

Internet Draft

Avi Berger
PowerDsine Inc.
Dan Romascanu
Avaya Inc.
25 June 2001

Power Ethernet (DTE Power via MDI) MIB

<[draft-ietf-hubmib-power-ethernet-mib-00.txt](#)>

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of [Section 10 of RFC2026](#). Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/lid-abstracts.html>

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

Copyright Notice

Copyright (C) The Internet Society (2001). All Rights Reserved.

Abstract

This memo defines a portion of the Management Information Base (MIB) for use with network management protocols in the Internet community. The document proposes an extension to the Ethernet-like Interfaces MIB [[RFC2665](#)] with a set of objects for managing a power Ethernet Powered Device (PD) and/or Power Source Equipment (PSE).

Distribution of this memo is unlimited.

Table of Contents

Status of this Memo	1
Abstract	1
1 Introduction	2

3 Overview	3
4 MIB Structure	4
5 Evolution of the Document, Limitations and Future Work	4
6 Changes log	4
7 Definitions	5
8 References	20
9 Intellectual Property	22
10 Security Considerations	22
11 Authors Addresses	23
A Full Copyright Statement	23

1. Introduction

This memo defines a portion of the Management Information Base (MIB) for use with network management protocols in the Internet community. In particular, it defines a set of MIB objects to manage a Power Ethernet (DTE Power via MDI) Powered Device (PD) and/or power Source Equipment (PSE).

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 [[RFC2863](#)].

2. The SNMP Management Framework

The SNMP Management Framework presently consists of five major components:

- o An overall architecture, described in [RFC 2571](#) [[RFC2571](#)].
- o Mechanisms for describing and naming objects and events for the purpose of management. The first version of this Structure of Management Information (SMI) is called SMIV1 and described in STD 16, [RFC 1155](#) [[RFC1155](#)], STD 16, [RFC 1212](#) [[RFC1212](#)] and [RFC 1215](#) [[RFC1215](#)]. The second version, called SMIV2, is described in STD 58, [RFC 2578](#) [[RFC2578](#)], STD 58, [RFC 2579](#) [[RFC2579](#)] and STD 58, [RFC 2580](#) [[RFC2580](#)].
- o Message protocols for transferring management information. The first version of the SNMP message protocol is called SNMPv1 and

described in STD 15, [RFC 1157](#) [[RFC1157](#)]. A second version of the SNMP message protocol, which is not an Internet standards track protocol, is called SNMPv2c and described in [RFC 1901](#) [[RFC1901](#)] and [RFC 1906](#) [[RFC1906](#)]. The third version of the message protocol is called SNMPv3 and described in [RFC 1906](#) [[RFC1906](#)], [RFC 2572](#) [[RFC2572](#)] and [RFC 2574](#) [[RFC2574](#)].

- o Protocol operations for accessing management information. The

first set of protocol operations and associated PDU formats is described in STD 15, [RFC 1157](#) [[RFC1157](#)]. A second set of protocol operations and associated PDU formats is described in [RFC 1905](#) [[RFC1905](#)].

- o A set of fundamental applications described in [RFC 2573](#) [[RFC2573](#)] and the view-based access control mechanism described in [RFC 2575](#) [[RFC2575](#)].

A more detailed introduction to the current SNMP Management Framework can be found in [RFC 2570](#) [[RFC2570](#)].

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. Objects in the MIB are defined using the mechanisms defined in the SMI.

This memo specifies a MIB module that is compliant to the SMIV2. A MIB conforming to the SMIV1 can be produced through the appropriate translations. The resulting translated MIB must be semantically equivalent, except where objects or events are omitted because no translation is possible (use of Counter64). Some machine readable information in SMIV2 will be converted into textual descriptions in SMIV1 during the translation process. However, this loss of machine readable information is not considered to change the semantics of the MIB.

[3.](#) Overview

The emergence of IP telephony as an application that allows for voice applications to be run over the same infrastructure as data applications led to the emergence of Ethernet IP phones, with similar functions and characteristics as the traditional phones. Powering a phone is one of these functions that are being taken as granted. The

IEEE 802.3 Working Group initiated a standard work on this subject, currently known as the IEEE 802.3af work [[IEEE-802.3af](#)].

The IEEE 802.3af WG will not define a full management interface, but only the hardware registers that will allow for a management interfaces to be built for a powered Ethernet device. The MIB module defined in this document extends the Ethernet-like Interfaces MIB [[RFC2665](#)] with the management objects required for the management of the powered Ethernet devices and ports.

The following abbreviations are defined in [[IEEE-802.3af](#)] and will be used with the same significance in this document: PSE - Power Sourcing Equipment; PD - Powered Device

Ethernet MIB WG

Expires December 2001

[Page 3]

INTERNET DRAFT

Power Ethernet MIB

June 2001

[4.](#) MIB Structure

This MIB objects are included in four MIB groups - three of them include MIB tables, and the fourth scalar objects

The pethPsePortTable defines the objects used for the configuration and describing the status of ports on a PSE device. Examples of PSE devices are Ethernet switches that support power Ethernet and mid-span boxes.

The pethPdPortTable defines the objects used for the configuration and describing the status of ports on a PD device. Examples of PD devices are Ethernet phones.

The pethMainPseObjects MIB group defines the management objects for a managed main power source in a PSE device. Ethernet switches are one example of boxes that would support these objects.

The pethTrapsControlTable includes objects that control the transmission of traps by the agent to a management application.

[5.](#) Evolution of the Document, Limitations and Future Work

The IEEE 802.3af is at this stage work in progress. The scope of this document is to initiate standards work in the IETF in order to allow for the publication of a standard track document containing an SNMP

MIB simultaneously or close to the date of the publication of the IEEE revised standard. It is expected that changes may be brought to the IEEE proposal, and the Ethernet MIB Working Group will work in order to ensure consistency between the two standards proposals.

6. Changes Log

The following changes were introduced relative to the first proposal for a Power Ethernet MIB [[PWR-MIB](#)]

- a. pethPsePortTable has to index pethPsePortGroupIndex & pethPsePortIndex
- b. pethPsePortIndex INTEGER instead of InterfaceIndex
- c. Name change pethPsePortStatus insted of pethPsePortFaultError
- d. Name change pethPsePortStatusClear instead of pethPsePortFaultErrorClear
- e. DESCRIPTION update for pethPsePortPowerDetectionStatus test(3)

- f. DESCRIPTION update pethPsePortDetectionOperStatus off(2)
- g. Adding to pethPsePortStatus one more item both(4)
- h. Adding pethMainPseTable with a pethMainPseGroupIndex
- i. Deletting to objects pethMainPseMaxVoltage & pethMainPseMinVoltage
- j. Change SYNTAX of pethMainPseUsagePower form INTEGER to Gauge32
- k. Change SYNTAX of pethMainPseUsageCurrent form INTEGER to Gauge32
- l. Adding pethMainPseBackupActivated & pethMainPseBackupPresent
- m. Adding Traps Control Objects
- n. Adding Notifications Section (5 notifications)
- o. Adding pethTrapsControlGroup to Conformance Section

- p. Adding pethPsePortPowerClassifications to pethPsePortTable Class 1-5
- q. Adding pethPsePortPowerClassifications to pethPsePortGroup
- r. Change in pethPsePortStatus none(1) to ok(1)
- s. Change in DESCRIPTION of pethMainPseUsagePower from mW to Watt
- t. Change pethMainPseUsagePower to pethMainPseConsumptionPower
- u. Delete of pethMainPseUsageCurrent

7. Definitions

PETH-MIB DEFINITIONS ::= BEGIN

IMPORTS

```

MODULE-IDENTITY, OBJECT-TYPE, Integer32 , Gauge32, NOTIFICATION-TYPE
    FROM SNMPv2-SMI
dot3
    FROM EtherLike-MIB
TruthValue
    FROM SNMPv2-TC
MODULE-COMPLIANCE, OBJECT-GROUP , NOTIFICATION-GROUP
    FROM SNMPv2-CONF;

```

Ethernet MIB WG

Expires December 2001

[Page 5]

INTERNET DRAFT

Power Ethernet MIB

June 2001

```

powerEthernetMIB MODULE-IDENTITY
    LAST-UPDATED "200106200000Z" -- June 20, 2001
    ORGANIZATION "IETF Ethernet Interfaces and Hub MIB
        Working Group"
    CONTACT-INFO
    "

```

Chair: Dan Romascanu
 Avaya Inc.
 Tel: +972-3-645-8414
 Email: dromasca@avaya.com

Editor: Avi Berger
PowerDsine Inc.
Tel: 972-9-7755100 Ext 307
Fax: 972-9-7755120
E-mail: avib@PowerDsine.com

"

DESCRIPTION

"The MIB module for for managing Powered Devices (PD) or Power Source Equipment (PSE) working according to the IEEE 802.af Powere Ethernet (DTE Power via MDI) standard."

::= { dot3 20 }

pethObjects OBJECT IDENTIFIER ::= { powerEthernetMIB 1 }
pethNotifications OBJECT IDENTIFIER ::= { powerEthernetMIB 2 }
pethConformance OBJECT IDENTIFIER ::= { powerEthernetMIB 3 }

-- pethAgentControl MIB group defines the control objects for the power
-- Ethernet Agent

pethPsePortTable OBJECT-TYPE

SYNTAX SEQUENCE OF PethPsePortEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A table of objects that display and control the power characteristics power Ethernet ports on a Power Source Entity (PSE) device. This group will be implemented in managed power Ethernet switches and mid-span devices."

::= { pethObjects 1 }

pethPsePortEntry OBJECT-TYPE

SYNTAX PethPsePortEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A set of objects that display and control the power characteristics of a power Ethernet PSE port."

INDEX { pethPsePortGroupIndex , pethPsePortIndex }

::= { pethPsePortTable 1 }

```

PethPsePortEntry ::= SEQUENCE {
    pethPsePortGroupIndex
        INTEGER,
    pethPsePortIndex
        INTEGER,
    pethPsePortPowerEnable
        INTEGER,
    pethPsePortPowerIdPairsControl
        TruthValue,
    pethPsePortPowerIdPairs
        INTEGER,
    pethPsePortPowerDetectionStatus
        INTEGER,
    pethPsePortDetectionOperStatus
        INTEGER,
    pethPsePortPowerPriority
        INTEGER,
    pethPsePortDenyError
        INTEGER,
    pethPsePortStatus
        INTEGER,
    pethPsePortStatusClear
        INTEGER,
    pethPsePortType
        INTEGER,
    pethPsePortPowerClassifications
        INTEGER
}

```

pethPsePortGroupIndex OBJECT-TYPE

SYNTAX INTEGER (1..2147483647)

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This variable uniquely identifies the group containing the port to which power Ethernet PSE is connected. Group means a box in the stack, or a module in a rack. For non-modular devices the value 1 is recommended."

::= { pethPsePortEntry 1 }

pethPsePortIndex OBJECT-TYPE

SYNTAX INTEGER(1..2147483647)
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
 "This variable uniquely identifies the power Ethernet PSE
 port within group pethPseGroupIndex to which the
 power Ethernet PSE entry is connected."
 ::= { pethPsePortEntry 2 }

pethPsePortPowerEnable OBJECT-TYPE
SYNTAX INTEGER {
 auto(1),
 off(2)
}
MAX-ACCESS read-write
STATUS current
DESCRIPTION
 "Enables power supply on this port.
 Setting this object at a value auto(1) enables power
 and detection mechanism for this port.
 Setting this object at a value off(2) disables power
 and detection mechanism for this port."
 ::= { pethPsePortEntry 3 }

pethPsePortPowerIdPairsControl OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-only
STATUS current
DESCRIPTION
 "Describes the capability of controlling the power
 pairs functionality to switch pins for sourcing power."
 ::= { pethPsePortEntry 4 }

pethPsePortPowerIdPairs OBJECT-TYPE
SYNTAX INTEGER {
 signal(1),
 spare(2),
 both(3)
}
MAX-ACCESS read-write
STATUS current
DESCRIPTION
 "Describes or controls the pairs in use. If the value of
 pethPsePortPowerIdpairsControl is true, thisobject is
 writable.
 A value of signal(1) menas that the signal pairs
 only are in use."

INTERNET DRAFT

Power Ethernet MIB

June 2001

A value of spare(2) means that the spare pairs only are in use.

A value of both(3) means that both the signal and the spare pairs are in use."

::= { pethPsePortEntry 5 }

pethPsePortPowerDetectionStatus OBJECT-TYPE

```
SYNTAX INTEGER {
    auto(1),
    off(2),
    test(3)
}
```

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Controls the power detection mechanism of the port.

Setting the value auto(1) enables the power detection mechanism of the port.

Setting the value off(2) disables the power detection mechanism of the port.

Setting the value test(3) puts the port in a test mode forcing continuous discovery without applying power regardless of whether PD detected."

::= { pethPsePortEntry 6 }

pethPsePortDetectionOperStatus OBJECT-TYPE

```
SYNTAX INTEGER {
    deliveringPower(1),
    off(2),
    searching(3),
    fault(4)
}
```

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Describes the operational status of the port detection.

A value of deliveringPower(1) indicates that the port executed the detection algorithm, found a PD connection and is currently delivering power.

A value of off(2) indicates that the port did not find a PD connection and is not in searching mode.

A value of searching(3) indicates that the detection

algorithm is in work, and did not detect a valid PD. No power is currently provided.
A value of fault(4) indicates that a fault was detected on the port . "
 ::= { pethPsePortEntry 7 }

pethPsePortPowerPriority OBJECT-TYPE
SYNTAX INTEGER {
 critical(1),
 high(2),
 low(3)
}
MAX-ACCESS read-write
STATUS current
DESCRIPTION
 "This object controls the priority of the port from the point of view of a power management algorithm. The priority that is set by this variable could be used by a control mechanism that prevents over current situations by disconnecting first ports with lower power priority. Ports that connect devices critical to the operation of the network - like the E911 telephones ports - should be set to higher priority."
 ::= { pethPsePortEntry 8 }

pethPsePortDenyError OBJECT-TYPE
SYNTAX INTEGER {
 other(1),
 lowPriority(2)
}
MAX-ACCESS read-only
STATUS current
DESCRIPTION
 "This object describes an error resulted from an action of the power management mechanism. The value lowPriority(2) indicates that the port was disabled by the power management system, in order to keep active higher priority ports."
 ::= { pethPsePortEntry 9 }

pethPsePortStatus OBJECT-TYPE
SYNTAX INTEGER {
 ok(1),

```

        underCurrent(2),
        overCurrent(3),
        both(4)
    }
MAX-ACCESS read-only
STATUS current
DESCRIPTION
    "Describes a current port status related to the power generation
    The value underCurrent(2) indicates that the port current
    is below the minimal value.
    The value overCurrent(3) indicates that the port current
    exceeds the maximal value.
    The value both(4) indicates that both underCurrent and overCurrent

```

```

        status happend from the lase Clear."
 ::= { pethPsePortEntry 10 }

    pethPsePortStatusClear OBJECT-TYPE
SYNTAX INTEGER {
        clear(1),
        off(2)
}
MAX-ACCESS read-write
STATUS current
DESCRIPTION
    "Setting the value of this object to clear(1) clears the value
    of the pethPsePortStatus to none(1)."
```

```

 ::= { pethPsePortEntry 11 }

    pethPsePortType OBJECT-TYPE
SYNTAX INTEGER {
        other(1),
        telephone(2),
        webcam(3),
        wireless(4)
}
MAX-ACCESS read-write
STATUS current
DESCRIPTION
    "A manager will set the value of this variable to a value
    that indicates the type of the device that is connected
    to theport. This value can be the result of the mapping

```

the address of the station connected to the port and of
the value of the pethPdPortType of the respective PD port."
 ::= { pethPsePortEntry 12 }

pethPsePortPowerClassifications OBJECT-TYPE

```
SYNTAX INTEGER {  
    class0(1),  
    class1(2),  
    class2(3),  
    class3(4),  
    class4(5),  
    class5(6)  
}
```

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Classification is a way to tag different terminals on the power over LAN network according to their power consumption. Devices such as IP telephones, WLAN access points and others, will be classified according to their power requirements.

Class	Usage	Maximum Power
0	Default	0.5 - 15.0 W
1	Optional	0.5 - 4.0 W
2	Optional	4.0 - 7.0 W
3	Optional	7.0 - 15.0 W
4	Optional	Future Use
5	Optional	Future Use

"

::= { pethPsePortEntry 13 }

-- PD Port table

pethPdPortTable OBJECT-TYPE

SYNTAX SEQUENCE OF PethPdPortEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A table of objects that display and control the power characteristics power Ethernet ports on a Powered Device(PD) device. This group will be implemented in managed powered and mid-span devices."

```

 ::= { pethObjects 2 }

pethPdPortEntry OBJECT-TYPE
    SYNTAX      PethPdPortEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A set of objects that display and control the power
        characteristics of a Powered Device port."
    INDEX       { pethPdPortIndex }
    ::= { pethPdPortTable 1 }

PethPdPortEntry ::= SEQUENCE {
    pethPdPortIndex
        INTEGER,
    pethPdPortPowerPairs
        INTEGER,
    pethPdPortDetectionOperStatus
        INTEGER,
    pethPdPortType
        INTEGER
}

pethPdPortIndex OBJECT-TYPE
    SYNTAX      INTEGER (0..65535)
    MAX-ACCESS  not-accessible
    STATUS      current

```

DESCRIPTION

"An index value that uniquely identifies an interface to a PD device. The interface identified by a particular value of this index is the same interface as identified by the same value of ifIndex. The mapping between the ifIndex values and the numbering of the port on the device is an implementation issue."

```
 ::= { pethPdPortEntry 1 }
```

```

pethPdPortPowerPairs OBJECT-TYPE
    SYNTAX INTEGER {
        signal(1),

```

```

        spare(2),
        both(3)
    }
MAX-ACCESS read-only
STATUS current
DESCRIPTION
    "Describes the pairs in use.
    A value of signal(1) means that the signal pairs
    only are in use.
    A value of spare(2) means that the spare pairs
    only are in use.
    A value of both(3) means that both the signal
    and the spare pairs are in use."
 ::= { pethPdPortEntry 2 }

```

```

    pethPdPortDetectionOperStatus OBJECT-TYPE

```

```

SYNTAX INTEGER {
    off(1),
    receivingPower(2)
}

```

```

MAX-ACCESS read-only

```

```

STATUS current

```

```

DESCRIPTION

```

```

    "Describes the operational status of the port detection.

```

```

    The value off(1) means that the port does not receive
    power and the detection algorithm might still be operating.

```

```

    The value receivingPower(2) means that the port is
    receiving power. "

```

```

 ::= { pethPdPortEntry 3 }

```

```

    pethPdPortType OBJECT-TYPE

```

```

SYNTAX INTEGER {
    other(1),
    telephone(2),
}

```

```

        webcam(3),
        wireless(4)
    }

```

```

MAX-ACCESS read-only

```

```

STATUS current

```

```

DESCRIPTION

```

```

    "The type of the device. A management application may read

```

```

        the value of this variable and use it for setting the
        corresponding value of pethPsePortType of the port that
        connects the device."
 ::= { pethPdPortEntry 4 }

-- Main PSE Objects

pethMainPseObjects          OBJECT IDENTIFIER ::= { pethObjects 3 }

pethMainPseTable OBJECT-TYPE
    SYNTAX          SEQUENCE OF PethMainPseEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "A table of objects that display and control the Main power
        on a PSE device."
    ::= { pethMainPseObjects 1 }

pethMainPseEntry OBJECT-TYPE
    SYNTAX          PethMainPseEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "A set of objects that display and control the Main power
        of a PSE."
    INDEX          { pethMainPseGroupIndex }
    ::= { pethMainPseTable 1 }

PethMainPseEntry ::= SEQUENCE {
    pethMainPseGroupIndex
        INTEGER,
    pethMainPsePower
        Integer32,
    pethMainPseOperStatus
        INTEGER,
    pethMainPseConsumptionPower
        Gauge32,
    pethMainPseBackupPresent
        INTEGER,
    pethMainPseBackupActivated
        INTEGER,

```

```

    pethMainPseUsageThreshold
        INTEGER
}
pethMainPseGroupIndex OBJECT-TYPE
    SYNTAX      INTEGER (0..65535)
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "This variable uniquely identifies the group to which
        power Ethernet PSE is connected. Group means box in the stack,
        module in a rack. It is recommended that the value 1 be
        used for non-modular devices "
    ::= { pethMainPseEntry 1 }

pethMainPsePower OBJECT-TYPE
    SYNTAX      Integer32 (0..65535)
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "The nominal power of the PSE expressed in Watts."
    ::= { pethMainPseEntry 2 }

pethMainPseOperStatus OBJECT-TYPE
    SYNTAX INTEGER {
        on(1),
        off(2),
        faulty(3)
    }
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The operational status of the main PSE."
    ::= { pethMainPseEntry 3 }

pethMainPseConsumptionPower OBJECT-TYPE
    SYNTAX      Gauge32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Measured usage power expressed in Watts."
    ::= { pethMainPseEntry 4 }

pethMainPseBackupPresent OBJECT-TYPE
    SYNTAX INTEGER {
        present(1),
        notPresent(2),

```

INTERNET DRAFT

Power Ethernet MIB

June 2001

```

        faulty(3)
    }
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "This value reflects the presence and status
        of a backup PSE ."
    ::= { pethMainPseEntry 5 }

pethMainPseBackupActivated OBJECT-TYPE
    SYNTAX          INTEGER {
                        activated(1),
                        notActivated(2)
                    }
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "This variable reflects the activation status
        of the backup PSE"
    ::= { pethMainPseEntry 6 }

pethMainPseUsageThreshold OBJECT-TYPE
    SYNTAX          INTEGER (1..99)
    MAX-ACCESS      read-write
    STATUS          current
    DESCRIPTION
        "The usage threshold expressed in percents for
        comparing the measured power and initiating
        an alarm if the threshold is exceeded."
    ::= { pethMainPseEntry 7 }
```

-- Traps Control Objects

```
pethTrapsControl          OBJECT IDENTIFIER ::= { pethObjects 4 }
```

```
pethTrapsControlTable OBJECT-TYPE
    SYNTAX          SEQUENCE OF PethTrapsControlEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "A table of objects that display and control the Main power
        on a PSE device."
```

```
::= { pethTrapsControl 1 }
```

```
pethTrapsControlEntry OBJECT-TYPE  
SYNTAX      PethTrapsControlEntry  
MAX-ACCESS  not-accessible
```

Ethernet MIB WG

Expires December 2001

[Page 16]

INTERNET DRAFT

Power Ethernet MIB

June 2001

```
STATUS      current
```

```
DESCRIPTION
```

```
"A set of objects that control the Trap events."
```

```
INDEX      { pethTrapsControlGroupIndex }
```

```
::= { pethTrapsControlTable 1 }
```

```
PethTrapsControlEntry ::= SEQUENCE {  
    pethTrapsControlGroupIndex  
        INTEGER,  
    pethTrapsControlEnable  
        INTEGER  
}
```

```
pethTrapsControlGroupIndex OBJECT-TYPE
```

```
SYNTAX      INTEGER (0..65535)
```

```
MAX-ACCESS  not-accessible
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"This variable uniquely identifies the group. Group means  
box in the stack, or module in a rack. It is recommended  
that the value 1 is used for non-modular devices."
```

```
::= { pethTrapsControlEntry 1 }
```

```
pethTrapsControlEnable OBJECT-TYPE
```

```
SYNTAX      INTEGER
```

```
{
```

```
    enable(1),
```

```
    disable(2)
```

```
}
```

```
MAX-ACCESS  read-write
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"Enable Traps from Agent"
```

```
::= { pethTrapsControlEntry 2 }
```

--
-- Notifications Section
--
--

```
pethPsePortOnOffTrap NOTIFICATION-TYPE
  OBJECTS      { pethPsePortGroupIndex,pethPsePortIndex,pethPsePortDet
  STATUS       current
  DESCRIPTION  " This trap indicate if Pse Port is delivering or not
  ::= { pethNotifications 1 }
```

Ethernet MIB WG

Expires December 2001

[Page 17]

INTERNET DRAFT

Power Ethernet MIB

June 2001

```
pethPsePortDenyTrap NOTIFICATION-TYPE
  OBJECTS      { pethPsePortGroupIndex,pethPsePortIndex,pethPsePortDen
  STATUS       current
  DESCRIPTION  " This trap indicate Port Deny service."
  ::= { pethNotifications 2 }
```

```
pethPsePortStatusTrap NOTIFICATION-TYPE
  OBJECTS      { pethPsePortGroupIndex,pethPsePortIndex,pethPsePortSta
  STATUS       current
  DESCRIPTION  " This trap indicate Port Change Status."
  ::= { pethNotifications 3 }
```

```
pethMainPseBackUpActivatedTrap NOTIFICATION-TYPE
  OBJECTS      { pethPsePortGroupIndex,pethMainPseBackupActivated }
  STATUS       current
  DESCRIPTION  " This trap indicates that BackUp PSE is Activated ."
  ::= { pethNotifications 4 }
```

```
pethMainPowerUsageTrap NOTIFICATION-TYPE
  OBJECTS      { pethPsePortGroupIndex }
  STATUS       current
  DESCRIPTION  " This trap indicate PSE Threshold usage indication ."
  ::= { pethNotifications 5 }
```

```

--
-- Conformance Section
--
pethCompliances OBJECT IDENTIFIER ::= { pethConformance 1 }
pethGroups      OBJECT IDENTIFIER ::= { pethConformance 2 }

pethCompliance MODULE-COMPLIANCE
    STATUS current
    DESCRIPTION
        "Describes the requirements for conformance to the
        Power Ethernet MIB."
    MODULE -- this module
        GROUP pethPsePortGroup
        DESCRIPTION
            "The pethPsePortGroup is mandatory for systems which
            implement PSE ports."
        GROUP pethPdPortGroup
        DESCRIPTION

```

```

        "The pethPdPortGroup is mandatory for systems which
        implement PD Ports."
    GROUP pethMainPseGroup
    DESCRIPTION
        "The pethMainPseGroup is mandatory for systems which
        implement main power supply within a PSE Device."
    GROUP pethTrapsControlGroup
    DESCRIPTION
        "The pethTrapsControlGroup is mandatory for systems which
        implement PSE ports."
    ::= { pethCompliances 1 }

pethPsePortGroup OBJECT-GROUP
    OBJECTS {
        pethPsePortPowerEnable,
        pethPsePortPowerIdPairsControl,
        pethPsePortPowerIdPairs,
        pethPsePortPowerDetectionStatus,
        pethPsePortDetectionOperStatus,
        pethPsePortPowerPriority,
        pethPsePortDenyError,
        pethPsePortStatus,

```

```

        pethPsePortStatusClear,
        pethPsePortType,
        pethPsePortPowerClassifications
    }
    STATUS current
    DESCRIPTION
        "PSE Port Objects."
    ::= { pethGroups 1 }

pethPdPortGroup OBJECT-GROUP
    OBJECTS {
        pethPdPortPowerPairs,
        pethPdPortDetectionOperStatus,
        pethPdPortType
    }
    STATUS current
    DESCRIPTION
        "PD Port Objects."
    ::= { pethGroups 2 }

pethMainPseGroup OBJECT-GROUP
    OBJECTS {
        pethMainPsePower,
        pethMainPseOperStatus,
        pethMainPseConsumptionPower,
        pethMainPseBackupPresent,

```

```

        pethMainPseBackupActivated,
        pethMainPseUsageThreshold
    }
    STATUS current
    DESCRIPTION
        "Main PSE Objects. "
    ::= { pethGroups 3 }

pethTrapsControlGroup OBJECT-GROUP
    OBJECTS {
        pethTrapsControlEnable
    }
    STATUS current
    DESCRIPTION
        "Trap Control Objects. "

```

```
::= { pethGroups 4 }  
END
```

8. References

- [RFC2571] Harrington, D., Presuhn, R., and B. Wijnen, "An Architecture for Describing SNMP Management Frameworks", [RFC 2571](#), April 1999.
- [RFC1155] Rose, M., and K. McCloghrie, "Structure and Identification of Management Information for TCP/IP-based Internets", STD 16, [RFC 1155](#), May 1990.
- [RFC1212] Rose, M., and K. McCloghrie, "Concise MIB Definitions", STD 16, [RFC 1212](#), March 1991.
- [RFC1215] M. Rose, "A Convention for Defining Traps for use with the SNMP", [RFC 1215](#), March 1991.
- [RFC2578] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M., and S. Waldbusser, "Structure of Management Information Version 2 (SMIv2)", STD 58, [RFC 2578](#), April 1999.
- [RFC2579] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M., and S. Waldbusser, "Textual Conventions for SMIv2", STD 58, [RFC 2579](#), April 1999.
- [RFC2580] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M., and S. Waldbusser, "Conformance Statements for

SMIv2", STD 58, [RFC 2580](#), April 1999.

- [RFC1157] Case, J., Fedor, M., Schoffstall, M., and J. Davin, "Simple Network Management Protocol", STD 15, [RFC 1157](#), May 1990.
- [RFC1901] Case, J., McCloghrie, K., Rose, M., and S. Waldbusser, "Introduction to Community-based SNMPv2", [RFC 1901](#), January 1996.

- [RFC1906] Case, J., McCloghrie, K., Rose, M., and S. Waldbusser, "Transport Mappings for Version 2 of the Simple Network Management Protocol (SNMPv2)", [RFC 1906](#), January 1996.
- [RFC2572] Case, J., Harrington D., Presuhn R., and B. Wijnen, "Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)", [RFC 2572](#), April 1999.
- [RFC2574] Blumenthal, U., and B. Wijnen, "User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)", [RFC 2574](#), April 1999.
- [RFC1905] Case, J., McCloghrie, K., Rose, M., and S. Waldbusser, "Protocol Operations for Version 2 of the Simple Network Management Protocol (SNMPv2)", [RFC 1905](#), January 1996.
- [RFC2573] Levi, D., Meyer, P., and B. Stewart, "SNMPv3 Applications", [RFC 2573](#), April 1999.
- [RFC2575] Wijnen, B., Presuhn, R., and K. McCloghrie, "View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)", [RFC 2575](#), April 1999.
- [RFC2570] Case, J., Mundy, R., Partain, D., and B. Stewart, "Introduction to Version 3 of the Internet-standard Network Management Framework", [RFC 2570](#), April 1999.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC2665] Flick, J., and J. Johnson, "Definitions of Managed Objects for the Ethernet-like Interface Types", [RFC 2665](#), August 1999.
- [IEEE-802.3af] IEEE 802.3af Working Group, "Data Terminal Equipment (DTE) Power via Media Dependent Interface (MDI)", Draft D1.1, January 2001.
- [PWR-MIB] Romascanu, D., " Power Ethernet (DTE Power via MDI) MIB", Internet-Draft, [draft-romascanu-hubmib-power-ethernet-mib-00.txt](#)

9. Intellectual Property

The IETF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on the IETF's procedures with respect to rights in standards-track and standards-related documentation can be found in [BCP-11](#). Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementors or users of this specification can be obtained from the IETF Secretariat.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this standard. Please address the information to the IETF Executive Director.

[10.](#) Security Considerations

There are a number of management objects defined in this MIB that have 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.

There are a number of managed objects in this MIB that may contain sensitive information. These are:

It is thus important to control even GET access to these objects and possibly to even encrypt the values of these object when sending them over the network via SNMP. Not all versions of SNMP provide features for such a secure environment.

SNMPv1 by itself is not a secure environment. 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.

It is RECOMMENDED that the implementers consider the security features as provided by the SNMPv3 framework. Specifically, the use of the User-based Security Model [[RFC2274](#)] and the View-based Access Control Model [[RFC2275](#)] is RECOMMENDED.

It is then a customer/user responsibility to ensure that the SNMP entity giving access to an instance of this MIB, 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.

11. Authors Addresses

Avi Berger
PowerDsine Inc.
1, Hanagar St., P.O. Box 7220
Hod Hasharon 45421,
Israel
Tel: 972-9-7755100 Ext 307
Fax: 972-9-7755120
E-mail: avib@PowerDsine.com

Dan Romascanu
Avaya Inc.
Atidim Technology Park, Bldg. #3
Tel Aviv, 61131
Israel
Tel: +972-3-645-8414
Email: dromasca@avaya.com

A. Full Copyright Statement

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

English.

Ethernet MIB WG

Expires December 2001

[Page 23]

INTERNET DRAFT

Power Ethernet MIB

June 2001

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

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

