Printer Management Interface

Management Object Specification

Version 0.7, April 27, 1994

By

Ronald L. Smith

Texas Instruments

email: rlsmith@nb.ppd.ti.com
phone: (817) 774-6151

F.D. Wright

Lexmark International

email: don@lexmark.com

phone: 606-232-4808

Tom Hastings

Digital Equipment Corp.

email: hastings@hannah.enet.dec.com

phone: (508) 467-8299

This document specifies the semantics of the IETF Printer MIB and the DMTF Printer MIFin a manner that is independent of the syntax for MIB and MIF specifications. After this document is relatively stable, it will be converted into the appropriate MIB and MIF format specifications.

Introduction

The Printer Management Interface (PMI) Object Specification specifies the semantics of the IETF Printer MIB and the DMTF Printer MIF in a manner that is independent of the syntax for MIB and MIF specifications.

<Model Description and Introduction goes here. But I couldn't read Steve's pmi_mod3 file.>

<Picture of printers, print systems, and Network Management Stations (NMSs) go here.>

Enumerations

This working group 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 "enum(n)", where n can be 1, 2 or 3 can be found in the various tables. The definitions of these types of enumerations are:

ENUM(1) Additional enumerated values require a new RFC.

ENUM(2) Additional enumerated values are registered after review by this working group

ENUM(3) Additional enumerated values are registered without working group review.

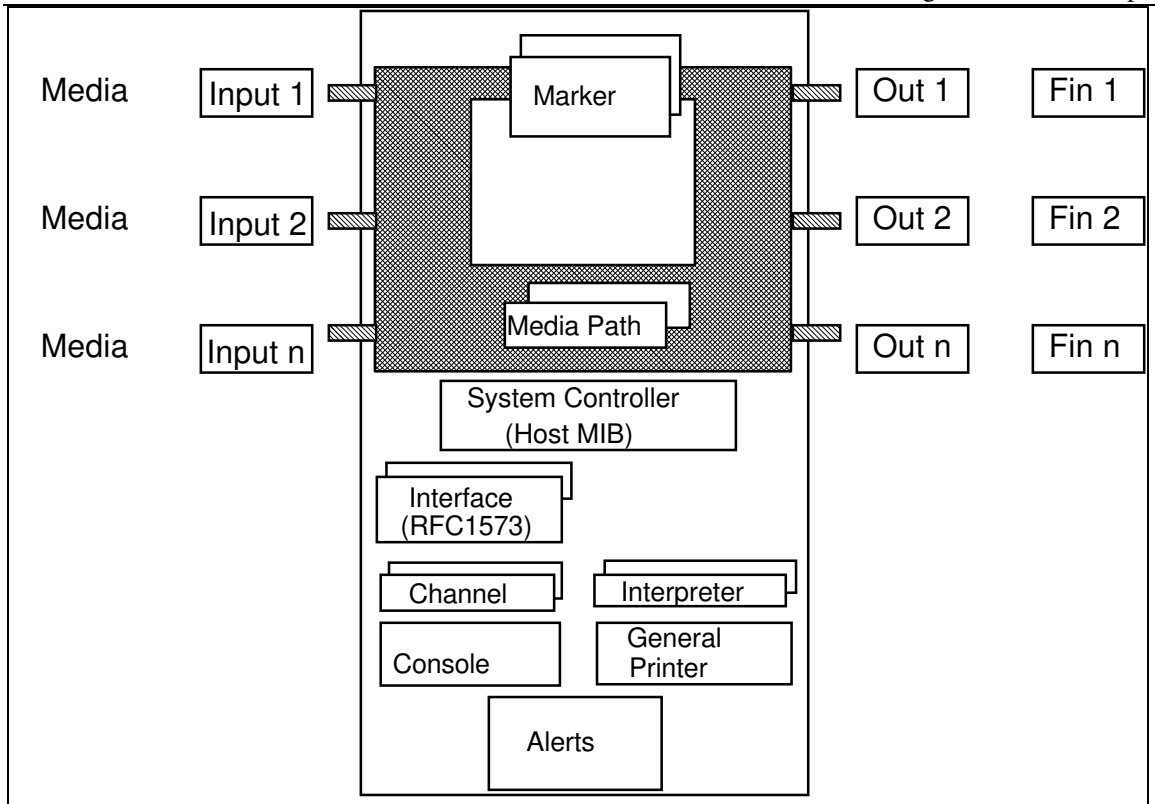
Each enumeration is given a symbolic name and the values are listed at the end of this specification.

In IETF MIBs, the objects defined by this specification will have a prefix of **prt** and the name of the group (**Input**, **Output, Interpreter**, etc. In DMTF MIBs, the same attributes will have ??? prefixes. However, to save space in this specification and to be neutral between the IETF and the DMTF conventions, the **prt** and the group name prefix will not generally be given. The prefix is given at the top of each MS-WORD table.

When a description of an object/attribute refers to another object/attribute, the name of that other object/attribute is given in **boldface**. Such references to include neither the **prt** prefix nor the group name prefix, if the object/attribute is in the same group. In the rare cases that a reference is made to an object in another group, both the **prt** prefix and the group name are included in the reference.

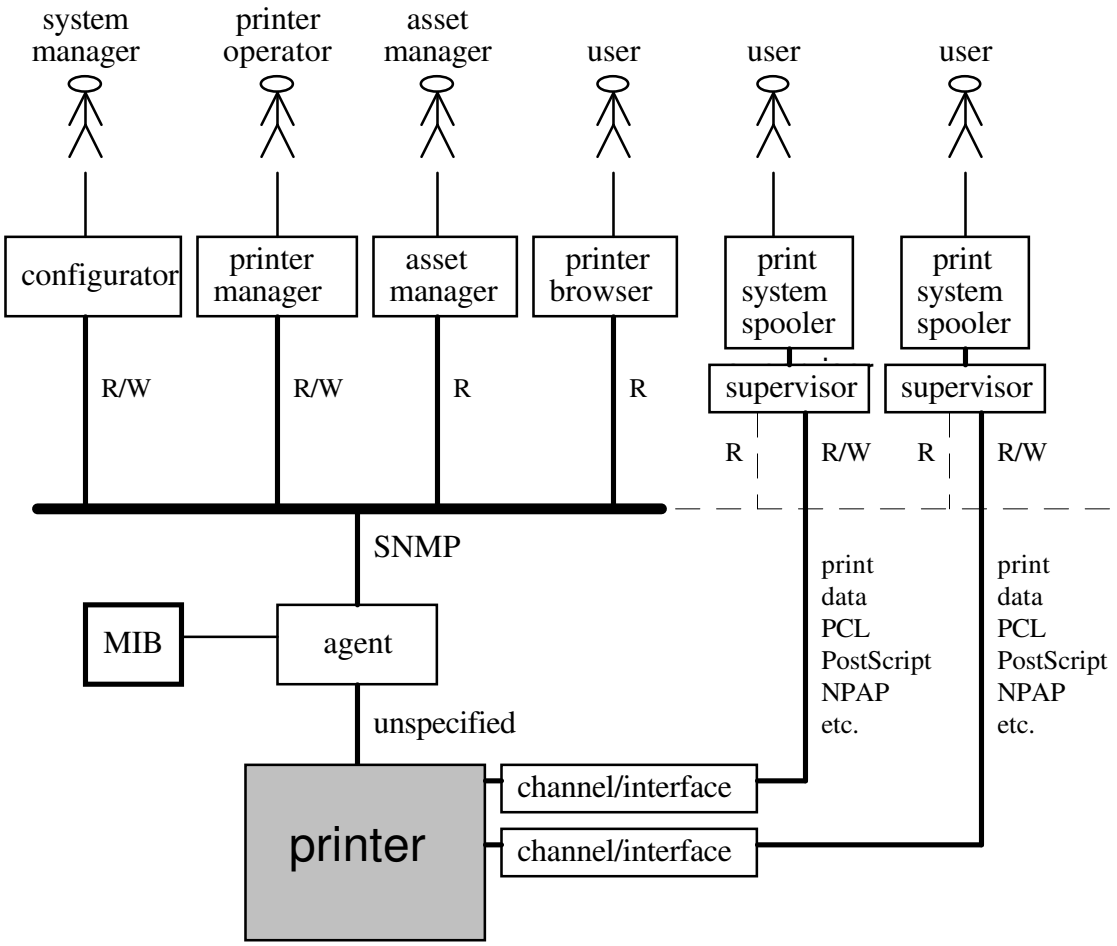
Printer Model

This working group has developed a model of a printer which is useful in understanding the classification of the various object. Graphically this model is as shown below:



Printer Groups of the Printer Model (Updated)

One Printer's View of the Network



Printer Component

To be supplied by DMTF to correspond to MIB II

DMTF MIF "Component ID: (From DMI Specification - Draft 4.2 - 12 January 1994)

1.	Manufacturer	String64	Name of the organization that produces this component.
2.	Product	String64	Name of this Printer
3.	Version	String64	
4.	Serial Number	String64	
5.	Installation	Date	Time and Date of the Last Installation
6.	Verify	Integer	Verification Level for this printer

"Asking for the 'Verify' attribute causes the component instrumentation code to perform checks to verify that the component is still in the system and working properly. It should return on of the following values:

Value	Meaning
0	An error occurred; Check status code
1	Component does not exist
2	Verify not supported
4	Component exists, functionally untested
5	Component exists, functionally unknown
6	Component exists, functionally no good
7	Component exists, functionally good

System Group objects

This is place holder to list the System Group objects from MIB-II RFC 1213 that shall be implemented when implementing the Printer MIF/MIF.....

System Controller

This is a place holder to list the Host Resources objects from RFC 1514 that shall be implemented when implementing the Printer MIB/MIF. Possible groups from which we may choose objects include: System, Storage, Device, Running Software, Running Software Performance, and Installed Software.....

Interface Group objects

This is a place holder to list the Interface Group objects from RFC 1573 that shall be implemented when implementing the Printer MIB/MIF.....

General Printer Group Objects

prtGeneral	Mandatory			
Object Name	**Access**	**Type**	**Value**	**Description**
ConfigChange	Read-Only	Counter32		Counts configuration changes.
DefaultOrientation	Read-Write	Enum (1)	MediaOrientation	The current orientation default. This value applies to all input, all interpreters, etc. unless overridden by other control.
CurrentLocalization	Read-Write	Integer32		Index into the localization table specifying the current language, country and character set for **Description** string values.

Notes:

1. Overall printer status can be determined using the following objects from the host MIB: (hrDevice Status, hrPrinterStatus, and hrPrinterDetectedErrorState.

prtGeneral	Mandatory		prtSkinTable	
Object Name	**Access**	**Type**	**Value**	**Description**
ID	Read-Only	Integer32		A unique value used by the printer to identify this skin sub-unit. Although these values may change due to a major reconfiguration of the device (e.g. the addition of new skin components to the printer), values are expected to remain stable across successive printer power cycles.
Description	Read-Only	String		The manufacturer provided skin component name in the localization specified by **CurrentLocalization**. This object should be in localization specified by CurrentLocalization.
Status	Read-Only	Enum(2)	SkinStatus	The status of this skin sub-unit.

Notes:

prtGeneral	Mandatory		prtLocalizationTable	
Object Name	**Access**	**Type**	**Value**	**Description**
Language	Read-Only	String(2)		A two character language code from ISO 639. Examples: EN, FR, DE, ...
Country	Read-Only	String(2)		A two character country code from ISO 3166, a blank string (two space characters) shall indicate that the country is not defined. Examples: US, GB, CA, FR, DE, ...
CharacterSet	Read-Only	Integer32		An index into the character set table identifying the character set used for this localization.

Notes:

prtGeneral	Mandatory		prtCharacterSetTable	
Object Name	**Access**	**Type**	**Value**	**Description**
CharacterSet	Read-Only	String		A character set name or identifier that identifies this character set.

Notes:

prtGeneral - Extended	Optional			
Object Name	**Access**	**Type**	**Value**	**Description**
Reset	Write-Only	Enum(3)	ResetType	Writing causes a reset

Notes:

Input Group Objects - Mandatory

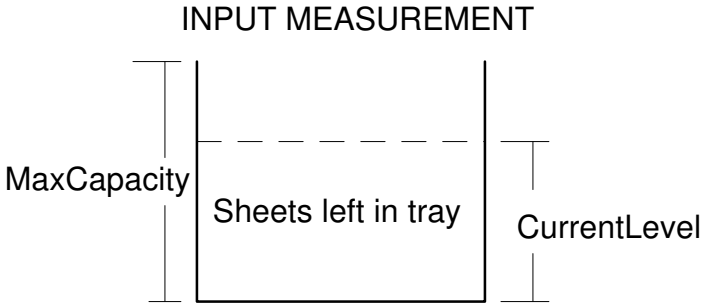
Input sub-units are managed as a tabular, indexed collection of possible devices capable of providing media for input to the printing process. Input sub-units typically have a location, a type, an identifier, a set of constraints on possible media sizes and potentially other media characteristics, and may be capable of indicating current status or capacity.

prtInput	Mandatory			
Object Name	**Access**	**Type**	**Value**	**Description**
DefaultID	Read-Write	Integer32		The ID of the default input device, i.e., the value of the corresponding input device's ID value.

Notes:

prtInput	Mandatory	prtInputTable		
Object Name	**Access**	**Type**	**Value**	**Description**
ID	Read-Only	Integer32		A unique value used by the printer to identify this input device. Although these values may change due to a major reconfiguration of the device (e.g. the addition of new input devices to the printer), values are expected to remain stable across successive printer power cycles.
Type	Read-Only	Enum (2)	InputType	The type of technology (discriminated primarily according to feeder mechanism type) employed by the input device. Note, the Optional Input Class provides for a descriptor field to further qualify the other choice.
DimUnit	Read-Write	Enum (1)	PhyMediaUnit	The unit of measurement for use in calculating and relaying dimensional values for this input device.
MediaDimAcrossFeedDir	Read-Write	Integer32		This object provides the value of the current dimensions supported by the input device for measurements taken ninety degrees relative to the feed direction associated with that device in input device dimensional units (**DimUnit**). The value (-1) means other and specifically indicates that the device places no restriction on this parameter. The value (-2) indicates unknown.
MediaDimFeedDir	Read-Write	Integer32		The current dimensions supported by the input device for measurements taken parallel relative to the feed direction associated with that device in input device dimensional units (**DimUnit**). The value (-1) means other and specifically indicates that the device places no restriction on this parameter. The value (-2) indicates unknown.
CapacityUnit	Read-Write	Enum (1)	CapacityUnit	The unit of measurement for use in calculating and relaying capacity values for this input device.
MaxCapacity	Read-Write	Integer32		The maximum capacity of the input sub-unit in input device capacity units (**CapacityUnit**). There is no convention associated with the media itself so this value reflects claimed capacity. This field is read-write in order to allow a management application to request the desired capacity input device. The value (-1) means other and specifically indicates that the device places no restrictions on this parameter. The value (-2) means unknown.
CurrentLevel	Read-Write	Integer32		The current capacity of the input subunit in input device capacity units (**CapacityUnit**). This field is read-write to accommodate remote op-panel implementation. The value (-1) means other and specifically indicates that the device places no restrictions on this parameter. The value (-2) means unknown.
Status	Read-Only	Enum	EntStatus	The current status of this input device.
MediaName	Read-Write	String		The name of the current medium contained in this input sub-uni. Examples: "iso-a4-white", "letter-white", "Hillary's Pink Presidential Letterhead"

Notes:


INPUT MEASUREMENT
MaxCapacity
Sheets left in tray
CurrentLevel

Input Group Objects - Optional

prtInput - TypeExtended	Optional		prtInputTableAugment	
Object Name	**Access**	**Type**	**Value**	**Description**
Name	Read-Write	String		The name assigned to this input device.
VendorName	Read-Only	String		The vendor name of this input device.
Model	Read-Only	String		The model name of this input device.
Version	Read-Only	String		The version of this input device.
SerialNumber	Read-Only	String		The serial number assigned to this input device.
Description	Read-Only	String		A free-form text description of this input device in the localization specified by **CurrentLocalization**.
Security	Read-Write	Enum (2)	YesNo-OnOff	Indicates if this input device has some security associated with it.

Notes:

Media Extended Group - OPTIONAL

The Media Extended Group supports identification of media installed or available for use on a printing device. Medium resources are identified by name, and include a collection of characteristic attributes that may further be used for selection and management of them. The Media Extended group is conceptually built in the form of an SNMP "AUGMENTS" construction off of the InputDevice table. In this manner, a minimally conforming implementation may choose to not support reporting of media resources if it cannot do so.

prtInput - MediaExtended	Optional		prtInputTableAugment	
Object Name	**Access**	**Type**	**Value**	**Description**
MediaWeight	Read-Write	Integer32	grams/meter^2	The weight of the medium in grams / per meter squared associated with this input device. The value (-1) means other and specifically indicates that the device places no restrictions on this parameter. The value (-2) means unknown.
MediaType	Read-Write	String		The type of medium associated with this input device. See Appendix for standardized names from ISO 10175 (DPA) & ISO 10180 (SPDL)
MediaColor	Read-Write	String		The color of the medium associated with this input device. See Appendix for standardized names from ISO 10175 (DPA) & ISO 10180 (SPDL)
MediaFormParts	Read-Write	Integer32		The number of parts associated with the medium associated with this input device if the medium is a multi-part form. The value (-1) means other and specifically indicates that the device places no restrictions on this parameter. The value (-2) means unknown.

Notes:

Output Group Objects - Mandatory

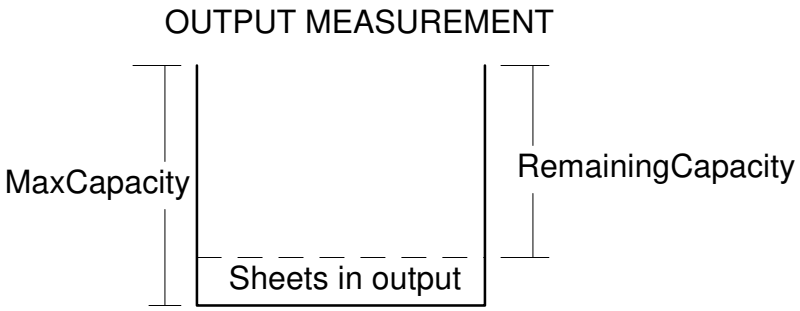
Output sub-units are managed as a tabular, indexed collection of possible devices capable of receiving media delivered from the printing process. Output sub-units typically have a location, a type, an identifier, a set of constraints on possible media sizes and potentially other characteristics, and may be capable of indicating current status or capacity.

prtOutput	Mandatory			
Object Name	**Access**	**Type**	**Value**	**Description**
DefaultID	Read-Write	Integer32		The ID of the default output device, i.e., the value of the corresponding output device's **ID** value.

Notes:

prtOutput	Mandatory		prtOutputTable	
Object Name	**Access**	**Type**	**Value**	**Description**
ID	Read-Only	Integer32		A unique value used by this printer to identify this output device. Although these values may change due to a major reconfiguration of the device (e.g. the addition of new output devices to the printer), values are expected to remain stable across successive printer power cycles.
Type	Read-Only	Enum (2)	OutputType	The type of technology supported by this output device.
CapacityUnit	Read-Only	Enum (1)	CapacityUnit	The unit of measurement for use in calculating and relaying capacity values for this output device.
MaxCapacity	Read-Write	Integer32		The maximum capacity of this output sub-unit in output device capacity units (**CapacityUnit**). There is no convention associated with the media itself so this value essentially reflects claimed capacity. This field is read-write in order to allow a management application to set the desired capacity of the output device. The value (-1) means other and specifically indicates that the device places no restrictions on this parameter. The value (-2) means unknown.
RemainingCapacity	Read-Write	Integer32		The remaining capacity of the possible output sub-unit capacity in output device capacity units (**CapacityUnit**) of this output device..
Status	Read-Only	Enum	EntStatus	The current status of this output device.

Notes:



Output Group Objects - Optional

prtOutput - TypeExtended	Optional		prtOutputTableAugment	
Object Name	**Access**	**Type**	**Value**	**Description**
Name	Read-Write	String		The name assigned to this output device.
VendorName	Read-Only	String		The vendor name of this output device.
Model	Read-Only	String		The name assigned to this output device.
Version	Read-Only	String		The version of this output device.
SerialNumber	Read-Only	String		The serial number assigned to this output device.
Description	Read-Only	String		A free-form text description of this output device in the localization specified by **CurrentLocalization**.
Security	Read-Write	Enum (2)	YesNo-OnOff	Indicates if this output device has some security associated with it.

Notes:

prtOutput - DimExtended	Optional		prtOutputTableAugment	
Object Name	**Access**	**Type**	**Value**	**Description**
DimUnit	Read-Only	Enum (1)	PhyMediaUnit	The unit of measurement for use in calculating and relaying dimensional values for this output device.
MaxDimAcrossFeedDir	Read-Write	Integer32		The maximum dimensions supported by this output device for measurements taken ninety degrees relative to the feed direction associated with that device in output device dimensional units (**DimUnit**).
MaxDimFeedDir	Read-Write	Integer32		The maximum dimensions supported by this output device for measurements taken parallel relative to the feed direction associated with that device in output device dimensional units (**DimUnit**).
MinDimAcrossFeedDir	Read-Write	Integer32		The minimum dimensions supported by this output device for measurements taken ninety degrees relative to the feed direction associated with that device in output device dimensional units (**DimUnit**).
MinDimFeedDir	Read-Write	Integer32		The minimum dimensions supported by this output device for measurements taken parallel relative to the feed direction associated with that device in output device dimensional units (**DimUnit**).

Notes:

prtOutput - Features	Optional		prtOutputTableAugment	
Object Name	**Access**	**Type**	**Value**	**Description**
StackingOrder	Read-Write	Enum (1)	StackingOrder	The current state of the stacking order for the associated output device. “1 to n” means pages are output in the order in which they are acquired. “n to 1” means pages are output from last to first which implies that all pages must be acquired before they can be output.
PageDeliveryOrientation	Read-Write	Enum (1)	PageDeliveryOrientation	Indicates which reading surface will be "up" when pages are delivered to the associated output device. Values are Face-Up and Face-Down. (Note: interpretation of these values is in general context-dependent based on locale; presentation of these values to an end-user should be normalized to the expectations of the user).
Bursting	Read-Write	Enum (1)	YesNo-OnOff	By its presence, this object indicates that the output device supports bursting, and if so, whether the feature is enabled. Bursting is the process by which continuous media is separated into individual sheets, typically by bursting along pre-formed perforations.
Decollating	Read-Write	Enum (1)	YesNo-OnOff	By its presence, this object indicates that the output device supports decollating, and if so, whether the feature is enabled. Decollating is the process by which the individual parts within a multi-part form are separated and sorted into separate stacks for each part.
PageCollated	Read-Write	Enum (1)	YesNo-OnOff	By its presence, this object indicates that the output device supports page collation, and if so, whether the feature is enabled.
Job Collated	Read-Write	Enum (1)	YesNo-OnOff	By its presence, this object indicates that the output device supports job collation, and if so, whether the feature is enabled.
OffsetStacking	Read-Write	Enum (1)	YesNo-OnOff	By its presence, this object indicates that the output device supports offset stacking, and if so, whether the feature is enabled.
Slipsheet	Read-Write	Enum (1)	YesNo-OnOff	By its presence, this object indicates that the output device supports slipsheet insertion between sets, and if so, whether the feature is enabled.

Notes:

Output Device Associated Finishing

Finisher devices are typically complicated systems and can (or should be) dealt with largely isolated from the "output device." Since there are so many configuration options for finisher devices (and some of them may indeed be integrated within the output device itself), the question is whether to deal with finishers as part of output device management, or whether to deal with them separately. The merit of separating the two "types" of devices is that it enables greater modularity across the widest variety of interesting configurations, and that is the approach taken in this specification.

A common model for basic management of finisher devices is provided later in this proposal in the Finisher Class section.

prtOutput - Associated Finisher	**Optional**		**prtOutputAssocFinisher**	
Object Name	**Access**	**Type**	**Value**	**Description**
ID	Read-Only	Integer32		The ID of the output device to which the finisher is connected.
Stitching	Read-Only	Enum(1)	YesNo-OnOff	By its presence, this object indicates that the output device supports stapling as a process accessible in the production path associated with it. The value returned by this object is the identifier for the associated finishing device; the value serves as the index for identifying the device in the **prtFinisherTable**.
Folding	Read-Only	Enum(1)	YesNo-OnOff	By its presence, this object indicates that the output device supports folding as a process accessible in the production path associated with it. The value returned by this object is the identifier for the associated finishing device; the value serves as the index for identifying the device in the **prtFinisherTable**.
Binding	Read-Only	Enum(1)	YesNo-OnOff	By its presence, this object indicates that the output device supports binding as a process accessible in the production path associated with it. The value returned by this object is the identifier for the associated finishing device; the value serves as the index for identifying the device in the **prtFinisherTable**.
Punching	Read-Only	Enum(1)	YesNo-OnOff	By its presence, this object indicates that the output device supports punching as a process accessible in the production path associated with it. The value returned by this object is the identifier for the associated finishing device; the value serves as the index for identifying the device in the **prtFinisherTable**.
Trimming	Read-Only	Enum(1)	YesNo-OnOff	By its presence, this object indicates that the output device supports trimming as a process accessible in the production path associated with it. The value returned by this object is the identifier for the associated finishing device; the value serves as the index for identifying the device in the **prtFinisherTable**.
Insertion	Read-Only	Enum(1)	YesNo-OnOff	By its presence, this object indicates that the output device supports inserting as a process accessible in the production path associated with it. The value returned by this object is the identifier for the associated finishing device; the value serves as the index for identifying the device in the **prtFinisherTable**.

Notes:

Finisher Group Objects - Optional

The Finisher Group is an optional group, and is included in this specification for completeness due to its relationship to the Output Group.

In general, finishers are very complicated devices and their management is a topic suitable for considerable, additional investigation. The goal of this class is therefore not to standardize management of arbitrarily complex finishers. Rather, this class only standardizes basic management objects suitable for a very generic abstract finisher. The value of this, is that by defining this basic class we establish the architectural framework for a common approach which can then be augmented as required in a consistent manner by vendors and consortia.

Finishing Configuration

The finishing table maintains mappings between configurations of possibly zero or more finishers and zero or more output devices. Each output device may provide a value for the optional column "**prtOutputAssociatedFinisher**". The value in that column is an integer which is the "**prtFinisherConfigurationId**". The value for the "**prtFinisherConfigurationId**" is the index into the "**prtFinisherTable**".

prtFinisher	Optional			
Object Name	**Access**	**Type**	**Value**	**Description**
DefaultID	Read-Write	Integer32		The ID of the default finisher device, i.e., the value of the corresponding finisher device's **ID** value. A value of 0 means that a finisher is not used in default configurations.

Notes:

prtFinisher	Optional		prtFinisherTable	
Object Name	**Access**	**Type**	**Value**	**Description**
ID	Read-Only	Integer32		A unique value used by the printer to identify this finisher device. Although these values may change due to a major reconfiguration of the device (e.g. the addition of new finisher devices to the printer), values are expected to remain stable across successive printer power cycles.
Type	Read-Only	Enum(2)	FinisherType	The type of finishing performed by this finisher device
NamedFinishing	Read-Only	Enum(2)	NamedFinishing	The finishing operations available through this finishing device.
ConsumableUnit	Read-Only	Enum(1)	FinisherConsumableUnit	The consumable resource unit appropriate to this finisher device and its operation.
MaxConsumableCapacity	Read-Write	Integer32		The maximum capacity of this finisher in consumable resource units of measure (**ConsumableUnit**). This value is particularly suited for providing a gauge that can reflect the maximum consumable resource capacity of this finisher.
CurrentConsumableCapacity	Read-Write	Integer32		The current capacity of this finisher in consumable resource units of measure (**ConsumableUnit**). This value is particularly suited for providing a gauge that can reflect the current available consumable resource capacity of this finisher.
Status	Read-Only	Enum(1)	EntStatus	The current status of this finisher.

Notes:

Marker Group Objects - Mandatory

prtMarker	Mandatory			
Object Name	**Access**	**Type**	**Value**	**Description**
DefaultID	Read-Write	Integer32		The ID of the default marker device, i.e., the value of the corresponding marker device's **ID** value.

Notes:

prtMarker	Mandatory		prtMarkerTable	
Object Name	**Access**	**Type**	**Value**	**Description**
ID	Read-Only	Integer32		A unique value used by the printer to identify this marking device. Although these values may change due to a major reconfiguration of the device (e.g. the addition of new marking devices to the printer), values are expected to remain stable across successive printer power cycles.
MarkTech	Read-Only	Enum (2)	MarkingTech	The type of marking technology used for this marking device
CounterUnit	Read-Only	Enum (1)	CounterUnit	The unit that will be used by the printer when reporting counter values for this marking device. The time units of measure are provided for a device like a strip recorder that does not or cannot track the physical dimensions of the media and does not use characters, lines or sheets.
LifeCount	Read-Only	Counter32		The count of the number of units of measure counted during the life of printer using units of measure as specified by **CounterUnit**.
PowerOnCount	Read-Only	Counter32		The count of the number of units of measure counted since the equipment was most recently powered on using units of measure as specified by **CounterUnit**.
ProcessColorants	Read-Only	Integer32		The number of process colors supported by this marker. A process color of 1 implies monochrome. The value of this object and **SpotColorants** cannot both be 0. A value of (-1) implies 'other' or 'infinite' while a value of (-2) implies 'unknown'.
SpotColorants	Read-Only	Integer32		The number of spot colors supported by this marker. The value of this object and **ProcessColorants** cannot both be 0. A value of (-1) implies 'other' or 'infinite' while a value of (-2) implies 'unknown'
AddressabilityUnit	Read-Only	Enum (1)	AddrUnit	The unit of measure of the denominator of Horizontal and Vertical addressability.
AddressabilityFeedDir	Read-Only	Integer32		The addressability of the engine in the feed direction in units of measure specified by **AddressabilityUnit**. A value of (-1) implies 'other' or 'infinite' while a value of (-2) implies 'unknown'.
AddressabilityCrossFeedDir	Read-Only	Integer32		The addressability of this marker in the cross feed direction in units of measure specified by **AddressabilityUnit**. A value of (-1) implies 'other' or 'infinite' while a value of (-2) implies 'unknown'.
NorthMargin	Read-Only	Integer32		The margin from the leading edge of the medium in units identified by **AddressabilityUnit**. Printing within this margin (and all others) is guaranteed for all interpreters. The value (-2) means unknown.
SouthMargin	Read-Only	Integer32		The margin from the trailing edge of the medium in units identified by **AddressabilityUnit**. Printing within this margin (and all others) is guaranteed for all interpreters. The value (-2) means unknown.
WestMargin	Read-Only	Integer32		The margin from the left edge of the medium in units identified by **AddressabilityUnit**. Printing within this margin (and all others) is guaranteed for all interpreters. The value (-2) means unknown.
EastMargin	Read-Only	Integer32		The margin from the right edge of the medium in units identified by **AddressabilityUnit**. Printing within this margin (and all others) is guaranteed for all interpreters. The value (-2) means unknown.
Status	Read-Only	Enum(1)	EntStatus	The current status of this marker device.

Printable Area Offsets and Extents

The printable area margins as listed above define an area of the print media which is guaranteed to be printable for all combinations of input, media paths, and interpreters for this marker.

Notes:

prtMarkerSupplies	Mandatory		prtMarkerSuppliesTable	
Object Name	**Access**	**Type**	**Value**	**Description**
MarkerSupplyID	Read-Only	Integer32		A unique value used by the printer to identify this marker supply. Although these values may change due to a major reconfiguration of the device (e.g. the addition of new marker supplies to the printer), values are expected to remain stable across successive printer power cycles.
MarkerID	Read-Only	Integer32		The Marker ID of the marking device with which this marker supply entity is associated.
ColorantID	Read-Only	Integer32		The Colorant ID of the colorant with which this marker supply entity is associated. This value shall be 0 if there is no colorant table.
Class	Read-Only	Enum (1)	MkrSupClass	Indicates whether this supply entity represents a supply container that is consumed or a receptacle that is filled.
Type	Read-Only	Enum(3)	MkrSupType	The type of this supply.
Description	Read-Only	String		The description of this supply container/receptacle in the localization specified by **CurrentLocalization**.
SupplyUnit	Read-Only	Enum(1)	MkrSupUnit	Unit of this marker supply container/receptacle.
MaxCapacity	Read-Only	Integer32		The maximum capacity of this supply container/receptacle expressed in **SupplyUnit**.
Level	Read-Only	Integer32		The current level if this supply is a container; the remaining space if this supply is a receptacle.

Notes:

prtMarkerColorant	Optional		prtMarkerColorantTable	
Object Name	**Access**	**Type**	**Value**	**Description**
ColorantID	Read-Only	Integer32		A unique value used by the printer to identify this colorant. Although these values may change due to a major reconfiguration of the device (e.g. the addition of new colorants to the printer), values are expected to remain stable across successive printer power cycles.
MarkerID	Read-Only	Integer32		The marker ID with which this colorant entry is associated.
Role	Read-Only	Enum (1)	ColorantRole	The role played by this colorant.
Value	Read-Only	String	ColorantValue	The name of the color of this colorant using standardized string names. Additional color string names may be added using ISO 9070 naming conventions.
Tonality	Read-Only	Integer32		The distinct levels of tonality realizable by a marking device when using this colorant. This value does not include the number of levels of tonal difference that an interpreter can obtain by techniques such as half toning. This value must be at least 2.

Notes:

Media Path Group Objects - Mandatory

prtMediaPath	Mandatory			
Object Name	**Access**	**Type**	**Value**	**Description**
DefaultID	Read-Write	Integer32		The ID of the default media path, i.e., the value of the corresponding media path's **ID** value.
MaxSpeedPrintUnit	Read-Only	Enum (1)	SpeedPrintUnit	The unit of measure used in specifying the speed of *all* media paths in the printer, i.e., the numerator of the units of measure of the media path speed.
MaxSpeedTimeUnit	Read-Only	Enum (1)	SpeedTimeUnit	The unit of time used in specifying the speed of *all* media paths in the printer, i.e., the denominator of the units of measure of the media path speed.
MediaSizeUnit	Read-Only	Enum (1)	PhyMediaUnit	The units of measure of media size for use in calculating and relaying dimensional values for *all* media paths in the printer.

Notes:

prtMediaPath	Mandatory		prtMediaPathTable	
Object Name	**Access**	**Type**	**Value**	**Description**
ID	Read-Only	Integer32		A unique value used by the printer to identify this media path. Although these values may change due to a major reconfiguration of the device (e.g. the addition of new media paths to the printer), values are expected to remain stable across successive printer power cycles.
MaxSpeed	Read-Only	Integer32		The maximum printing speed of this media path expressed in media path speed units (units of **MaxSpeedPrintUnit** per **MaxSpeedTimeUnit**). A value of (-1) implies 'other', while a value of (-2) implies 'unknown'.
MaxMediaFeedDir	Read-Only	Integer32		The maximum physical media size in the feed direction of this media path expressed in units of measure specified by **MediaSizeUnit**. A value of (-1) implies 'other' or 'infinite' while a value of -(-2) implies 'unknown'. A value of 0 implies unlimited.
MaxMediaAcrossFeedDir	Read-Only	Integer32		The maximum physical media size across the feed direction of this media path expressed in units of measure specified by **MediaSizeUnit**. A value of (-1) implies 'other' or 'infinite' while a value of (-2) implies 'unknown'.
MinMediaFeedDir	Read-Only	Integer32		The minimum physical media size in the feed direction of this media path expressed in units of measure specified by **MediaSizeUnit**. A value of (-1) implies 'other' while a value of (-2) implies 'unknown'.
MinMediaAcrossFeedDir	Read-Only	Integer32		The minimum physical media size across the feed direction of this media path expressed in units of measure specified by **MediaSizeUnit**. A value of (-2) implies 'unknown'.
MediaPathType	Read-Only	Enum(2)	MediaPathType	The type of media path for this media path.
MediaPathDescription	Read-Only	String		The manufacturer-provided description of this media path in the localization specified by **CurrentLocalization**.
Status	Read-Only	Enum	EntStatus	The current status of this media path.

Notes:

Channel Group Objects

Channels are sources of print data. There are many kinds of channels; some of which are based on networks and others which are not. For example, a channel can be a serial (or parallel) connection; it can be a service, such as the Unix Line Printer Daemon (LPD), offering itself over a network connection (interface); or it could be a disk drive into which a floppy disks with the print data is inserted.

Channel example	Implementation
serial port channel	bi-directional data channel
parallel port channel	often uni-directional channel
IEEE 1284 port channel	bi-directional channel
SCSI port channel	bi-directional
Apple PAP channel	may be based on Local-, Ether- or TokenTalk
LPD Server channel	typically TCP/IP based, port 515
Novell Remote Printer channel	typically SPX/IPX based
Novell Print Server channel	typically SPX/IPX based
port 9100 channel	HP and friends
Adobe AppSocket(9101) channel	a bi-directional extension of LPD

It is easy to note that this is a mixed bag. There are some physical connections over which no (or meager) protocols are run (e.g. the serial or old parallel ports) and there are services which often have elaborate protocols that run over a number of protocol stacks. In the end what is important is the delivery of print data through the channel.

Using the OSI seven layer model of for data interchange, the "channel" fits most accurately into the Application layer. Although this is not entirely clean (especially for the serial and old-parallel ports) it is clear that the interpreters that are consuming the print data are running at an application level.

The channel table and its underlying structure

The first five items in the Channel Table define the "channel" itself as seen at the OSI Application Layer (layer 7). This channel depends on lower layers of the OSI protocol, most often layers 1-4, to provide the data that flows through the channel. (Layer 1 is the physical connection; layer 2 is the data-link protocol, such as EtherNet; layer 3 is the inter-networking protocol, such as IP; and layer 4 is the transport protocol, such as TCP.) (Layers 5 (session) and 6(presentation) are most often irrelevant to the channel, but layer 6 could be used for character code conversion and/or decompression. A printer may be connected to print data sources in more than one way which means that there may be multiple variants at each of layers 1-4. Hence some way is needed to specify the "underlying" structure of each channel.

What is this underlying structure? A serial channel, has very little structure, except perhaps a protocol to allow binary data to be sent transparently over that channel along with control information. A network connection will typically have one or more variants at each of levels 1-4 of the OSI model. An example channel might have the Novell Remote Printer application protocol running on top of SPX (at layer 4) running on top of IPX (at layer 3) running on top of EtherNet (at layer 2) running over a large EtherNet cable (at layer 1).

In SNMP, there are groups which represent these various layers: the Interface Groups represent layers 1 and 2, including sub-layers. The inter-networking groups, such as the IP group represent layer 3 and its associated addressing. The Transmission groups, such as the TCP group and its associated connection table represent layer 4. The **IfTable** in the Interfaces Group has a which represents the various possible interfaces up to the Interment Level (level 3). There are no corresponding structures within SNMP that represent higher levels. For this reason, for this standard, the underlying structure will be represented only by identifying the topmost interface that underlies each channel.

The **prtChannelTable** represents the set of input data sources which can provide print data to one or more of the interpreters available on a printer

prtChannel	Mandatory		prtChannelTable	
Object Name	**Access**	**Type**	**Value**	**Description**
ID	Read-Only	Integer32		A unique value used by the printer to identify this data channel. Although these values may change due to a major reconfiguration of the device (e.g. the addition of new data channels to the printer), values are expected to remain stable across successive printer power cycles.
CurrentJobCntlLangID	Read-Write	Integer32		The value of this attribute is an Index that selects an entry in the **prtInterpreterTable**. This interpreter defines the syntax used for control functions, such as querying or changing environment variables and identifying job boundaries (e.g. PJL, PostScript, NPAP).
DefaultPageDescLangID	Read-Write	Integer32		The value of this attribute is an Index that selects an entry in the **prtInterpreterTable**. This interpreter defines the default Page Description Language interpreter to be used for the print data unless the Control Language is used to select a specific interpreter (e.g., PCL, PostScript Language, auto-sense).
State	Read-Write	Enum(1)	ChannelState	The state of this print data channel. The value determines whether control information and/or print data is allowed through this channel.
Type	Read-Only	Enum(2)	ChannelType	The type of this print data channel.
IfIndex	Read-Write	Integer		The value of **IfIndex** (in the **ifTable**; see the interface section of MIB-2/RFC 1573) which corresponds to this channel. When more than one row of the **ifTable** is relevant, this is the index of the row representing the topmost layer in the **if** hierarchy.
Status	Read-Only	Enum(1)	EntStatus	The current status of the channel.

Notes:

Console Group Objects

prtOperatorConsole				
Object Name	**Access**	**Type**	**Value**	**Description**
ConsoleLocalization	Read-Write	Integer32		An index into localization table giving the language, country, and character set to be used for the console.
NumberOfDisplayLines	Read-Only	Integer32		The number of lines on printers physical display.
NumberOfDisplayChars	Read-Only	Integer32		The number of characters per line displayed on the physical display.
ConsoleLock	Read-Write	Enum(2)	LockState	This object is used to lock or unlock the operatoon of the operators console.

prtOperatorConsole	Mandatory		prtDisplayBufferTable	
Object Name	**Access**	**Type**	**Value**	**Description**
LineText	Read-Write	String		This object is the content of a line in the logical display buffer of the operator's console of the printer. When a write operation occurs, normally a critical message, to one of the LineText strings, the agent should make that line displayable if a physical display is present. The printer should not overwrite this line once written until a zero length string is written to unlock that line. Printer generated strings should be in the localization specified by ConsoleLocalization. Management Application generated strings should be localized by the Management Application.

Notes:

prtOperatorConsole	Mandatory		prtConsoleLightsTable	
Object Name	**Access**	**Type**	**Value**	**Description**
LightID	Read-Only	Integer32		A unique value used by the printer to identify this light. Although these values may change due to a major reconfiguration of the device (e.g. the addition of new lights to the printer), values are expected to remain stable across successive printer power cycles.
OnTime	Read-Only	Integer32		The on time in milliseconds of blinking of this light; 0 indicates off always.
OffTime	Read-Only	Integer32		The off time in milliseconds of blinking of this light; 0 indicates on always.
Color	Read-Only	Enum(2)	LightColor	The color of this light.
Description	Read-Only	String		The vendor description or label of this light in the localization specified by **ConsoleLocalization**.

Notes:

Interpreter Group Objects - Mandatory

prtInterpreter	Mandatory			
Object Name	**Access**	**Type**	**Value**	**Description**
DefaultID	Read-Write	Integer32	IntpLangFam	The ID of the default interpreter, i.e., the value of the corresponding interpreter's **ID,** to be used in case no device specific method is used to specify which interpreter should be used. Priority on selecting the interpreter are 1) Commands in the data stream; 2) The default interpreter for the channel on which the data arrives; and lastly 3) This object.

The **prtInterpreterTable** is a table representing the interpreters in the printer. An entry shall be placed in the interpreter table for each interpreter on the printer.

prtInterpreter	Mandatory		prtInterpreterTable	
Object Name	**Access**	**Type**	**Value**	**Description**
ID	Read-Only	Integer32		A unique value for each PDL or control language for which there exists an interpreter or emulator in the printer. The value is used to identify this interpreter. Although these values may change due to a major reconfiguration of the device (e.g. the addition of new interpreters to the printer), values are expected to remain stable across successive printer power cycles.
LangFamily	Read-Only	Enum(2)	IntpLangFam	The family name of a Page Description Language (PDL) or control language which this interpreter in the printer can interpret or emulate. This list of enumerations requires review before additional entries are made.
LangLevel	Read-Only	String		The level of the language which this interpreter is interpreting or emulating. This might contain a value like '5e' for an interpreter which is emulating level 5e of the PCL language. It might contain '2' for an interpreter which is emulating level 2 of the PostScript language. Similarly it might contain '2' for an interpreter which is emulating level 2 of the HPGL language.
LangVersion	Read-Only	String		The date code or version of the language which this interpreter is interpreting or emulating.
Description	Read-Only	String		A string to identify this interpreter in the localization specified by **CurrentLocalization** as opposed to the language which is being interpreted. It is anticipated that this string will allow manufacturers to unambiguously identify their interpreters.
Version	Read-Only	String		The date code, version number, or other product specific information tied to this interpreter. This value is associated with the interpreter, rather than with the version of the language which is being interpreted or emulated.
FeedAddressability	Read-Only	Integer32		The maximum interpreter resolution in the feed direction in **prtMarkerAddressabilityUnit** (the same units of measure used by the object used to specify marker addressability) for this interpreter. If this interpreter has no limit on addressability the value is zero. The value (-1) means other and specifically indicates that the device places no restrictions on this parameter. The value (-2) means unknown.
CrossFeedAddressability	Read-Only	Integer32		The maximum interpreter resolution in the across feed direction in **prtMarkerAddressabilityUnit** (the same units of measure used by the object used to specify marker addressability) for this interpreter. If this interpreter has no limit on addressability the value of this object is zero. The value (-1) means other and specifically indicates that the device places no restrictions on this parameter. The value (-2) means unknown.
DefaultCharSetIn	Read-Write	Integer32		An index into the **CharacterSetTable** that identifies the default character set for data into this interpreter.
DefaultCharSetOut	Read-Write	Integer32		An index into the **CharacterSetTable** that identifies the default character set for data coming from this interpreter through the printer's output channel
TwoWay	Read-Only	Enum(1)	YesNo-OnOff	Indicates whether or not this interpreter returns information back across the data channel to the host.

Notes:

Alerts Group Object - Mandatory

The **prtAlertTable** lists all the critical and non-critical alerts currently active in the printer. A critical alert is one that stops the printer from printing immediately and printing can not continue until the critical alert condition is eliminated. Non-critical alerts are those items that do not stop printing but may at some future time. The table contains information on the severity, component, detail location within the component, alert code and description of each critical alert that is currently active within the printer.

prtAlert	Mandatory		prtAlertTable	
Object Name	**Access**	**Type**	**Value**	**Description**
Index	Read-Only	Integer32		The index value used to determine which alerts have been added or removed from the alert table. This in an incrementing integer starting from zero when the printer is powered on. When the printer adds an alert to the table, that alert is assigned the next higher integer value from the last item entered into the table. If the index value reaches its maximum value, the next item entered will cause the index value to roll over and start at zero again. The first event placed in the alert table after power up initialization of the printer will have an index value of 1.
SeverityLevel	Read-Only	Enum (1)	SeverityLevelAlertCode	The level of severity of this alert table entry. The printer determines the severity level assigned to each entry into the table.
TrainingLevel	Read-Only	Enum (2)	TrainingLevel	The level of training required to handle this alert. The training level is an enumeration that is determined and assigned by the printer manufacturer based on the information or the training required to handle this alert. The printer will break alerts into these different training levels. It is the responsibility of the management application in the system to determine how a particular alert is handled and how and to whom that alert is routed.
Group	Read-Only	Enum (1)	AlertGroupCode	The sub-unit within the printer model that this alert is related. Input, output, and markers are examples of printer model sub-units.
Group Index	Read-Only	Integer32		An integer designation, assigned by the printer manufacturer, of the specific sub-unit within the group that caused this alert. The combination of the **Group** and the **GroupIndex** defines exactly which printer sub-unit caused the alert. Input #3, Output #2, and Marker #1 are examples.
Location	Read-Only	Integer32		The sub-unit location that is defined by the printer manufacturer to further refine the location of this alert with the designated sub-unit. The location is used in conjunction with the **Group** and **Index** values. Example: There is an alert in Input #2 at location number 7.
Alert Code	Read-Only	Enum (2)		The code that describes the type of alert for this entry in the table. There are different codes for each sub-unit type. Jam, Media Low, Toner Out, and Output Full are examples of different codes for various sub-unit types.
Description	Read-Only	String		A description of this alert entry in the localization specified by **CurrentLocalization**. The description is provided by the printer to further elaborate on the enumerated alert or provide information in the case where the code is classified as “other” or “unknown”. The printer is required to return a description string but the string may be a null string.

Notes:

Group Status

EntStatus	
1	Other
2	Unknown
3	Ready/Available
4	Critical State
5	Non-critical state
6	Both Critical and Non-critical state
7	On-request
8	Unavailable
9	Busy/Temp Unavailable

PMI Enumerations - Units of measure

AddressabililtyUnit	**MediaUnit**	**CounterUnit**
1 Other	1 Other	1 Other
2 Unknown	2 Unknown	2 Unknown
3 .0001 Inches	3 .0001 Inches	3 .0001 inches
4 .001 millimeters	4 .001 Millimeters	4 .001 millimeters
	8 Sheets	5 Characters
		6 Lines
		7 Impressions
		8 Sheets
		9 Dot-Row
		11 Hours
		16 Feet
		17 Meters
SpeedPrintUnit	**SpeedTimeUnit**	**FinisherConsumableUnit**
1 Other	1 Other	1. Other
2 Unknown	2 Unknown	2. Unknown
3 .0001 inches	11 Hours	3. .0001 inches
4 .001 millimeters		4. .001 millimeters
5 Characters		8. Sheets
6 Lines		12. .001 ounces
7 Impressions		13. .1 grams
8 Sheets		14. .01 fluid ounces
9 Dot-Row		15. .1 milliliters
		16. Feet
		17. Meters
		18. Each
MarkerSupplyUnit	**CapacityUnit**	
1 Other	1 Other	
2 Unknown	2 Unknown	
3 .0001 inches	3 .0001 inches	
4 .001 millimeters	4 .001 millimeters	
8 Sheets	8 Sheets	
12 .001 ounces	16 Feet	
13 .1 grams	17 Meters	
14 .01 fluid ounces		
15 .1 milliliters		

PMI Enumerations - Device Types

InputType	OutputType	MediaPathType
1 Other	1 Other	1 Other
2 Unknown	2 Unknown	2 Unknown
3 Sheet-feed-auto-removable-tray	3 Removable Bin	3 Long Edge Binding Duplex
4 sheet-feed-auto-non removable tray	4 UnRemovable Bin	4 Short Edge Binding Duplex
5 Sheet Feed Manual	5 Continuous Roll Device	5 Either edge Duplex
6 continuous-roll	6 Mail Box	6 Simplex
7 continuous-fan-fold		
FinisherType	**SkinStatus**	**MarkerSupplyClass**
1 Other	1 Other	1 Other
2 Unknown	2 Unknown	2 Unknown
3 Stitching	3 Door Open	3 Supply that is consumed
4 Binding	4 Door Closed	4 Receptacle that is filled
5 Trimming	5 Interlock Open	
6 Die-cutting	6 Interlock Closed	
7 Punching		
8 Perforating		
9 Slitting		
10 Inserting		
11 Covers		
12 Folding		
MarkingTechnology	**ChannelType**	**MarkerSupplyType**
1 Other	1. Other	1 Other
2 Unknown	2. Unknown	2 Unknown
3 Electophotographic:LED	3. Serial Port	3 Toner
4 Electrophotographic:Laser	4. Parallel Port	4 Waste Toner
5 Electrophotographic:Other	5. IEEE1284 Port	5 Ink
6 Impact:Moving_Head:Dot_Matrix:9pin	6. SCSI Port	6 Ink Cartridge
7 Impact:Moving_Head:Dot_Matrix:24pin	7. Apple PAP	7 Ink Ribbon
8 Impact:Moving_Head:Dot_Matrix:other	8. LPD Server	8 Waste Ink
9 Impact:Moving_Head:Fully_Formed	9. Novell RPrinter	9 OPC (Organic Photo Conductor)
10 Impact:band	10. Novell Pserver	10 Developer
11 Impact:other	11. port 9100	11 Fuser Oil
12 Inkjet:Aqueous	12. Adobe AppSocket (9101)	12 Wax, Solid
13 Inkjet:Solid	13. FTP ("PUT" to printer)	13 Wax, ribbon
14 Inkjet:Other	14. TFTP (to printer)	14 Waste Wax
15 Pen	15. DLC/LLC Port	
16 Thermal:Transfer	16. IBM 3270	
17 Thermal:Sensitive	17. IBM 5250	
18 Thermal:Diffusion	18. Fax	
19 Thermal:Other	19. IEEE1394	
20 Electroerosion	20. Transport 1 (port 35)	
21 Electrostatic	21. CPAP (port 170)	
22 Photographic:Microfiche	22. IPC (?)	
23 Photographic:Imagesetter	23. OLE	
24 Photographic:Other	24. Named Pipe	
25 Ion-deposition		
26 E-beam		
27 Typesetter		

IntpLangFam - Interpreter Language Family

Value	Enum symbolic name	Description - becomes a MIB/MIF comment
1	other(1)	
2	unknown(2)	
3	PCL(3)	Printer Control Language, or Printer Command Language Starting with version 5, HP-GL/2 is part of PCL. Hewlett-Packard Co.
4	HPGL(4)	Hewlett-Packard Graphics Language. Originally used to control pen movements on plotters.Includes many device specific commands. Hewlett-Packard Co.
5	PJL(5)	Peripheral Job Language. Appears in the data stream between data intended for a page description language. Hewlett-Packard Co.
6	PostScript(6)	Adobe Systems Incorporated Mountain View, CA
7	IPDS(7)	Intelligent Printer Data Stream IBM
8	PPDS(8)	IBM Personal Printer Data Stream. Originall called IBM ASCII, the name was changed to PPDS when the LaserPrinter was introduced in 1989. Lexmark International, Inc.
9	EscapeP(9)	
10	Epson(10)	
11	DDIF(11)	Digital Document Interchange Format Digital Equipment Corp.
12	Interpress(12)	
13	ISO-6429(13)	ISO 6429. Control functions for Coded Character Sets (has ASCII control characters, plus additional controls for character imaging devices.) ISO Standard, Geneva, Switzerland
14	line-data(14)	Lines of data as separate ASCII or EBCDIC records and containing no control functions (no CR, LF, HT, FF, etc.)- traditional line printers
15	MODCA(15)	
16	ReGIS(16)	Remote Graphics Instruction Set, DIgital Equipment Corp, Maynard MA
17	SCS(17)	
18	SPDL(18)	ISO 10180 Standard Page Description Language ISO Standard, Geneva, Switzerland
19	Tek4014(19)	
20	PDS(20)	
21	IGP(21)	
22	Code-V(22)	Magnum Code-V, Image and printer control language used to control impact/dot-matrix printers. QMS, Inc., Mobile AL
23	DSC-DSE(23)	
24	WPS(24)	Windows Printing System, Resource based command/data stream used by Microsoft At Work Peripherals. Developed by the Microsoft Corporation.
25	LN03(25)	Early DEC-PPL3, Digital Equipment Corp.
26	CCITT(26)	
27	QUIC(27)	QUIC (Quality Information Code), Page Description Language for laser printers. Included graphics, printer control capability and emulation of other well-known printer languages. QMS, Inc. Mobile, AL
28	CPAP(28).	Common Printer Access Protocol Digital Equipment Corp.
29	DEC-PPL(29)	Digital ANSI-Compliant Printing Protocol Digital Equipment Corp.
30	simple-text(30)	Character coded data, including NUL, CR , LF, HT, and FF control characters. See ISO 10175 Document Printing Application (DPA) ISO standard, Geneva, Switzerland
31	NPAP(31)	Network Printer Alliance Protocol
32	DOC(32)	Document Option Commands, Appears in the data stream between data intended for a page description language. QMS, Inc. Mobile, AL

33	IMPRESS(33)	imPRESS, Page description language originally developed for the ImageServer line of systems. A binary language providing representations for text, simple graphics (rules,lines,conic sections), and some large forms (simple bit-map and CCITT group 3/4 encoded).The language was intended to be sent over an 8-bit channel and supported early document preparation languages (e.g. TeX and TROFF). QMS, Inc. Mobile, AL
34	Pinwriter(34)	24 wire dot matrix printer language for USA, Europe, and Asia except Japan. More widely used in Germany, and some Asian countries than in US. NEC
35	NPDL(35)	Page printer language for Japanese market. This is one of the majors in Japanese market. NEC
36	201PL(36)	Serial printer language used in the Japanese market.This is one of the dominant in Japanese market. NEC
37	automatic	Automatic sensing of the interpreter language family by the printer examining the document content. Which actual interpreter language families are sensed depends on the printer implementation.

PMI Enumerations - Misc

MediaOrientation	MediaColor	LineStatus
1 Other	1 Other	1 Other
2 Unknown	2 Unknown	2 Unknown
3 Portrait	3 White	3 On Line
4 Reverse Portrait	4 Pink	4 Pending On Line
5 Landscape	5 Yellow	5 Off Line
6 Reverse Landscape	6 Buff	6 Pending Off Line
	7 Goldenrod	
	8 Blue	
	9 Green	
	10 Transparent	
NamedFinshing	**ColorantValue - String**	**LightColor**
1 Other	Other	1. Other
2 Unknown	Unknown	2. Unknown
3 Staple	White	3. White
4 Staple Top Left	Red	4. Red
5 Staple Bottom Left	Green	5. Green
6 Stable Top Right	Blue	6. Blue
7 Staple Bottom Right	Cyan	7. Cyan
8 Saddle Stitch	Magenta	8. Magenta
9 Edge Stitch	Yellow	9. Yellow
10 Punch	Black	
11 Cover		
12 Bind	Additional names using ISO 9070 naming convention.	
YesNo-OnOff	**ResetType**	**ColorantRole**
1. Other	3. Power Cycle Reset	1. Other
2. Unknown	4. Reset NVRAM	2. Unknown
3. Yes - On	5. Reset to Factory Defaults	3. Process
4. No - Off		4. Spot
TrainingLevel	**ChannelState**	**PageDeliveryOrientation**
1 Other	1. Other	1. Other
2 Unknown	2. Unknown	2. Unknown
3 Untrained	3. Print and Control Data Accepted	3. Face-Up
4 Trained	4. No Data Accepted	4. Face-Down
5 Field Service		
6 Management		
StackingOrder	**LockState**	
1 Other	1. Other	
2 Unknown	2. Unknown	
3 1 to n	3. Locked	
4 n to 1	4. Unlocked	

PMI Enumerations - Alert Codes

SeverityLevelAlertCode

1	Other
2	Unknown
3	Critical
4	Warning

AlertGroupCode

1	Other
2	Unknown
3	Input
4	Output
5	Finisher
6	Media Path
7	General Printer
8	Marker
9	System Controller
10	Interface
11	Interpreter
12	Operator Console
13	Channel

InputAlertCode

1	Other
2	Unknown
3	Cover Open
4	Cover Closed
5	Interlock Open
6	Interlock Closed
7	Configuration Change
8	Jam
9	Media Tray Missing
10	Media Size Change
11	Media Weight Change
12	Media Type Change
13	Media Color Change
14	Media Form Parts Change
15	Media Supply Low
16	Media Supply Empty

OutputAlertCode

1	Other
2	Unknown
3	Cover Open
4	Cover Closed
5	Interlock Open
6	Interlock Closed
7	Configuration Change
8	Jam
9	Media Tray Missing
10	Media Tray Almost Full
11	Media Tray Full

MarkerAlertCode

1	Other
2	Unknown
3	Cover Open
4	Cover Closed
5	Interlock Open
6	Interlock Closed
7	Configuration Change
8	Jam
9	Toner Empty
10	Ink Empty
11	Print Ribbon Empty
12	Toner Almost Empty
13	Ink Almost Empty
14	Print Ribbon Almost Empty
15	Waste Toner Receptacle Almost Full
16	Waste Ink Receptacle Almost Full
17	Waste Toner Receptacle Full
18	Waste Ink Receptacle Full
19	Fuser Under Temperature
20	Fuser Over Temperature
21	OPC Life Almost Over
22	OPC Life Over
23	Developer Almost Empty
24	Developer Empty

GeneralPrinterAlertCode

1	Other
2	Unknown
3	Cover Open
4	Cover Closed
5	Interlock Open
6	Interlock Closed
7	Door Open
8	Door Closed
9	Power Up
10	Power Down

MediaPathDeviceAlertCode

1	Other
2	Unknown
3	Cover Open
4	Cover Closed
5	Interlock Open
6	Interlock Closed
7	Configuration Change
8	Jam
9	Media Tray Missing
10	Media Tray Almost Full
11	Media Tray Full

FinisherAlertCode

1	Other
2	Unknown
3	Cover Open
4	Cover Closed
5	Interlock Open
6	Interlock Closed
7	Configuration Change
8	Jam
9	Media Tray Missing
10	Supply Empty
11	Supply Low
12	Receptacle Almost Full
13	Receptacle Full

InterpreterAlertCode

1	Other
2	Unknown
3	Configuration Change
4	Memory Increase
5	Memory Decrease
6	Cartridge Added
7	Cartridge Deleted
8	Resource Added
9	Resource Deleted
10	Resource Unavailable

InterfaceAlertCode

1	Other
2	Unknown
3	Configuration Change

ChannelAlertCode

1	Other
2	Unknown
3	Configuration Change

Appendices

Appendix A: Glossary of terms

Alert -- a reportable event for which there is an entry in the alert table

Bin -- an output sub-unit which may or may not be removable

Component -- a complete product, such as a printer. (DMTF usage)

Critical Alert -- an alert triggered by an event which leads to a state in which printing is no longer possible; the printer is stopped

Description -- information about the configuration and capabilities of the printer and its various sub-units

Group -- a collection of attributes or objects that represent a sub-unit of a component

Input -- a tray or bin from which instances of the media are obtained and fed into the Media Path

Localization -- the specification of human language, country, and character set needed to present information to people in their native languages.

Management Application (a.k.a. Manager) -- a program which queries and controls one or more managed nodes

Management Station -- a physical computer on which one or more management applications can run

Media Path -- the mechanisms that transport instances of the media from an input, through the marker, possibly through media buffers and duplexing pathways, out to the output with optional finishing applied. The inputs and outputs are not part of the Media Path.

Non-critical Alert -- an alert triggered by a reportable event which does not lead to a state in which printing is no longer possible; such an alert may lead to a state from which printing may no longer be possible in the future, such as the low toner state or the alert may be pure informational, such as a configuration change at the printer.

Output -- a bin or stacker which accepts instances of media that have been processed by a printer

Printer -- a physical device that takes media from an input source, produces marks on that media according to some page description or page control language and puts the result in some output destination, possibly with finishing applied.

Printing -- the entire process of producing a printed document from generation of the file to be printed, choosing printing properties, selection of a printer, routing, queuing, resource management, scheduling, and finally printing including notifying the user

Reportable event -- an event that is deemed of interest to a management station watching the printer

Status -- information regarding the current operating state of the printer and its various sub-units. This is an abstraction of the exact physical condition of the printer.

Sub-mechanism -- a distinguishable part of a sub-unit

Sub-unit -- a part of the printer which may be a physical part, such as an input source or a logical part such as an interpreter.

Tray -- an input sub-unit which is typically removable

Visible state -- that portion of the state of the printer that can be examined by a management application

Appendix B - Media size names

This appendix lists the standardized media size names from ISO/IEC 10175 Document Printing Application (DPA). Management application developers are encouraged to use these media sizes in order to display a media size name to the user, since the names are more likely to be familiar to them than the numerical sizes returned as the values of **prtInputMediaDimAccrossFeedDir** and **prtInputMediaDimFeedDir**.

String name	Description
other	
unknown	
na-letter or letter	North American letter size: 8.5 by 11 inches
na-legal or legal	North American legal size: 8.5 by 14 inches
na-10x13-envelope	North American 10x13 size: 10 by 13 inches
na-9x12-envelope	North American 9x12 size: 9 by 12 inches
na-number-10-envelope	North American number 10 business envelope
na-7x9-envelope	North American 7x9 size: 7 by 9 inches
na-9x11-envelope	North American 9x11 size: 9 by 11 inches
na-10x14-envelope	North American 10x14 size: 10 by 14 inches
na-number-9-envelope	North American number 9 business envelope
na-6x9-envelope	North American 6x9 size: 6 by 9 inches
na-10x15-envelope	North American 10x15 size: 10 by 15 inches
a	engineering A size
b	engineering B size
c	engineering C size
d	engineering D size
e	engineering E size
iso-a0	ISO A0 size: 841 mm by 1189 mm
iso-a1	ISO A1 size: 594 mm by 841 mm
iso-a2	ISO A2 size: 420 mm by 594 mm
iso-a3	ISO A3 size: 297 mm by 420 mm
iso-a4	ISO A4 size: 210 mm by 297 mm
iso-a5	ISO A5 size: 148 mm by 210 mm
iso-a6	ISO A6 size: 105 mm by 148 mm
iso-a7	ISO A7 size: 74 mm by 105 mm
iso-a8	ISO A8 size: 52 mm by 74 mm
iso-a9	ISO A9 size: 37 mm by 52 mm
iso-a10	ISO A10 size: 26 mm by 37 mm
iso-b0	ISO B0 size: 1000 mm by 1414 mm
iso-b1	ISO B1 size: 707 mm by 1000 mm
iso-b2	ISO B2 size: 500 mm by 707 mm
iso-b3	ISO B3 size: 353 mm by 500 mm
iso-b4	ISO B4 size: 250 mm by 353 mm
iso-b5	ISO B5 size: 176 mm by 250 mm
iso-b6	ISO B6 size: 125 mm by 176 mm
iso-b7	ISO B7 size: 88 mm by 125 mm
iso-b8	ISO B8 size: 62 mm by 88 mm
iso-b9	ISO B9 size: 44 mm by 62 mm
iso-b10	ISO B10 size: 31 mm by 44 mm
iso-c3	ISO C3 size: 229 mm by 324 mm
iso-c4	ISO C4 size: 229 mm by 324 mm
iso-c5	ISO C5 size: 162 mm by 229 mm
iso-c6	ISO C6 size: 162 mm by 229 mm
iso-designated	ISO Designated Long size: 110 mm by 220 mm
jis-b0	JIS B0 size
jis-b1	JIS B1 size
jis-b2	JIS B2 size
jis-b3	JIS B3 size
jis-b4	JIS B4 size
jis-b5	JIS B5 size
jis-b6	JIS B6 size
jis-b7	JIS B7 size

jis-b8	JIS B8 size
jis-b9	JIS B9 size
jis-b10	JIS B10 size

Appendix Y - Issues

This appendix lists the issues that have been raised during the meetings and in between meetings. The issues listed under each MS-WORD table under NOTES: are additional issues which are not included here.

1) Is the media group optional?

- Leave MediaExtended as optional, media information in the InputTable will remain mandatory.

2) Will the MIB/MIF support multiple consoles?

- No, do with private extensions, a single "console" that contains the data for all the consoles, or some other means.

3) Is General Printer a table?

- No, support Virtual printers through some other means such as private extensions, etc.

4) How does "character set" tie into "channel"?

- There was no identified need to tie character set to channels. Character Set is more closely tied to the interpreter or console and those objects are tied to localization and character set settings.

5) Static versus dynamic - how efficient will this be?

- Because there is no distinction between static and dynamic information in SNMP, the printer must implement all the information so the agent can report it.

6) The MIB/MIF consist of 58 objects - is that too many?

- OPEN --- Defer for final review.

7) Should we endorse some type of interlock method for all the writable objects?

- OPEN --- One of the reasons it remains open is that there is no defined method for this function for DMTF.

8) Should we use separate community names or additional index to support multiple printers being managed through a printer server?

- Use community names with predictable "numbering" for each device. Randy Turner (QMS) will write up a proposal for inclusion in the RFC.

9) Should paper size be writable?

- Yes, to tell a printer that can't detect paper size what size paper is loaded.

10) Should units be writable?

- Yes, but manufacturers may choose to only implement R/O access. Persistance across power cycles may vary from manufacturer to manufacturer.

11) Should size units be only in millimeters?

- No, but we will reduce the number of units wherever possible.

12) How should out of range sets be handled - reject or snap to the nearest?

- Snap to the nearest integer and reject an invalid ENUM. The data returned in the SNMP packet will indicate the value of the integer snapped to.

13) Are "Input - dimensions extended" mandatory?

- This will remain at best optional; possibly could be removed.

14) In what units is printable area reported?

- .0001 inches or .001 mm

15) Should media type be a string or an enum?

- String, recommend using ISO DPA, ISO SPDL names where possible.

16) How do we associate media with inputs, outputs with finishers, etc?

- Input: The only media that is described is the media in an input tray and it is described there.

- Outputs: OPEN - Harry Lewis to make a proposal

17) Do margins (or imagable area) need to be PDL specific?

- No, but this object may not be needed.

18) Does "other" have a meaning for objects which normally have an integer value?

- Yes - potential meanings are defined with the object where it applies.

19) Is "unknown" the same as "uninitialized" and if not how do you tell the difference?

- Possibly, if the object's value is "unknown" failing when attempting to write to it means the object is truly unknown. If writing is successful, the object's value was just unitialized.

20) Should default interpreter be by channel or overall?

- OVERALL. There is a hierarchy used to determine the interpreter:

1) If contained in the datastream, used that interpreter

2) Otherwise use the default for the channel

3) Otherwise use the default for the interpreters

4) Otherwise the default interpreter is undefined or manufacturer specific

21) How do we handle the "parallel in, serial out" channel?

- Ignore it

22) Is "console width" "string width" or "screen width"?

- Closed by addressing issue #23.

23) Is "number of lines" and "number of characters" of the console mean physically or logical?

- They refer to the physical size of the console.

24) Which strings are localized and controlled by "current localization?"

- Spec has been updated to reflect which strings are localized by CurrentLocalization, which strings are localized by ConsoleLocalization and which strings are not localized. String not specifically stated to be localized should not be localized as that data may be used to make decisions.

25) Do we need an "ID" object for each table "index?"

- OPEN

26) Are all consoles using the same language?

- We only support one console.

27) Should we enable or disable a channel as a whole or provide a way to selectively enable and disable "data" and "control" flows on a channel?

- No convincing argument for separation of control and data. Therefore, the answer is no.

28) Is the consistent use of 1,2 (for enums) and -1, -2 (for integers) to represent "other" and "unknown" consistent with SNMP "tradition?"

- There is a tradition of having certain distinguished values but those values often vary from RFC to RFC. Consistant of use of these values within the PRINTER MIB should be OK.

Appendix Z - Change History

This appendix list the changes made to each version of the specification

Changes from Version 0.6, 26-Apr, to Version 0.7, 27-apr

1. Added console lock
2. Changed printable area to margins and moved to marker
3. Worked through all notes and removed after resolving the issues.
4. Updated issues list based upon second day of meeting (April 27, 1994)
5. Added finisher consumables units

Changes from Version 0.2, 25-Apr, to Version 0.6, 26-apr

1. Cleaned-up formating
2. Updated issues list based upon first day of meeting in April.
3. Changed version number to allign with original PMI_OBJ series of documents.
4. Hide Tom Hastings editorial comments on page 2.
5. This version was only distributed at the April IETF/DMTF meeting.

Changes from Version 0.1, 23-Apr, to Version 0.2, 25-apr

1. Added Appendix Y, List of Issues from the ftp-boi.external.hp.com directory.
2. Replaced Interpreter Language Family enum table with the contents of the Interpreter Spec, so that the comments are included as well for each enum value. Now this specification will be the only copy of the enum values assigned.

Changes to create Version 0.1, 23-Apr

1. Started with Version 0.5 of the Object Summary that Don edited with agreements reached at the March Seattle meeting.
2. Copied in the descriptions from the following specifications that the working group had reviewed: and had been posted in the ftp-boi.external.hp.com server:

number	file name	date	groups
1	general/pmi_obj5.doc	22-Apr	all
2	channel_class/channel3.mif	19-Apr	Channel
3	channel2 mail	28-Mar	Console
4	source_dest_class/pmi_inp_outp _V0.3.txt	23-Feb	Input, Output, Finisher
5	interpreters_class/interpreters.05	21-Apr	Interpreter
6	localization/localization.objects	16-Mar	General
7	printer_class/prt_mif4.doc	21-Feb	Marker, MediaPath
8	general/pmi_modl.doc	30-Mar	Alerts
9	general/glossary.txt	29-Mar	Appendix

My MS-WORD for Windows 2.0a could not read the pmi_mdl3.doc, 19-Apr.

3. Changed the names of objects/attributes and enum symbols that had the word **Units** to **Unit** to be consistent with the rest that had just **Unit** Also changed **Dims** to **Dim** to be consistent with the rest that had just **Dim**.
4. Changed the terminology of the descriptions that used "units" to "unit of measure", so as to avoid confustion with hardware units and sub-units.
5. Spell checked the entire document and created a custom dictionary, **pmi_spec.dic**.

6. Changed the names of a few of the Language Families that had come from the ISO DPA, to agree with the names in the ISO DPA. In half of the cases, the ISO DPA committee agreed to change their names. The following changes were made in this specification: 1) LinePrinter to line-data, 2) SimpleText to simple-text.

7. Changed the Language Family enum values that had been added to the Object Summary V0.5, after the Seattle meeting to agree with the 21-Apr version of the Interpreter specification: interpreters.05.

8. Added automatic as enum 37, since the committee had agree in Seattle and SnowBird to add automatic as a PDL for which the printer determines which interpreter to use by looking at the print data itself.

9. Editorially changed the beginning of each object/attribute description to be a noun that is the value and start each with the definite article: "The" or "A".

10. Removed the terms object in the object/attribute descriptions since IETF MIBs use the term objects, while the DMTF MIF use the term attribute. Still need to change the title of the specification though, since it containes “Object”.