. Note that even though some objects explicitly state the behavior of conditional ability to change values, any read-write object may act this way.

Generally, an object is given read-write access in the Finisher MIB specification if:

1. The object involves installation of a resource that some finishers cannot themselves detect. Therefore, external means are needed to inform the device of the installation. (Here external means include using the operator console, or remote management application) and

2. The finisher will behave differently if the installation of the resource is reported than if the installation were not reported; that is, the object is not to be used as a place to put information not used by the finisher, i.e., not a "sticky-note". Another way of saying this is that the finisher believes that information given it and acts as if the information were true.

3. The finisher may get hints that it may not know about the existence or properties of certain resources. For example, a paper tray may be removed and re-inserted. When this removal and insertion happens, the finisher may either assume that a property, such as the size of paper in the tray, has not changed or the finisher may change the value of the associated object to "unknown", as might be done for the amount of paper in the tray. As long as the finisher acts according to the value in the object either strategy is acceptable.

4. It is an implementation-specific matter as to whether or not MIB object values are persistent across power cycles or cold starts.

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 [RFC2911], 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".

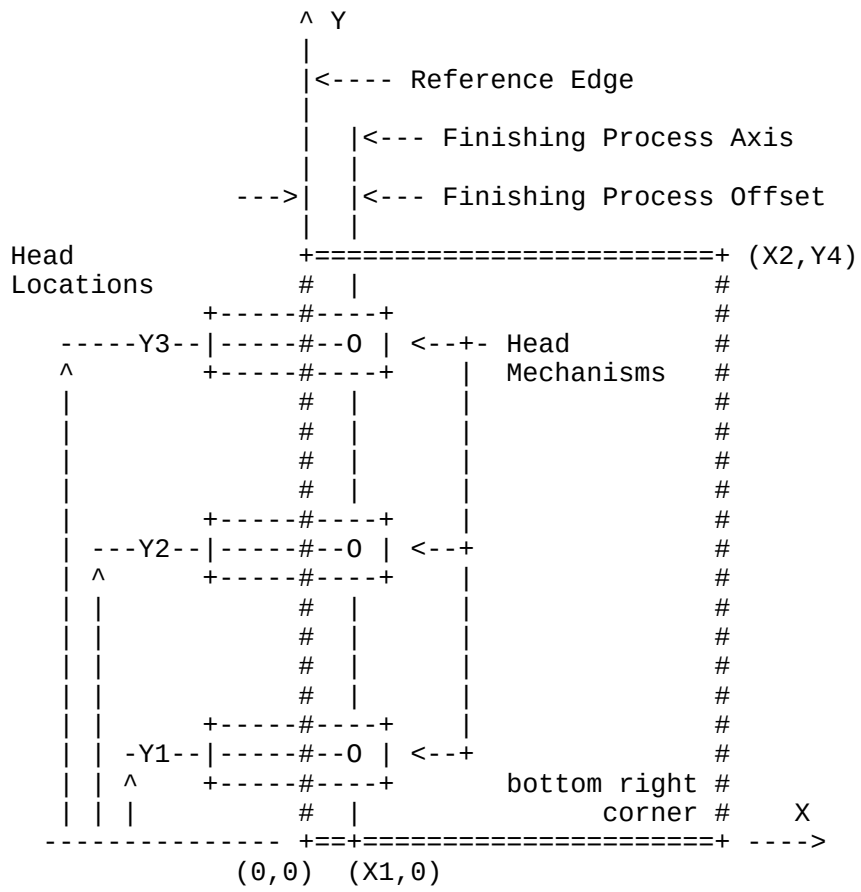


Figure 1 - Finishing Process Axis Parallel to Y Axis

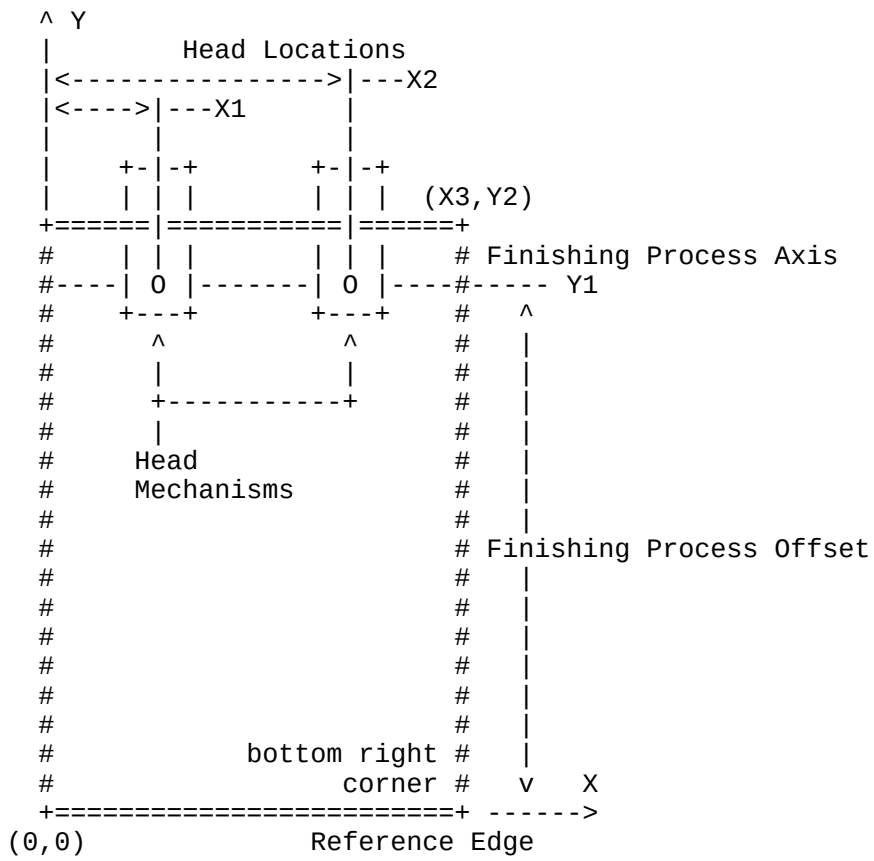


Figure 2 - Finishing Process Axis Parallel to X Axis

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.

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 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 of 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:

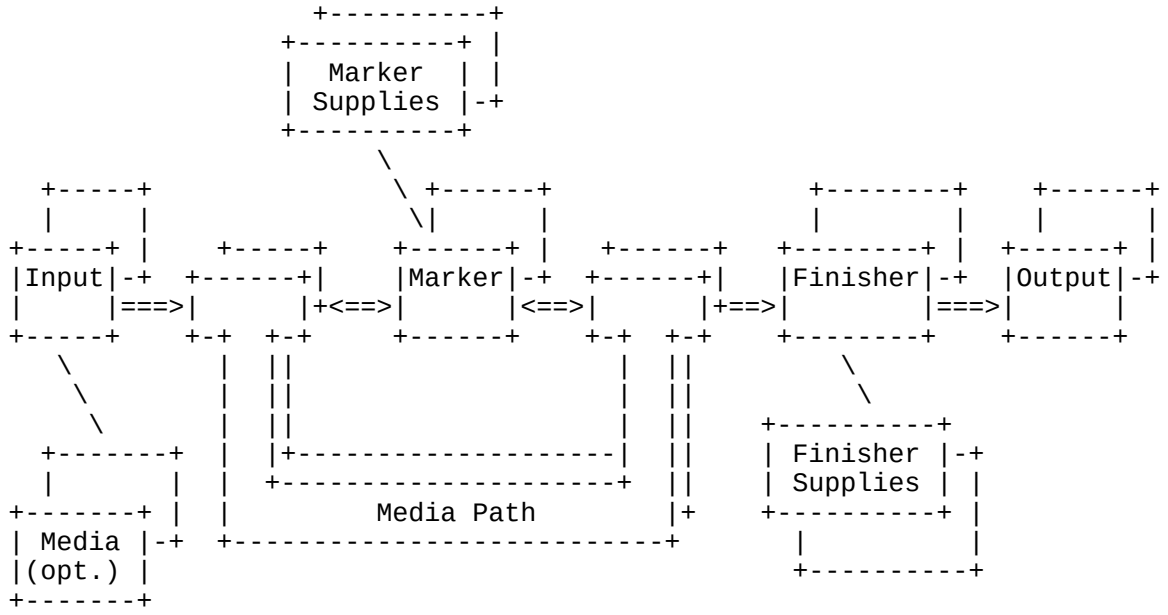
Band Wrap: Bundling a finished stack to prepare for shipment. Also known as Strap Wrap.

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 [RFC3805], is modified adding the finisher subunit(s) and finisher supplies between the media path and output subunits as follows:



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 5	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 create 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.

NOTE: In some instances, objects with a MAX-ACCESS of read-write will result in an SNMPv1 error or SNMPv2 exception during a write operation. The administrative security policy may restrict a class of users to read-only or, more importantly, the implementation may implement a subset of read-write objects as read-only. This should be expected to be the case for a device that can properly sense the value of an object and does not want the value to be externally modified.

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

Since each attribute is represented by a row consisting of both the finDeviceAttributeValueAsInteger and finDeviceAttributeValueAsOctets MANDATORY objects, SNMP requires that the agent SHALL always create an attribute row with both objects specified. However, for most attributes the agent SHALL return a "useful" value for one of the objects and SHALL return the 'other' value for the other object. For integer only attributes, the agent SHALL always return a zero-length string value for the finDeviceAttributeValueAsOctets object. For octet string only attributes, the agent SHALL always return a '-1' value for the finDeviceAttributeValueAsInteger object.

5.3. Data Sub-types and Attribute Naming Conventions

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

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

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

5.5. Linked MUTI-ROW Values

Some MULTI-ROW attributes are intended to go together. Thus a set of value instances represent a single instance. For example, the puncher attributes indicate the location, maximum size, minimum size and shape of the various holes that the puncher can produce. So the first set of values could represent one kind of hole, and the second set another kind of hole, etc.

5.6. Index Value Attributes

A number of attributes are indexes in other tables. Such attribute names end with the word 'Index'. If the agent has not (yet) assigned an index value for a particular index attribute for a finishing subunit, the agent shall either: (1) return the value 0 or (2) not add this attribute to the finDeviceAttributeTable until the index

value is assigned. In the interests of brevity, the semantics for 0 is specified once here and is not repeated for each index attribute specification and a DEFVAL of 0 is indicated.

5.7. Attribute Specifications

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

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

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

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

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.

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, finHeadLocation, punchHoleSizeLongDim, and punchHoleSizeShortDim.

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 offset of the Finishing Process Axis from the parallel 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 finProcessOffsetUnits.

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 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), FinStitchingDirTypeTC
INTEGER: Defines the orientation of the stitching process.

stitchingAngle(32), FinStitchingAngleTypeTC
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. The units of measure are defined by the attribute finProcessOffsetUnits.

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. The units of measure are defined by the attribute finProcessOffsetUnits.

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. Commonly used enumeration sets are assigned a symbolic data type name (textual convention), rather than being specified in the SYNTAX clause of each individual object definition.

Textual conventions defined in the Finisher MIB or the companion IANA Finisher MIB are extensible by RFC publication or Designated Expert Review (see 'IANA Considerations' section of this Finisher MIB and the DESCRIPTION clause in MODULE-IDENTITY of IANA Finisher MIB). All of these textual conventions are:

a) used more than once in the Finisher MIB itself; or

b) imported and used in any other, including vendor private, MIB modules.

The Finisher MIB has also defined the following special values for use with objects of the syntax "Integer32" to define conditions that are outside of the normal numeric range: other(-1), unknown(-2), and partial(-3). The 'partial' value means that there is some supply remaining (but the amount is indeterminate) or there is some capacity remaining (but the amount is indeterminate). The Integer32 range field indicates in which objects these special values are valid.

6.1. Registering Additional Enumerated Values

The Finisher MIB and the companion IANA Finisher MIB each defines one category of textual convention, according to the process employed to control the addition of new enumerations:

Type 1 - All of the legal values are defined in the Finisher MIB. Additional enumerated values require the publication of a new Finisher MIB.

Type 2 - All of the legal values are registered in the IANA Finisher MIB. Additional enumerated values require a Designated Expert Review defined in "Guidelines for Writing an IANA Considerations Section in RFCs" [RFC2434]. The Designated Expert will be selected by the IETF Area Director(s) of the Applications Area.

7. IANA Printer Finishing MIB Specification

```
IANA-FINISHER-MIB DEFINITIONS ::= BEGIN
 -- http://www.iana.org/assignments/ianafinisher-mib

IMPORTS
    MODULE-IDENTITY,
        mib-2
            FROM SNMPv2-SMI                                 -- [RFC2578]
    TEXTUAL-CONVENTION
            FROM SNMPv2-TC;                                 -- [RFC2579]

ianafinisherMIB MODULE-IDENTITY
    LAST-UPDATED  "200406020000Z" -- June 2, 2004
    ORGANIZATION  "IANA"
    CONTACT-INFO  "Internet Assigned Numbers Authority

                   Postal: ICANN
                           4676 Admiralty Way, Suite 330
                           Marina del Rey, CA 90292

                   Tel:    +1 310 823 9358
                   E-Mail: iana@iana.org"

    DESCRIPTION   "This MIB module defines a set of finishing-related
                   TEXTUAL-CONVENTIONs for use in Finisher MIB (RFC 3806)
                   and other MIBs which need to specify finishing
                   mechanism details.

                   Any additions or changes to the contents of this MIB
                   module require either publication of an RFC, or
                   Designated Expert Review as defined in RFC 2434,
                   Guidelines for Writing an IANA Considerations Section
                   in RFCs.  The Designated Expert will be selected by
                   the IESG Area Director(s) of the Applications Area.

                   Copyright (C) The Internet Society (2004). The
```

```
                  initial version of this MIB module was published
                  in RFC 3806. For full legal notices see the RFC
                  itself or see:
                  http://www.ietf.org/copyrights/ianamib.html"

    REVISION      "200406020000Z"  -- June 2, 2004
    DESCRIPTION   "Original version, published in coordination
                  with Finisher MIB (RFC 3806)."

    ::= { mib-2 110 }

-- TEXTUAL-CONVENTIONs for this MIB module

FinDeviceTypeTC ::= TEXTUAL-CONVENTION
    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),
         inserter(18)
        }

FinAttributeTypeTC  ::= TEXTUAL-CONVENTION
    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),
          foldingType(40),
          bindingType(50),
          punchHoleType(80),
          punchHoleSizeLongDim(81),
          punchHoleSizeShortDim(82),
          punchPattern(83),
          slittingType(100),
          wrappingType(130),
          stackOutputType(160),
          stackOffset(161),
          stackRotation(162)
         }

FinEdgeTC ::= TEXTUAL-CONVENTION
    STATUS       current
    DESCRIPTION
      "Specifies an edge for a Finishing Process."
    SYNTAX       INTEGER {
          topEdge(3),
          bottomEdge(4),
          leftEdge(5),
          rightEdge(6)
         }

FinStitchingTypeTC ::= TEXTUAL-CONVENTION
    STATUS       current
    DESCRIPTION
      "The defined stitching type enumerations.  For the edgeStitch and
       stapleDual enums, the finReferenceEdge attribute is recommended
```

```
       to define the edge to which the operation applies."
    SYNTAX       INTEGER {
         other(1),         -- More information in other attributes
         unknown(2),
         stapleTopLeft(4),
         stapleBottomLeft(5),
         stapleTopRight(6),
         stapleBottomRight(7),
         saddleStitch(8),
         edgeStitch(9),
         stapleDual(10)
        }

FinStitchingDirTypeTC ::= TEXTUAL-CONVENTION
    STATUS       current
    DESCRIPTION
      "Defines the direction, relative to the top sheet in the output
       subunit, that the stitching operation was performed.  For a
       topDown(3) process, the staple will be clinched on the bottom
       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)
        }

FinStitchingAngleTypeTC ::= TEXTUAL-CONVENTION
    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
   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
    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
   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
   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
    STATUS       current
    DESCRIPTION
      "The defined slitting type enumerations."
    SYNTAX       INTEGER {
          other(1),         -- More information in other attributes
          unknown(2),
          slitAndSeparate(4),
          slitAndMerge(5)
         }

FinWrappingTypeTC ::= TEXTUAL-CONVENTION
   STATUS       current
   DESCRIPTION
     "The defined wrapping device process enumerations."
   SYNTAX       INTEGER {
          other(1),        -- More information in other attributes
          unknown(2),
          shrinkWrap(4),
          paperWrap(5)
         }

FinStackOutputTypeTC ::= TEXTUAL-CONVENTION
   STATUS       current
   DESCRIPTION
     "The defined stack output type enumerations."
   SYNTAX       INTEGER {
          other(1),        -- More information in other attributes
          unknown(2),
          straight(4),     -- No offset, one on top of another
          offset(5),
          crissCross(6)    -- Rotated
```

```
        }
END
```

8. Printer Finishing MIB Specification

```
Finisher-MIB DEFINITIONS ::= BEGIN

IMPORTS
    MODULE-IDENTITY, OBJECT-TYPE, Integer32, mib-2
        FROM SNMPv2-SMI                                 -- [RFC2578]
    MODULE-COMPLIANCE, OBJECT-GROUP
        FROM SNMPv2-CONF                                -- [RFC2580]
    hrDeviceIndex
        FROM HOST-RESOURCES-MIB                         -- [RFC2790]
    PrtInputTypeTC, PrtMarkerSuppliesTypeTC
        FROM IANA-PRINTER-MIB                           -- [RFC3805]
    printmib, PrtSubUnitStatusTC, PrtLocalizedDescriptionStringTC,
        PrtMarkerSuppliesSupplyUnitTC, PrtMediaUnitTC,
        PrtCapacityUnitTC, PrtMarkerSuppliesClassTC,
        PresentOnOff, prtMIBConformance
        FROM Printer-MIB                                -- [RFC3805]
    FinDeviceTypeTC, FinAttributeTypeTC
        FROM IANA-FINISHER-MIB;

finisherMIB MODULE-IDENTITY
    LAST-UPDATED "200406020000Z"
    ORGANIZATION "PWG IEEE/ISTO Printer Working Group"
    CONTACT-INFO

        "Harry Lewis
        IBM
        Phone (303) 924-5337
        Email: harryl@us.ibm.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 'Finisher
        MIB':      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 printers.
         Copyright (C) The Internet Society (2004). This
         version of this MIB module was published
```

```
         in RFC 3806. For full legal notices see the RFC itself."
    REVISION "200406020000Z"
    DESCRIPTION
        "The original version of this MIB."
    ::= { mib-2  111 }

 -- Finisher Device Group (Mandatory)
--
-- A printer may support zero or more finishing subunits.  A
-- finishing device subunit may be associated with one or more
-- output subunits and one or more media path subunits.

finDevice  OBJECT IDENTIFIER  ::=  { printmib 30 }

finDeviceTable OBJECT-TYPE
    SYNTAX     SEQUENCE OF FinDeviceEntry
    MAX-ACCESS not-accessible
    STATUS     current
    DESCRIPTION
      "This table defines the finishing device subunits,
       including information regarding possible configuration
       options and the status for each finisher device subunit."
    ::= { finDevice 1 }

finDeviceEntry OBJECT-TYPE
    SYNTAX      FinDeviceEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
      "There is an entry in the finishing device table for each
       possible finisher process.  Each individual finisher process is
       implemented by a finishing device represented in this table."
    INDEX  { hrDeviceIndex, finDeviceIndex }
    ::= { finDeviceTable 1 }

FinDeviceEntry ::= SEQUENCE {
      finDeviceIndex                Integer32,
      finDeviceType                 FinDeviceTypeTC,
      finDevicePresentOnOff         PresentOnOff,
      finDeviceCapacityUnit         PrtCapacityUnitTC,
      finDeviceMaxCapacity          Integer32,
      finDeviceCurrentCapacity      Integer32,
      finDeviceAssociatedMediaPaths OCTET STRING,
      finDeviceAssociatedOutputs    OCTET STRING,
      finDeviceStatus               PrtSubUnitStatusTC,
      finDeviceDescription          PrtLocalizedDescriptionStringTC
     }
```

```
finDeviceIndex OBJECT-TYPE
    SYNTAX      Integer32 (1..65535)
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
      "A unique value used to identify a finisher process.
       Although these values may change due to a major
       reconfiguration of the printer system (e.g., the addition
       of new finishing processes), the values are normally
       expected to remain stable across successive power cycles."
    ::= { finDeviceEntry 1 }

finDeviceType OBJECT-TYPE
    SYNTAX      FinDeviceTypeTC
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
      "Defines the type of finishing process associated with this
       table row entry."
    ::= { finDeviceEntry 2 }

finDevicePresentOnOff OBJECT-TYPE
    SYNTAX      PresentOnOff
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
      "Indicates if this finishing device subunit is available
       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 (SIZE(1..63))
    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 (SIZE(1..63))
    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 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      PrtLocalizedDescriptionStringTC
    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         PrtLocalizedDescriptionStringTC,
      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."
    ::= { finSupplyEntry 1 }

finSupplyDeviceIndex 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 supply is associated.
       The value zero indicates the associated finishing device is
       Unknown."
    ::= { finSupplyEntry 2 }
```

```
finSupplyClass OBJECT-TYPE
    SYNTAX      PrtMarkerSuppliesClassTC
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
      "This value indicates whether this supply entity
       represents a supply that is consumed or a container that
       is filled."
    ::= { finSupplyEntry 3 }

finSupplyType OBJECT-TYPE
    SYNTAX      PrtMarkerSuppliesTypeTC
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
      "The type of this supply."
    ::= { finSupplyEntry 4 }

finSupplyDescription OBJECT-TYPE
    SYNTAX      PrtLocalizedDescriptionStringTC
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
      "The description of this supply/receptacle in text useful
       for operators and management applications and in the
       localization specified by prtGeneralCurrentLocalization."
    ::= { finSupplyEntry 5 }

finSupplyUnit OBJECT-TYPE
    SYNTAX      PrtMarkerSuppliesSupplyUnitTC
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
      "Unit of measure of this finisher supply container or
       receptacle."
    ::= { finSupplyEntry 6 }

finSupplyMaxCapacity OBJECT-TYPE
    SYNTAX      Integer32 (-2..2147483647)
    MAX-ACCESS  read-write
    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 (-3..2147483647)
    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 associated with this supply."

    REFERENCE
       "The PWG Standardized Media Names specification [PWGMEDIA],
        section 4 Media Color Names, contains the recommended values
```

```
        for this object. Implementers 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    PrtLocalizedDescriptionStringTC,
      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.  The value zero indicates the associated device
       is unknown."
    ::= { 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.  The value zero indicates the associated finishing
       supply is unknown or there is no applicable finisher supply
       table entry."
    ::= { 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      PrtLocalizedDescriptionStringTC
    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
       unknown.  This object can be used to calculate the weight
       of individual pages processed by the document finisher.
       This value, when multiplied by the number of pages in a
       finished set, can be used to calculate the weight of a set
       before it is inserted into a mailing envelope."
    ::= { finSupplyMediaInputEntry 13 }

finSupplyMediaInputMediaThickness OBJECT-TYPE
    SYNTAX      Integer32 (-2..2147483647)
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
      "This object identifies the thickness of the input media
       processed by this document input subunit measured in
       micrometers.  This value may be used by devices (or
       operators) to set up proper machine tolerances for the
       feeder operation.  The value (-2) indicates that the media
       thickness is unknown or not used in the setup for this
       input subunit."
    ::= { finSupplyMediaInputEntry 14 }

finSupplyMediaInputMediaType OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE(0..63))
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
      "The name of the type of medium associated with this input
       subunit. "

    REFERENCE
       "The PWG Standardized Media Names specification [PWGMEDIA],
        section 3 Media Type Names, contains the recommended values
        for this object. Implementers may add additional string values.
        The naming conventions in ISO 9070 are recommended in order to
        avoid potential name clashes."
    ::= { finSupplyMediaInputEntry 15 }
```

```
-- Finisher Device Attribute Group (Mandatory)
--
-- A finisher device subunit may have one or more parameters that
-- cannot be specified by any other objects in the MIB.  The
-- Device Attribute group facilitates the definition of these
-- parameters.  The objects which define the attributes are
-- read-write, to allow both Set and Get operations.
--
-- At least one table entry must exist for each finisher device defined
-- by the MIB.  If no other entry is possible for a finisher device, the
-- deviceName(3) attribute MUST be returned.

finDeviceAttribute  OBJECT IDENTIFIER  ::=  { printmib 33 }

finDeviceAttributeTable OBJECT-TYPE
    SYNTAX     SEQUENCE OF FinDeviceAttributeEntry
    MAX-ACCESS not-accessible
    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.
       Additional values shall be assigned in a contiguous manner."
    ::= { finDeviceAttributeEntry 2 }

finDeviceAttributeValueAsInteger OBJECT-TYPE
    SYNTAX      Integer32 (-2..2147483647)
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
      "Defines the integer value of the attribute.  The value of
       the attribute is represented as an integer if the
       finAttributeTypeTC description for the attribute has the
       tag 'INTEGER:'.

       Depending upon the attribute enum definition, this object
       may be either an integer, a counter, an index, or an enum.
       Attributes for which the concept of an integer value is
       not meaningful SHALL return a value of -1 for this
       attribute."
    DEFVAL      { -2 }      -- unknown
    ::= { finDeviceAttributeEntry 3 }

finDeviceAttributeValueAsOctets OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE(0..63))
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
      "Contains the octet string value of the attribute.  The
       value of the attribute is represented as a string if the
       finAttributeTypeTC description for the attribute has the
       tag 'OCTETS:'.
```

```
       Depending upon the attribute enum definition, this object
       may be either a coded character set string (text) or a
       binary octet string.  Attributes for which the concept of
       an octet string value is not meaningful SHALL contain a
       zero length string."
    DEFVAL      { ''H }      -- empty string
    ::= { finDeviceAttributeEntry 4 }


-- Conformance Information

-- compliance statements

finMIBCompliance MODULE-COMPLIANCE
    STATUS  current
    DESCRIPTION
        "The compliance statement for agents that implement the
        finisher MIB."
    MODULE -- this module
    MANDATORY-GROUPS { finDeviceGroup, finSupplyGroup,
                       finDeviceAttributeGroup }

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

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

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

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

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

        OBJECT      finSupplyMediaInputMediaDimFeedDir
```

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

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

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

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

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

        OBJECT      finSupplyMediaInputMediaWeight
        MIN-ACCESS  read-only
        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 5 }


finMIBGroups      OBJECT IDENTIFIER ::= { prtMIBConformance 6 }

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.  This group is mandatory
         for a finisher device that contains an inserter subunit."
    ::= { finMIBGroups 4 }

END
```

9. IANA Considerations

The initial version of the IANA Finisher MIB defined in section 7 of this document is to be archived by IANA and subsequently maintained according to the Process specified in section 6.1 of this document. The most current and authoritative version of the IANA Finisher MIB is available at:

http://www.iana.org/assignments/ianafinisher-mib

10. Internationalization Considerations

See the Printer MIB [RFC3805] section 2.2.1.1, 'International Considerations'.

11. References

11.1. Normative References

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

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

[PWGMEDIA] IEEE-ISTO PWG "The Printer Working Group Standard for Media Standardized Names", IEEE-ISTO PWG 5101.1-2002.

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

[RFC2434] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 2434, October 1998.

[RFC2578] McCloghrie, K., Perkins, D., and J. Schoenwaelder, "Structure of Management Information Version 2 (SMIv2)", STD 58, RFC 2578, April 1999.

[RFC2579] McCloghrie, K., Perkins, D., and J. Schoenwaelder, "Textual Conventions for SMIv2", STD 58, RFC 2579, April 1999.

[RFC2580] McCloghrie, K., Perkins, D., and J. Schoenwaelder, "Conformance Statements for SMIv2", STD 58, RFC 2580, April 1999.

[RFC2790] Waldbusser, S. and P. Grillo, "Host Resources MIB", RFC 2790, March 2000.

[RFC3805] Bergman, R., Lewis, H., and I. McDonald, "The Printer MIB v2", RFC 3805, June 2004.

11.2. Informative References

[RFC2911] Hastings, T. Ed., Herriot, R., deBry, R., Issacson, S., and P. Powell, "Internet Printing Protocol/1.1: Model and Semantics", RFC 2911, September 2000.

[RFC3410] Case, J., Mundy, R., Partain, D., and B. Stewart, "Introduction and Applicability Statements for Internet-Standard Management Framework", RFC 3410, December 2002.

12. Security Considerations

There are a number of management objects defined in this MIB module with a MAX-ACCESS clause of read-write and/or read-create. Such objects may be considered sensitive or vulnerable in some network environments. The support for SET operations in a non-secure environment without proper protection can have a negative effect on network operations. These are the tables and objects and their sensitivity/vulnerability:

```
finDeviceTable:
   finDevicePresentOnOff -Possible severe inconvenience
   finDeviceMaxCapacity -Possible minor inconvenience
   finDeviceCurrentCapacity -Possible minor inconvenience
finSupplyTable:
   finSupplyMaxCapacity -Possible minor inconvenience
```

finSupplyCurrentLevel -Possible minor inconvenience
finSupplyMediaInputTable
finSupplyMediaInputMediaDimFeedDir -Possible severe inconvenience
finSupplyMediaInputMediaDimXFeedDir -Possible severe inconvenience
finSupplyMediaInputMediaName -Possible Minor inconvenience
finSupplyMediaInputName -Possible Minor inconvenience
finSupplyMediaInputSecurity -Possible Minor inconvenience
finSupplyMediaInputMediaWeight -Possible Minor inconvenience
finSupplyMediaInputMediaThickness -Possible Minor inconvenience
finSupplyMediaInputMediaType -Possible Minor inconvenience
finDeviceAttributeTable
finDeviceAttributeValueAsInteger -Possible Minor inconvenience
finDeviceAttributeValueAsOctets -Possible Minor inconvenience

SNMP versions prior to SNMPv3 did not include adequate security. Even if the network itself is secure (for example by using IPSec), even then, there is no control as to who on the secure network is allowed to access and GET/SET (read/change/create/delete) the objects in this MIB module.

It is RECOMMENDED that implementers consider the security features as provided by the SNMPv3 framework (see [RFC3410], section 8), including full support for the SNMPv3 cryptographic mechanisms (for authentication and privacy).

Further, deployment of SNMP versions prior to SNMPv3 is NOT RECOMMENDED. Instead, it is RECOMMENDED to deploy SNMPv3 and to enable cryptographic security. It is then a customer/operator responsibility to ensure that the SNMP entity giving access to an instance of this MIB module is properly configured to give access to the objects only to those principals (users) that have legitimate rights to indeed GET or SET (change/create/delete) them.

Where the operational capability of the printing device are especially vulnerable or difficult to administer, certain objects within this MIB have been tagged as READ-ONLY, preventing modification. Further, for all READ-WRITE objects within the MIB, the working group has included specific conformance guidelines stating that vendors are free to implement certain objects as READ-ONLY. This conformance allowance should cover cases where specific vendor vulnerabilities may differ from product to product. (See conformance section with regards to MIN-ACCESS clauses).

13. Acknowledgements

The Printer MIB Working Group would like to extend a special thank you to the following individuals that put forth a significant effort to review this document and provide numerous suggestions for improvement.

David Harrington - Enterasys Networks
Juergen Schoenwaelder - TU Braunschweig
Bert Wijnen - Lucent Technologies and IETF Op & Mngmt, Area Director

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

14. Authors' Addresses

Ron Bergman (Editor)
Hitachi Printing Solutions America
2635 Park Center Drive
Simi Valley, CA 93065-6209

Phone: 805-578-4421
Fax: 805-578-4001
EMail: Ron.Bergman@hitachi-ps.us

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

Phone: (303) 924-5337
EMail: harryl@us.ibm.com

Ira McDonald
High North Inc.
P.O. Box 221
Grand Marais, MI 49839

Phone: (906) 494-2434 or (906) 494-2697
EMail: imcdonald@sharplabs.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/

15. Full Copyright Statement

Intellectual Property

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.