

Network Working Group
Internet-Draft
Intended status: Informational
Expires: July 8, 2011

M. Azinger
Frontier Communications
Corporation
L. Vegoda
ICANN
January 4, 2011

Issues Associated with Designating Additional Private IPv4 Address Space
draft-azinger-additional-private-ipv4-space-issues-05

Abstract

When a private network or internetwork grows very large it is sometimes not possible to address all interfaces using private IPv4 address space because there are not enough addresses. This document describes the problems faced by those networks, the available options and the issues involved in assigning a new block of private IPv4 address space.

While this informational document does not make a recommendation for action, it documents the issues surrounding the various options that have been considered.

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79.

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/ietf/lid-abstracts.txt>.

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

This Internet-Draft will expire on July 8, 2011.

Copyright Notice

Copyright (c) 2011 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the BSD License.

Table of Contents

1.	Introduction	3
2.	Large Networks	3
3.	Non-Unique Addresses	3
3.1.	Subscriber Use Network Address Translation	3
3.2.	Carrier Grade Network Address Translation	4
4.	Available Options	4
4.1.	IPv6 Options	4
4.1.1.	Unique Globally Scoped IPv6 Unicast Addresses	4
4.1.2.	Unique Local IPv6 Unicast Addresses	4
4.2.	IPv4 Options	5
4.2.1.	Address Transfers or Leases From Organizations with Available Address Space	5
4.2.2.	Using Unannounced Address Space Allocated to Another Organization	5
4.2.3.	Unique IPv4 Space Registered by an RIR	6
5.	Options and Consequences for Defining New Private Use Space	6
5.1.	Redefining Existing Unicast Space as Private Address Space	6
5.2.	Unique IPv4 Space Shared by a Group of Operators	7
5.3.	Potential Consequences of Not Redefining Existing Unicast Space as Private Address Space	8
5.4.	Redefining Future Use Space as Unicast Address Space	8
6.	Security Considerations	8
7.	IANA Considerations	8
8.	References	9
8.1.	Normative References	9
8.2.	Informative References	9
Appendix A.	Acknowledgments	11
Authors' Addresses	11

1. Introduction

[RFC1918] sets aside three blocks of IPv4 address space for use in private networks: 192.168.0.0/16, 172.16.0.0/12 and 10.0.0.0/8. These blocks can be used simultaneously in multiple, separately managed networks without registration or coordination with IANA or any Internet registry. Very large networks can find that they need to number more device interfaces than there are available addresses in these three ranges. It has occasionally been suggested that additional private IPv4 address space should be reserved for use by these networks. Although such an action might address some of the needs for these very large network operators it is not without consequences, particularly as we near the date when the IANA free pool will be fully allocated.

2. Large Networks

The main categories of very large networks using private address space are: cable operators, wireless (cell phone) operators, private internets and VPN service providers. In the case of the first two categories, the complete address space reserved in [RFC1918] tends to be used by a single organization. In the case of private internets and VPN service providers there are multiple independently managed and operated networks and the difficulty is in avoiding address clashes.

3. Non-Unique Addresses

3.1. Subscriber Use Network Address Translation

The address space set aside in [RFC1918] is a finite resource which can be used to provide limited Internet access via Network Address Translation (NAT). A discussion of the advantages and disadvantages of NATs is outside the scope of this document but an analysis of the advantages, disadvantages and architectural implications can be found in [RFC2993]. Nonetheless, it must be acknowledged that NAT is adequate in some situations and not in others. For instance, it might be technically feasible to use NAT or even multiple layers of NAT within the networks operated by residential users or corporations where only limited Internet access is required. A more detailed analysis can be found in [RFC3022]. Where true peer to peer communication is needed or where services or applications do not work properly behind NAT, globally unique address space is required. In other cases, NAT traversal techniques facilitate peer-to-peer like communication for devices behind NATs.

In many cases it is possible to use multiple layers of NAT to re-use parts of the address space defined in [RFC1918]. It is not always possible to rely on CPE devices using any particular range, however. In some cases this means that unorthodox workarounds including assigning CPE devices unallocated address space or address space allocated to other network operators are feasible. In other cases, organizations choose to operate multiple separate routing domains to allow them to re-use the same private address ranges in multiple contexts. One consequence of this is the added complexity involved in identifying which system is referred to when an IP address is identified in a log or management systems.

3.2. Carrier Grade Network Address Translation

Another option is to share one address across multiple interfaces and in some cases, subscribers. This model breaks the classical model used for logging address assignments and creates significant risks and additional burdens, as described in [CLAYTON] and more fully discussed in [FORD] and is documented in [DS-LITE].

4. Available Options

When a network operator has exhausted the private address space set aside in [RFC1918] but needs to continue operating a single routing domain a number of options are available. These include:

4.1. IPv6 Options

4.1.1. Unique Globally Scoped IPv6 Unicast Addresses

Using unique, globally scoped IPv6 unicast addresses is the best permanent solution as it removes any concerns about address scarcity within the next few decades. Implementing IPv6 is a major endeavor for service providers with millions of consumer customers and is likely to take considerable effort and time. In some cases implementing a new network protocol on a very large network takes more time than is available, based on network growth and the proportion of private space that has already been used. In these cases, there is a call for additional private address space that can be shared by all network operators. [DAVIES] makes one such case.

4.1.2. Unique Local IPv6 Unicast Addresses

Using the unique, local IPv6 unicast addresses defined in [RFC4193] is another approach and does not require coordination with an Internet registry. Although the addresses defined in [RFC4193] are probabilistically unique, network operators on private internets and

those providing VPN services might not want to use them because there is a very low probability of non-unique locally assigned global IDs being generated by the algorithm. Also, in the case of private internets, it can be very challenging to coordinate the introduction of a new network protocol to support the internet's continued growth.

4.2. IPv4 Options

4.2.1. Address Transfers or Leases From Organizations with Available Address Space

The Regional Internet Registry (RIR) communities have recently been developing policies to allow organizations with available address space to transfer such designated space to other organizations [RIR-POLICY]. In other cases, leases might be arranged. This approach is only viable for operators of very large networks if enough address space is made available for transfer or lease and if the very large networks are able to pay the costs of these transfers. It is not possible to know how much address space will become available in this way, when it will be available and how much it will cost. However, it is unlikely to become available in large contiguous blocks and this would add to the network management burden for the operator as a significant number of small prefixes would inflate the size of the operators routing table at a time when it is also adding an IPv6 routing table. These reasons will make address transfers a less attractive proposition to many large network operators. Leases might not be attractive to some organizations if both parties cannot agree a suitable length of time. Also, the lessor might worry about its own unanticipated needs for additional IPv4 address space.

4.2.2. Using Unannounced Address Space Allocated to Another Organization

Some network operators have considered using IP address space which is allocated to another organization but is not publicly visible in BGP routing tables. This option is very strongly discouraged as the fact that an address block is not visible from one view does not mean that it is not visible from another. Furthermore, address usage tends to leak beyond private network borders in e-mail headers, DNS queries, traceroute output and other ways. The ambiguity this causes is problematic for multiple organizations. This issue is discussed in [RFC3879], section 2.3.

It is also possible that the registrant of the address block might want to increase its visibility to other networks in the future, causing problems for anyone using it unofficially. In some cases there might also be legal risks involved in using address space

officially allocated to another organization.

Where this has happened in the past it has caused operational problems [FASTWEB].

4.2.3. Unique IPv4 Space Registered by an RIR

RIRs policies allow network operators to receive unique IP addresses for use on internal networks. Further, network operators are not required to have already exhausted the private address space set aside in [RFC1918]. Nonetheless, network operators are naturally disinclined to request unique IPv4 addresses for the private areas of their networks as using addresses in this way means they are not available for use by new Internet user connections.

It is likely to become more difficult for network operators to obtain large blocks of unique address space as we approach the point where all IPv4 unicast /8s have been allocated. Several RIRs already have policies how to allocate from their last /8 [RIR-POLICY-FINAL-8] and there have been policy discussions that would reduce the maximum allocation size available to network operators [MAX-ALLOC] or would reduce the period of need for which the RIR can allocate [SHORTER-PERIODS].

5. Options and Consequences for Defining New Private Use Space

5.1. Redefining Existing Unicast Space as Private Address Space

It is possible to re-designate a portion of the current global unicast IPv4 address space as private unicast address space. Doing this could benefit a number of operators of large network for the short period before they complete their IPv6 roll-out. However, this benefit incurs a cost by reducing the pool of global unicast addresses available to users in general.

When discussing re-designating a portion of the current global unicast IPv4 address space as private unicast address space it is important to consider how much space would be used and for how long it would be sufficient. Not all of the large networks making full use of the space defined in [RFC1918] would have their needs met with a single /8. In 2005, [HAIN] suggested reserving three /8s for this purpose while in 2009 [DAVIES] suggested a single /10 would be sufficient. There does not seem to be a consensus for a particular prefix length nor an agreed basis for deciding what is sufficient. The problem is exacerbated by the continually changing needs of ever expanding networks.

A further consideration is which of the currently unallocated IPv4 unicast /8 blocks should be used for this purpose. Using address space which is known to be used unofficially is tempting. For instance, 1.0.0.0/8, which was unallocated until January 2010, was proposed in [HAIN] and is known to be used by a number of different users. These include networks making use of HIP LSIs [RFC4423], [WIANA], [anoNet] and others. There is anecdotal [VEGODA] and research [WESSELS] evidence to suggest that several other IPv4 /8s are used in this fashion. Also there have been discussions [NANOG] about some sections of these /8's being carved out and filtered therefore unofficially enabling the use of these sections for private use.

Although new IPv4 /8s are allocated approximately once a month, they are not easy to bring into use because network operators are slow to change their filter configurations. This is despite long-running awareness campaigns [CYMRU], [LEWIS] and active work [ripe-351] to notify people whose filters are not changed in a timely fashion. Updating code that recognises private address space in deployed software and infrastructure systems is likely to be far more difficult as many systems have these ranges hard-coded and cannot be quickly changed with a new configuration file.

Another consideration when redefining existing unicast space as private address space is that no single class of user can expect the space to stay unique to them. This means that an ISP using a new private address range cannot expect its customers not to already be using that address range within their own networks.

5.2. Unique IPv4 Space Shared by a Group of Operators

Where a group of networks find themselves in a position where they each need a large amount of IPv4 address space from an RIR in addition to that defined in [RFC1918] they might cooperatively agree to all use the same address space to number their networks. The clear benefit to this approach is that it significantly reduces the potential demand on the pool of unallocated IPv4 address space. However, the issues discussed in 4.4 could also be of concern here, particularly the possibility that one operator might decide to use the address space to number customer connections, rather than private infrastructure.

Nonetheless, this approach has the potential to create an unofficial new private address range without proper scrutiny.

5.3. Potential Consequences of Not Redefining Existing Unicast Space as Private Address Space

If additional private address space is not defined and the large network operators affected by this problem are not able to solve their problems with IPv6 address space or by segmenting their networks into multiple routing domains, those networks will need unique IPv4 addresses. It is possible and even likely that a single network could consume a whole IPv4 /8 in a year. At the time of writing there are just 24 unallocated IPv4 /8s, so it would not take many such requests to make a major dent in the available IPv4 address space. [POTAROO] provides an analysis of IPv4 address consumption and projects the date on which the IANA and RIR pools will be fully allocated.

5.4. Redefining Future Use Space as Unicast Address Space

There have also been proposals to re-designate the former Class E space (240.0.0.0/4) as unicast address space. [WILSON] suggests that it should be privately scoped while [FULLER] does not propose a scope. Both proposals note that existing deployed equipment may not be able to use addresses from 240.0.0.0/4. Potential users would need to be sure of the status of the equipment on their network and the networks with which they intend to communicate.

It is not immediately clear how useful 240.0.0.0/4 could be in practice. While [FULLER] documents the status of several popular desktop and server operating systems, the status of the most widely deployed routers and switches is less clear and it is possible that 240.0.0.0/4 might only be useful in very large, new green field deployments where full control of all deployed systems is available. However, in such cases it might well be easier to deploy an IPv6 network.

6. Security Considerations

This document has no security implications.

7. IANA Considerations

This document makes no request of IANA.

8. References

8.1. Normative References

- [RFC1918] Rekhter, Y., Moskowitz, R., Karrenberg, D., Groot, G., and E. Lear, "Address Allocation for Private Internets", BCP 5, RFC 1918, February 1996.
- [RFC2993] Hain, T., "Architectural Implications of NAT", RFC 2993, November 2000.
- [RFC3022] Srisuresh, P. and K. Egevang, "Traditional IP Network Address Translator (Traditional NAT)", RFC 3022, January 2001.
- [RFC4193] Hinden, R. and B. Haberman, "Unique Local IPv6 Unicast Addresses", RFC 4193, October 2005.

8.2. Informative References

- [RFC3879] Huitema, C. and B. Carpenter, "Deprecating Site Local Addresses", RFC 3879, September 2004.
- [RFC4423] Moskowitz, R. and P. Nikander, "Host Identity Protocol (HIP) Architecture", RFC 4423, May 2006.
- [anoNet] anoNet, "anoNet: Cooperative Chaos", <<http://www.anonet.org/>>.
- [CLAYTON] Clayton, R., "Practical mobile Internet access traceability", January 2010, <<http://www.lightbluetouchpaper.org/2010/01/13/practical-mobile-internet-access-traceability/>>.
- [CYMRU] Greene, B., "The Bogon Reference", <<http://www.team-cymru.org/Services/Bogons/>>.
- [DAVIES] Davies, G. and C. Liljenstolpe, "Work in Progress: Transitional non-conflicting reusable IPv4 address block", November 2009, <<http://tools.ietf.org/html/draft-davies-reusable-ipv4-address-block-00>>.
- [DS-LITE] Durand, A., Droms, R., Woodyatt, J., and Y. Lee, "Work in Progress: Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion", August 2010, <<http://tools.ietf.org/html/draft-ietf-softwire-dual-stack-lite-06>>.
- [FASTWEB] Aina, A., "41/8 announcement", May 2006, <<http://www.afnog.org/archives/2006-May/002117.html>>.

- [FORD] Ford, M., Boucadair, M., Durand, A., Levis, P., and P. Roberts, "Work in Progress: Issues with IP Address Sharing", March 2010, <<http://tools.ietf.org/html/draft-ford-shared-addressing-issues-02>>.
- [FULLER] Fuller, V., Lear, E., and D. Meyer, "Work in Progress: Reclassifying 240/4 as usable unicast address space", March 2008, <<http://tools.ietf.org/html/draft-fuller-240space-02>>.
- [HAIN] Hain, T., "Work in Progress: Expanded Address Allocation for Private Internets", January 2005, <<http://tools.ietf.org/html/draft-hain-1918bis-01>>.
- [LEWIS] Lewis, J., "This system has been setup for testing purposes for 69/8 address space", March 2003, <<http://69box.atlantic.net/>>.
- [MAX-ALLOC] Spenceley, J. and J. Martin, "prop-070: Maximum IPv4 allocation size", January 2009, <<http://www.apnic.net/policy/proposals/prop-070>>.
- [NANOG] Dickson, B., "1/8 and 27/8 allocated to APNIC", January 2010, <<http://mailman.nanog.org/pipermail/nanog/2010-January/017451.html>>.
- [POTAROO] Huston, G., "IPv4 Address Report", <<http://www.potaroo.net/tools/ipv4/index.html>>.
- [ripe-351] Karrenberg, D., "De-Bogonising New Address Blocks", October 2005, <<http://www.ripe.net/ripe/docs/ripe-351.html>>.
- [RIR-POLICY] Number Resource Organization, "RIR Comparative Policy Overview, October 2009, Section 1.3.2 Transfer of Custodianship", <<http://www.nro.net/documents/comp-pol-200910.html#1-3-2>>.
- [RIR-POLICY-FINAL-8] Number Resource Organization, "RIR Comparative Policy Overview, October 2009, 2.6. Use of Final Unallocated IPv4 Address Space", October 2009, <<http://www.nro.net/documents/comp-pol-200910.html#2-6>>.
- [SHORTER-PERIODS]

Karrenberg, D., O'Reilly, N., Titley, N., and R. Bush,
"RIPE Policy Proposal 2009-03", April 2009,
<[http://www.ripe.net/ripe/policies/proposals/
2009-03.html](http://www.ripe.net/ripe/policies/proposals/2009-03.html)>.

[VEGODA] Vegoda, L., "Awkward /8 Assignments", September 2007, <[http://www.cisco.com/web/about/ac123/ac147/archived_issues/
ipj_10-3/103_awkward.html](http://www.cisco.com/web/about/ac123/ac147/archived_issues/ipj_10-3/103_awkward.html)>.

[WESSELS] Wessels, D., "Searching for Evidence of Unallocated
Address Space Usage in DITL 2008 Data", June 2008, <[https://www.dns-oarc.net/files/dnsops-2008/
Wessels-Unused-space.pdf](https://www.dns-oarc.net/files/dnsops-2008/Wessels-Unused-space.pdf)>.

[WIANA] WIANA, "The Wireless Internet Assigned Numbers Authority",
<<http://www.wiana.org/>>.

[WILSON] Wilson, P., Michaelson, G., and G. Huston, "Work in
Progress: Redesignation of 240/4 from "Future Use" to
"Private Use" ",
<<http://tools.ietf.org/html/draft-wilson-class-e-02>>.

Appendix A. Acknowledgments

The authors would also like to thank Ron Bonica, Michelle Cotton, Lee Howard and Barbara Roseman for their assistance in early discussions of this document and to Maria Blackmore, Alex Bligh, Mat Ford, Thomas Narten, Ricardo Patara and for improvement suggestions.

Authors' Addresses

Marla Azinger
Frontier Communications Corporation
Vancouver, WA
United States of America

Email: marla.azinger@ftr.com
URI: <http://www.frontiercorp.com/>

Leo Vegoda
Internet Corporation for Assigned Names and Numbers
4676 Admiralty Way, Suite 330
Marina del Rey, CA 90292
United States of America

Phone: +1-310-823-9358
Email: leo.vegoda@icann.org
URI: <http://www.iana.org/>

Network Working Group
Internet-Draft
Intended Status: Standards Track
Expires: January 8 2012

M. Chandramouli
Cisco Systems, Inc.
B. Schoening
Independent Consultant
J. Quittek
T. Dietz
NEC Europe Ltd.
B. Claise
Cisco Systems, Inc.
July 8, 2011

Power and Energy Monitoring MIB
draft-claise-energy-monitoring-mib-09

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79.

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/ietf/lid-abstracts.txt>

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

This Internet-Draft will expire on January 2012.

<Claise, et. Al>

Expires January 8, 2012

[Page 1]

Copyright (c) 2011 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Abstract

This document defines a subset of the Management Information Base (MIB) for power and energy monitoring of devices.

Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

Table of Contents

1. Introduction.....	3
2. The Internet-Standard Management Framework.....	4
3. Use Cases.....	4
4. Terminology.....	5
5. Architecture Concepts Applied to the MIB Module.....	5
5.1. Power Monitor Information.....	11
5.2. Power State.....	12
5.2.1. Power State Set.....	13
5.2.2. IEEE1621 Power State Set.....	13
5.2.3. DMTF Power State Set.....	13
5.2.4. EMAN Power State Set.....	14
5.3. Power Monitor Usage Information.....	17
5.4. Optional Power Usage Quality.....	18
5.5. Optional Energy Measurement.....	19

Internet-Draft	<Energy Monitoring MIB>	July 2011
5.6. Fault Management.....		22
6. Discovery.....		23
6.1. ENERGY-AWARE-MIB Module Implemented.....		23
6.2. ENERGY-AWARE-MIB Module Not Implemented, ENTITY-MIB Implemented.....		24
6.3. ENERGY-AWARE-MIB Module and ENTITY-MIB Not Implemented..		24
7. Link with the other IETF MIBs.....		24
7.1. Link with the ENTITY MIB and the ENTITY-SENSOR MIB..		24
7.2. Link with the ENTITY-STATE MIB.....		26
7.3. Link with the POWER-OVER-ETHERNET MIB.....		26
7.4. Link with the UPS MIB.....		27
7.5. Link with the LLDP and LLDP-MED MIBs.....		28
8. Implementation Scenarios.....		29
9. Structure of the MIB.....		31
10. MIB Definitions.....		31
11. Security Considerations.....		66
12. IANA Considerations.....		67
12.1. IANA Considerations for the MIB Modules.....		67
12.2. IANA Registration of new Power State Set.....		67
12.2.1. IANA Registration of the IEEE1621 Power State Set..		68
12.2.2. IANA Registration of the DMTF Power State Set.....		68
12.2.3. IANA Registration of the EMAN Power State Set.....		69
12. Contributors.....		69
13. Acknowledgment.....		69
14. Open Issues.....		69
15. References.....		71
15.2. Normative References.....		71
15.3. Informative References.....		71

1. Introduction

This document defines a subset of the Management Information Base (MIB) for use in energy management of devices within or connected to communication networks. The MIB modules in this document are designed to provide a model for energy management, which includes monitoring for power state and energy consumption of networked elements. This MIB takes into account the Power Management Architecture [EMAN-FRAMEWORK], which in turn, is based on the Power Monitoring Requirements [EMAN-REQ].

Energy management is applicable to devices in communication networks. Target devices for this specification include (but are not limited to): routers, switches, Power over Ethernet (PoE) endpoints, protocol gateways for building management

systems, intelligent meters, home energy gateways, hosts and servers, sensor proxies, etc.

Where applicable, device monitoring extends to the individual components of the device and to any attached dependent devices. For example: A device can contain components that are independent from a power-state point of view, such as line cards, processor cards, hard drives. A device can also have dependent attached devices, such as a switch with PoE endpoints or a power distribution unit with attached endpoints.

Devices and their sub-components may be characterized by the power-related attributes of a physical entity present in the ENTITY MIB, even though the ENTITY MIB compliance is not a requirement due to the variety and broad base of devices concerned with energy management.

2. The Internet-Standard Management Framework

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

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

3. Use Cases

Requirements for power and energy monitoring for networking devices are specified in [EMAN-REQ]. The requirements in [EMAN-REQ] cover devices typically found in communications networks, such as switches, routers, and various connected endpoints. For a power monitoring architecture to be useful, it should also apply to facility meters, power distribution units, gateway proxies for commercial building control, home automation devices, and devices that interface with the utility and/or smart grid. Accordingly, the scope of the MIB modules in this document is broader than that specified in [EMAN-REQ]. Several use cases for Energy Management have been identified in the "Energy Management (EMAN) Applicability Statement" [EMAN-AS].

4. Terminology

The definitions of basic terms like Power Monitor, Power Monitor Parent, Power Monitor Child, Power Monitor Meter Domain, Power State can be found in the Power Management Architecture [EMAN-FRAMEWORK].

EDITOR'S NOTE: it is foreseen that some more term will follow such a Proxy, Aggregator, Energy Management, etc...

Power State Set

A Power State Set is defined as a sequence of incremental energy saving modes of a device. The elements of this set can be viewed as an interface for the underlying device-implemented power settings of a device. Examples of Power State Sets include DTMF [DMTF], IEEE1621 [IEEE1621], ACPI [ACPI] and EMAN.

Power State

A Power State is defined as a specific power setting for a Power Monitor (e.g., shut, hibernate, sleep, high). Within the context of a Power State Set, the Power State of a device is one of the power saving modes in that Power State Set.

EDITOR'S NOTE: the definitions of Power State Series and Power State should be copied over in [EMAN-FRAMEWORK], and referenced here.

5. Architecture Concepts Applied to the MIB Module

This section describes the concepts specified in the Power Monitor Architecture [EMAN-FRAMEWORK] that pertain to power usage, with specific information related to the MIB module specified in this document. This subsection maps to the section "Architecture High Level Concepts" in the Power Monitoring Architecture [EMAN-FRAMEWORK].

The Energy Monitoring MIB has 2 independent MIB modules. The first MIB module powerMonitorMIB is focused on measurement of power and energy. The second MIB module powerQualityMIB is focused on Power Quality measurement.

The powerMonitorMIB MIB module consists of four tables. The first table pmPowerTable is indexed by pmPowerIndex and pmPowerStateSetIndex. The second table pmPowerStateTable indexed by pmPowerIndex, pmPowerStateSetIndex and pmPowerStateIndex. pmEnergyParametersTable and pmEnergyTable are indexed by pmPowerIndex.

```

pmPowerTable(1)
|
+---pmPowerEntry(1) [pmPowerIndex, pmPowerStateSet]
|
|   +--- --- Integer32          pmPowerIndex(1)
|   +--- --- PowerStateSet    pmPowerStateSet(2)
|   +--- r-n Integer32        pmPower(3)
|   +--- r-n Integer32        pmPowerNamePlate(4)
|   +--- r-n UnitMultiplier   pmPowerUnitMultiplier(5)
|   +--- r-n Integer32        pmPowerAccuracy(6)
|   +--- r-n INTEGER          pmMeasurementCaliber(7)
|   +--- r-n INTEGER          pmPowerCurrentType(8)
|   +--- r-n INTEGER          pmPowerOrigin(9)
|   +--- rwn Integer32        pmPowerAdminState(10)
|   +--- r-n Integer32        pmPowerOperState(11)
|   +--- r-n OwnerString      pmPowerStateEnterReason(12)
|
+---pmPowerStateTable(2)
|
|   +---pmPowerStateEntry(1)
|   |   [pmPowerIndex,
|   |   pmPowerStateSet,
|   |   pmPowerStateIndex]
|   +--- --- Integer32          pmPowerStateIndex(1)
|   +--- r-n Integer32          pmPowerStateMaxPower (2)
|   +--- r-n UnitMultiplier
|   |   pmPowerStatePowerUnitMultiplier (3)
|   +--- r-n TimeTicks          pmPowerStateTotalTime(4)
|   +--- r-n Counter64          pmPowerStateEnterCount(5)
|
+pmEnergyParametersTable(1)
+---pmEnergyParametersEntry(1) [pmPowerIndex]
|
|   +--- r-n TimeInterval
|   |   pmEnergyParametersIntervalLength (1)
|   +--- r-n Integer32
|   |   pmEnergyParametersIntervalNumber (2)
|   +--- r-n Integer32
|   |   pmEnergyParametersIntervalMode (3)

```

```

|   +-- r-n TimeInterval
|       pmEnergyParametersIntervalWindow (4)
|   +-- r-n Integer32
|       pmEnergyParametersSampleRate (5)
|   +-- r-n RowStatus pmEnergyParametersStatus (6)
|
+pmEnergyTable(1)
+----pmEnergyEntry(1) [pmPowerIndex]
|
|   +-- r-n TimeInterval pmEnergyIntervalStartTime (1)
|   +-- r-n Integer32 pmEnergyIntervalEnergyUsed (2)
|   +-- r-n UnitMultiplier
|       pmEnergyIntervalEnergyUnitMultiplier (3)
|   +-- r-n Integer32 pmEnergyIntervalMax (4)
|   +-- r-n TimeTicks
|       pmEnergyIntervalDiscontinuityTime(5)
|   +-- r-n RowStatus pmEnergyParametersStatus (6)

```

The powerQualityMIB consists of four tables. PmACPwrQualityTable is indexed by pmPowerIndex. PmACPwrQualityPhaseTable is indexed by pmPowerIndex and pmPhaseIndex. pmACPwrQualityWyePhaseTable and pmACPwrQualityDelPhaseTable are indexed by pmPowerIndex and pmPhaseIndex.

```

pmPowerTable(1)
+----PmACPwrQualityEntry (1) [pmPowerIndex]
|
|   +----- INTEGER pmACPwrQualityConfiguration (1)
|   +-- r-n Integer32 pmACPwrQualityAvgVoltage (2)
|   +-- r-n Integer32 pmACPwrQualityAvgCurrent (3)
|   +-- r-n Integer32 pmACPwrQualityFrequency (4)
|   +-- r-n UnitMultiplier
|       pmACPwrQualityPowerUnitMultiplier (5)
|   +-- r-n Integer32 pmACPwrQualityPowerAccuracy (6)
|   +-- r-n Integer32 pmACPwrQualityTotalActivePower (7)
|   +-- r-n Integer32
|       pmACPwrQualityTotalReactivePower (8)
|   +-- r-n Integer32 pmACPwrQualityTotalApparentPower (9)
|   +-- r-n Integer32 pmACPwrQualityTotalPowerFactor(10)
|   +-- r-n Integer32 pmACPwrQualityThdAmpheres (11)
|
+pmACPwrQualityPhaseTable (1)
+----PmACPwrQualityPhaseEntry(1)[pmPowerIndex,
|                                     pmPhaseIndex]

```

```

|   +-- r-n Integer32 pmPhaseIndex (1)
|   +-- r-n Integer32
|       |
|       |   pmACPwrQualityPhaseAvgCurrent (2)
|   +-- r-n Integer32
|       |
|       |   pmACPwrQualityPhaseActivePower (3)
|   +-- r-n Integer32
|       |
|       |   pmACPwrQualityPhaseReactivePower (4)
|   +-- r-n Integer32
|       |
|       |   pmACPwrQualityPhaseApparentPower (5)
|   +-- r-n Integer32
|       |
|       |   pmACPwrQualityPhasePowerFactor (6)
|   +-- r-n Integer32
|       |
|       |   pmACPwrQualityPhaseImpedance (7)
|
+pmACPwrQualityDelPhaseTable (1)
+-- pmACPwrQualityDelPhaseEntry(1)
|
|   [pmPowerIndex,
|   pmPhaseIndex]
|   +-- r-n Integer32
|       |
|       |   pmACPwrQualityDelPhaseToNextPhaseVoltage (1)
|   +-- r-n Integer32
|       |
|       |   pmACPwrQualityDelThdPhaseToNextPhaseVoltage (2)
|   +-- r-n Integer32 pmACPwrQualityDelThdCurrent (3)
|
+pmACPwrQualityWyePhaseTable (1)
+-- pmACPwrQualityWyePhaseEntry (1)
|
|   [pmPowerIndex,
|   pmPhaseIndex]
|   +-- r-n Integer32
|       |
|       |   pmACPwrQualityWyePhaseToNeutralVoltage (1)
|   +-- r-n Integer32
|       |
|       |   pmACPwrQualityWyePhaseCurrent (2)
|   +-- r-n Integer32
|       |
|       |   pmACPwrQualityWyeThdPhaseToNeutralVoltage (3)
|
.

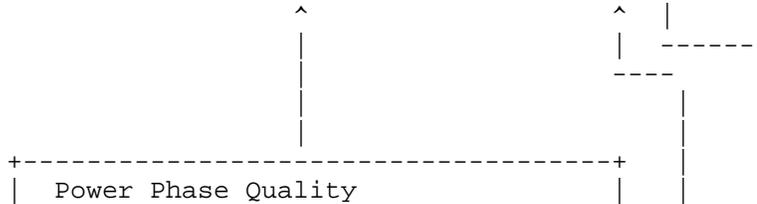
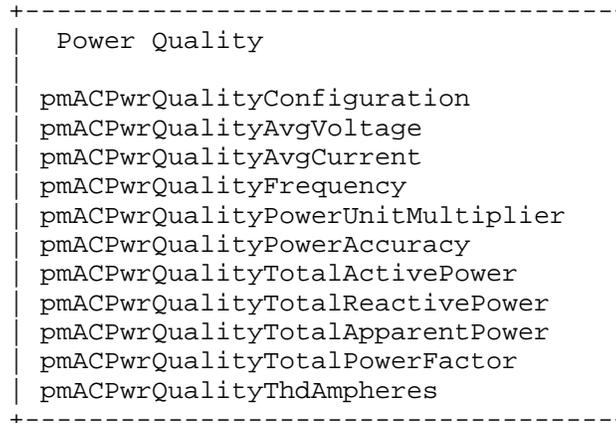
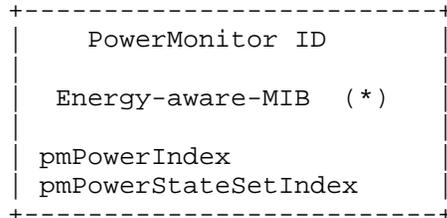
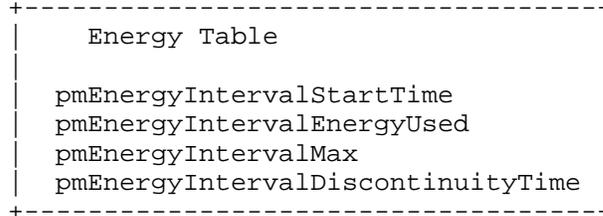
```

A UML representation of the MIB objects in the two MIB modules are powerMonitorMIB and powerQualityMIB are presented.



Figure 1:UML diagram for powerMonitor MIB

(*) Link with the ENERGY-AWARE-MIB



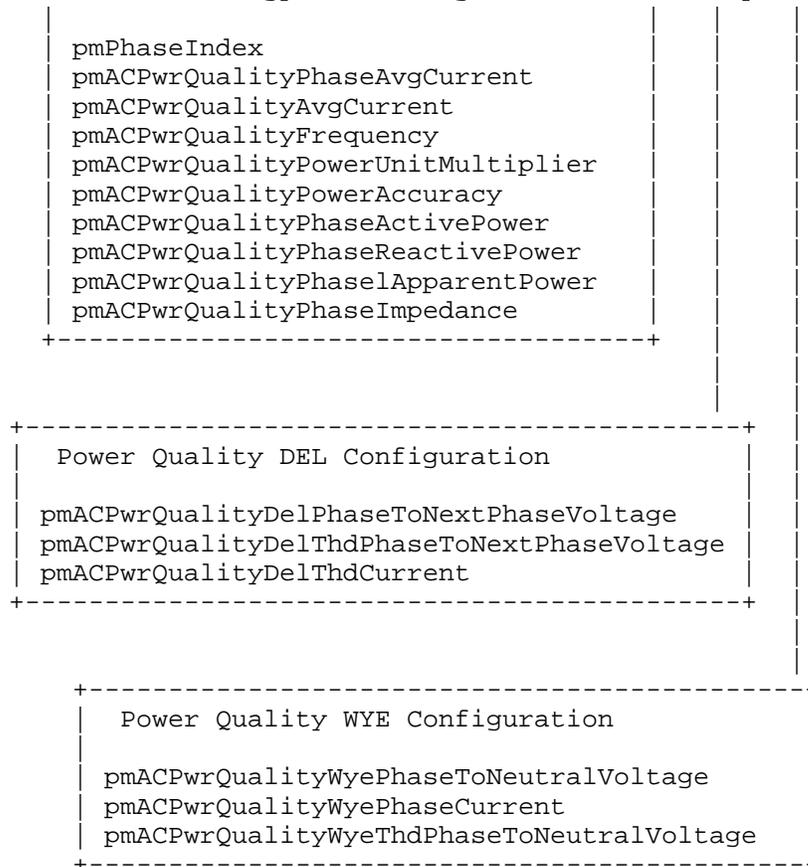


Figure 2: UML diagram for the powerQualityMIB

5.1. Power Monitor Information

Refer to the "Power Monitor Information" section in [EMAN-FRAMEWORK] for background information. An energy aware device is considered an instance of a Power Monitor as defined in the [EMAN-FRAMEWORK].

The Power Monitor identity information is specified in the MIB ENERGY-AWARE-MIB module [EMAN-AWARE-MIB] primary table, i.e. the pmTable. In this table, every Power Monitor SHOULD have a printable name pmName, and MUST HAVE a unique Power Monitor index pmIndex. The ENERGY-AWARE-MIB module returns the relationship (parent/child) between Power Monitors.

EDITOR'S NOTE: this last sentence will have to be updated with terms such as Aggregator, Proxy, etc... when the [EMAN-FRAMEWORK] will stabilize.

5.2. Power State

Refer to the "Power Monitor States" section in [EMAN-FRAMEWORK] for background information.

A Power Monitor may have energy conservation modes called Power States. Between the ON and OFF states of a device, there can be several intermediate energy saving modes. Those energy saving modes are called as Power States.

Power States, which represent universal states of power management of a Power Monitor, are specified by the pmPowerState MIB object. The actual Power State is specified by the pmPowerOperState MIB object, while the pmPowerAdminState MIB object specifies the Power State requested for the Power Monitor. The difference between the values of pmPowerOperState and pmPowerAdminState can be attributed that the Power Monitor is busy transitioning from pmPowerAdminState into the pmPowerOperState, at which point it will update the content of pmPowerOperState. In addition, the possible reason for change in Power State is reported in pmPowerStateEnterReason. Regarding pmPowerStateEnterReason, management stations and Power Monitors should support any format of the owner string dictated by the local policy of the organization. It is suggested that this name contain at least the reason for the transition change, and one or more of the following: IP address, management station name, network manager's name, location, or phone number.

The MIB objects pmPowerOperState, pmPowerAdminState, and pmPowerStateEnterReason are contained in the pmPowerTable MIB table.

The pmPowerStateTable table enumerates the maximum power usage in watts, for every single supported Power State of each Power State Set supported by the Power Monitor. In addition, PowerStateTable provides additional statistics: pmPowerStateEnterCount, the number of times an entity has visited a particular Power State, and pmPowerStateTotalTime, the total time spent in a particular Power State of a Power Monitor.

5.2.1. Power State Set

There are several standards and implementations of Power State Sets. A Power Monitor can support one or multiple Power State Set implementation(s) concurrently.

There are currently three Power State Sets advocated:

- Reserved(0)
- IEEE1621(1) - [IEEE1621]
- DMTF(2) - [DMTF]
- EMAN(3) - [EMAN-MONITORING-MIB]

The respective specific states related to each Power State Set are specified in the following sections.

5.2.2. IEEE1621 Power State Set

The IEEE1621 Power State Set [IEEE1621] consists of 3 rudimentary states : on, off or sleep.

- on(0) - The device is fully On and all features of the device are in working mode.
- off(1) - The device is mechanically switched off and does not consume energy.
- sleep(2) - The device is in a power saving mode, and some features may not be available immediately.

5.2.3. DMTF Power State Set

DMTF [DMTF] standards organization has defined a power profile standard based on the CIM (Common Information Model) model that consists of 15 power states ON (2), SleepLight (3), SleepDeep (4), Off-Hard (5), Off-Soft (6), Hibernate(7), PowerCycle Off-Soft (8), PowerCycle Off-Hard (9), MasterBus reset (10), Diagnostic Interrupt (11), Off-Soft-Graceful (12), Off-Hard Graceful (13), MasterBus reset Graceful (14), Power-Cycle Off-Soft Graceful (15), PowerCycle-Hard Graceful (16). DMTF standard is targeted for hosts and computers. Details of the semantics of each Power State within the DMTF Power State Set can be obtained from the DMTF Power State Management Profile specification [DMTF].

DMTF power profile extends ACPI power states. The following table provides a mapping between DMTF and ACPI Power State Set:

DMTF Power State	ACPI Power State
Reserved(0)	
Reserved(1)	
ON (2)	G0-S0
Sleep-Light (3)	G1-S1 G1-S2
Sleep-Deep (4)	G1-S3
Power Cycle (Off-Soft) (5)	G2-S5
Off-hard (6)	G3
Hibernate (Off-Soft) (7)	G1-S4
Off-Soft (8)	G2-S5
Power Cycle (Off-Hard) (9)	G3
Master Bus Reset (10)	G2-S5
Diagnostic Interrupt (11)	G2-S5
Off-Soft Graceful (12)	G2-S5
Off-Hard Graceful (13)	G3
MasterBus Reset Graceful (14)	G2-S5
Power Cycle off-soft Graceful (15)	G2-S5
Power Cycle off-hard Graceful (16)	G3

Figure 3: DMTF and ACPI Powe State Set Mapping

5.2.4. EMAN Power State Set

The EMAN Power State Set represents an attempt for a uniform standard approach to model the different levels of power consumption of a device. The EMAN Power States are an expansion of the basic Power States as defined in IEEE1621 that also

incorporate the Power States defined in ACPI and DMTF. Therefore, in addition to the non-operational states as defined in ACPI and DMTF standards, several intermediate operational states have been defined.

There are twelve Power States, that expand on IEEE1621 on, sleep and off. The expanded list of Power States are divided into six operational states, and six non-operational states. The lowest non-operational state is 1 and the highest is 6. Each non-operational state corresponds to an ACPI state [ACPI] corresponding to Global and System states between G3 (hard-off) and G1 (sleeping). For Each operational state represent a performance state, and may be mapped to ACPI states P0 (maximum performance power) through P5 (minimum performance and minimum power).

An Power Monitor may have fewer Power States than twelve and would then map several policy states to the same power state. Power Monitor with more than twelve states, would choose which twelve to represent as power policy states.

In each of the non-operational states (from mechoff(1) to ready(6)), the Power State preceding it is expected to have a lower power consumption and a longer delay in returning to an operational state:

IEEE1621 Power(off):

mechoff(1) : An off state where no entity features are available. The entity is unavailable. No energy is being consumed and the power connector can be removed. This corresponds to ACPI state G3.

softoff(2) : Similar to mechoff(1), but some components remain powered or receive trace power so that the entity can be awakened from its off state. In softoff(2), no context is saved and the device typically requires a complete boot when awakened. This corresponds to ACPI state G2.

IEEE1621 Power(sleep)

hibernate(3): No entity features are available. The entity may be awakened without requiring a complete boot, but the time for

availability is longer than sleep(4). An example for state hibernate(3) is a save-to-disk state where DRAM context is not maintained. Typically, energy consumption is zero or close to zero. This corresponds to state G1, S4 in ACPI.

sleep(4) : No entity features are available, except for out-of-band management, for example wake-up mechanisms. The time for availability is longer than standby(5). An example for state sleep(4) is a save-to-RAM state, where DRAM context is maintained. Typically, energy consumption is close to zero. This corresponds to state G1, S3 in ACPI.

standby(5) : No entity features are available, except for out-of-band management, for example wake-up mechanisms. This mode is analogous to cold-standby. The time for availability is longer than ready(6). For example, the processor context is not maintained. Typically, energy consumption is close to zero. This corresponds to state G1, S2 in ACPI.

ready(6) : No entity features are available, except for out-of-band management, for example wake-up mechanisms. This mode is analogous to hot-standby. The entity can be quickly transitioned into an operational state. For example, processors are not executing, but processor context is maintained. This corresponds to state G1, S1 in ACPI.

IEEE1621 Power(on):

lowMinus(7) : Indicates some entity features may not be available and the entity has selected measures/options to provide less than low(8) usage. This corresponds to ACPI State G0. This includes operational states lowMinus(7) to full(12).

low(8) : Indicates some features may not be available and the entity has taken

measures or selected options to provide less than mediumMinus(9) usage.

mediumMinus(9): Indicates all entity features are available but the entity has taken measures or selected options to provide less than medium(10) usage.

medium(10) : Indicates all entity features are available but the entity has taken measures or selected options to provide less than highMinus(11) usage.

highMinus(11): Indicates all entity features are available and power usage is less than high(12).

high(12) : Indicates all entity features are available and the entity is consuming the highest power.

5.3. Power Monitor Usage Information

Refer to the "Power Monitor Usage Measurement" section in [EMAN-FRAMEWORK] for background information.

For a Power Monitor, power usage is reported using pmPower. The magnitude of measurement is based on the pmPowerUnitMultiplier MIB variable, based on the UnitMultiplier Textual Convention (TC). Power measurement magnitude should conform to the IEC 62053-21 [IEC.62053-21] and IEC 62053-22 [IEC.62053-22] definition of unit multiplier for the SI (System International) units of measure. Measured values are represented in SI units obtained by BaseValue * 10 raised to the power of the scale.

For example, if current power usage of a Power Monitor is 3, it could be 3 W, 3 mW, 3 KW, or 3 MW, depending on the value of pmPowerUnitMultiplier. Note that other measurements throughout the two MIB modules in this document use the same mechanism, including pmPowerStatePowerUnitMultiplier, pmEnergyIntervalEnergyUnitMultiplier, and pmACPwrQualityPowerUnitMultiplier.

In addition to knowing the usage and magnitude, it is useful to know how a pmPower measurement was obtained. An NMS can use this to account for the accuracy and nature of the reading

between different implementations. For this `pmPowerOrigin` describes whether the measurements were made at the device itself or from a remote source. The `pmPowerMeasurementCaliber` describes the method that was used to measure the power and can distinguish actual or estimated values. There may be devices in the network, which may not be able to measure or report power consumption. For those devices, the object `pmPowerMeasurementCaliber` shall report that measurement mechanism is "unavailable" and the `pmPower` measurement shall be "0".

The nameplate power rating of a Power Monitor is specified in `pmPowerNameplate` MIB object.

5.4. Optional Power Usage Quality

Refer to the "Optional Power Usage Quality" section in [EMAN-FRAMEWORK] for background information.

The optional `powerQualityMIB` MIB module can be implemented to further describe power usage quality measurement. The `powerQualityMIB` MIB module adheres closely to the IEC 61850 7-2 standard to describe AC measurements.

The `powerQualityMIB` MIB module contains a primary table, the `pmACPwrQualityTable` table, that defines power quality measurements for supported `pmIndex` entities, as a sparse extension of the `pmPowerTable` (with `pmPowerIndex` as primary index). This `pmACPwrQualityTable` table contains such information as the configuration (single phase, DEL 3 phases, WYE 3 phases), voltage, frequency, power accuracy, total active/reactive power/apparent power, amperage, and voltage.

In case of 3-phase power, the `pmACPwrQualityPhaseTable` additional table is populated with power quality measurements per phase (so double indexed by the `pmPowerIndex` and `pmPhaseIndex`). This table, which describes attributes common to both WYE and DEL configurations, contains the average current, active/reactive/apparent power, power factor, and impedance.

In case of 3-phase power with a DEL configuration, the `pmACPwrQualityDelPhaseTable` table describes the phase-to-phase power quality measurements, i.e., voltage and current.

In case of 3-phase power with a Wye configuration, the `pmACPwrQualityWyePhaseTable` table describes the phase-to-neutral power quality measurements, i.e., voltage and current.

5.5. Optional Energy Measurement

Refer to the "Optional Energy and demand Measurement" section in [EMAN-FRAMEWORK] for the definition and terminology information.

It is relevant to measure energy when there are actual power measurements from a Power Monitor, and not when the power measurement is assumed or predicted as specified in the description clause of the object pmPowerMeasurementCaliber.

Two tables are introduced to characterize energy measurement of a Power Monitor: pmEnergyTable and pmEnergyParametersTable. Both energy and demand information can be represented via the pmEnergyTable. Energy information will be an accumulation with no interval. Demand information can be represented as an average accumulation per interval of time.

The pmEnergyParametersTable consists of the parameters defining the duration of measurement intervals in seconds, (pmEnergyParametersIntervalLength), the number of successive intervals to be stored in the pmEnergyTable, (pmEnergyParametersIntervalNumber), the type of measurement technique (pmEnergyParametersIntervalMode), and a sample rate used to calculate the average (pmEnergyParametersSampleRate). Judicious choice of the sampling rate will ensure accurate measurement of energy while not imposing an excessive polling burden.

There are three pmEnergyParametersIntervalMode types used for energy measurement collection: period, sliding, and total. The choices of the the three different modes of collection are based on IEC standard 61850-7-4. Note that multiple pmEnergyParametersIntervalMode types MAY be configured simultaneously.

These three pmEnergyParametersIntervalMode types are illustrated by the following three figures, for which:

- The horizontal axis represents the current time, with the symbol <--- L ---> expressing the pmEnergyParametersIntervalLength, and the pmEnergyIntervalStartTime is represented by S1, S2, S3, S4, ..., Sx where x is the value of pmEnergyParametersIntervalNumber.
- The vertical axis represents the time interval of sampling and the value of pmEnergyIntervalEnergyUsed can be obtained at the

end of the sampling period. The symbol ===== denotes the duration of the sampling period.

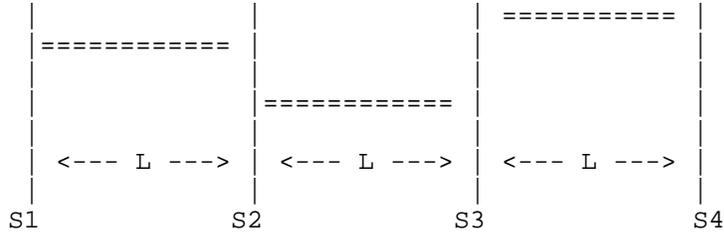


Figure 4 : Period pmEnergyParametersIntervalMode

A pmEnergyParametersIntervalMode type of 'period' specifies non-overlapping periodic measurements. Therefore, the next pmEnergyIntervalStartTime is equal to the previous pmEnergyIntervalStartTime plus pmEnergyParametersIntervalLength. S2=S1+L; S3=S2+L, ...

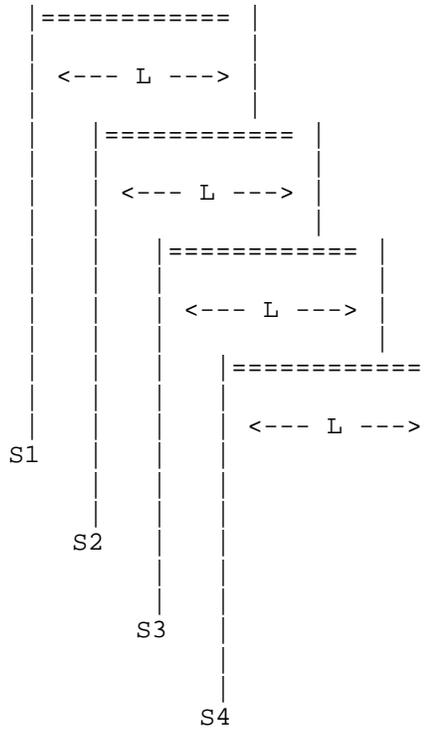


Figure 5 : Sliding pmEnergyParametersIntervalMode

A pmEnergyParametersIntervalMode type of 'sliding' specifies overlapping periodic measurements.

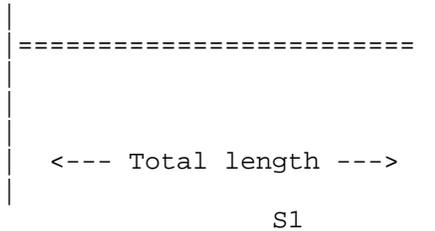


Figure 4 : Total pmEnergyParametersIntervalMode

A pmEnergyParametersIntervalMode type of 'total' specifies a continuous measurement since the last reset. The value of pmEnergyParametersIntervalNumber should be (1) one and pmEnergyParametersIntervalLength is ignored.

The pmEnergyParametersStatus is used to start and stop energy usage logging. The status of this variable is "active" when all the objects in pmEnergyParametersTable are appropriate which in turn indicates if pmEnergyTable entries exist or not.

The pmEnergyTable consists of energy measurements inpmEnergyIntervalEnergyUsed , the units of the measured energy pmEnergyIntervalEnergyUnitMultiplier, and the maximum observed energy within a window - pmEnergyIntervalMax.

Measurements of the total energy consumed by a Power Monitor may suffer from interruptions in the continuous measurement of energy consumption. In order to indicate such interruptions, the object pmEnergyIntervalDiscontinuityTime is provided for indicating the time of the last interruption of total energy measurement. pmEnergyIntervalDiscontinuityTime shall indicate the sysUpTime [RFC3418] when the device was reset.

The following example illustrates the pmEnergyTable and pmEnergyParametersTable:

First, in order to estimate energy, a time interval to sample energy should be specified, i.e. pmEnergyParametersIntervalLength can be set to "900 seconds" or 15 minutes and the number of consecutive intervals over which

the maximum energy is calculated (`pmEnergyParametersIntervalNumber`) as "10". The sampling rate internal to the Power Monitor for measurement of power usage (`pmEnergyParametersSampleRate`) can be "1000 milliseconds", as set by the Power Monitor as a reasonable value. Then, the `pmEnergyParametersStatus` is set to active (value 1) to indicate that the Power Monitor should start monitoring the usage per the `pmEnergyTable`.

The indices in the `pmEnergyTable` are `pmPowerIndex`, which identifies the Power Monitor, and `pmEnergyIntervalStartTime`, which denotes the start time of the energy measurement interval based on `sysUpTime` [RFC3418]. The value of `pmEnergyIntervalEnergyUsed` is the measured energy consumption over the time interval specified (`pmEnergyParametersIntervalLength`) based on the Power Monitor internal sampling rate (`pmEnergyParametersSampleRate`). While choosing the values for the `pmEnergyParametersIntervalLength` and `pmEnergyParametersSampleRate`, it is recommended to take into consideration either the network element resources adequate to process and store the sample values, and the mechanism used to calculate the `pmEnergyIntervalEnergyUsed`. The units are derived from `pmEnergyIntervalPowerUnitMultiplier`. For example, `pmEnergyIntervalPowerUsed` can be "100" with `pmEnergyIntervalPowerUnits` equal to 0, the measured energy consumption of the Power Monitor is 100 watt-hours. The `pmEnergyIntervalMax` is the maximum energy observed and that can be "150 watt-hours".

The `pmEnergyTable` has a buffer to retain a certain number of intervals, as defined by `pmEnergyParametersIntervalNumber`. If the default value of "10" is kept, then the `pmEnergyTable` contains 10 energy measurements, including the maximum.

Here is a brief explanation of how the maximum energy can be calculated. The first observed energy measurement value is taken to be the initial maximum. With each subsequent measurement, based on numerical comparison, maximum energy may be updated. The maximum value is retained as long as the measurements are taking place. Based on periodic polling of this table, an NMS could compute the maximum over a longer period, i.e. a month, 3 months, or a year.

5.6. Fault Management

[EMAN-REQ] specifies requirements about Power States such as "the current power state" , "the time of the last state change",

"the total time spent in each state", "the number of transitions to each state" etc. Some of these requirements are fulfilled explicitly by MIB objects such as pmPowerOperState, pmPowerStateTotalTime and pmPowerStateEnterCount. Some of the other requirements are met via the SNMP NOTIFICATION mechanism. pmPowerStateChange SNMP notification which is generated when the value(s) of pmPowerStateSet, pmPowerOperState, pmPowerAdminState have changed.

6. Discovery

6.1. ENERGY-AWARE-MIB Module Implemented

The NMS must first poll the ENERGY-AWARE-MIB module [EMAN-AWARE-MIB], if available, in order to discover all the Power Monitors and the relationships between those (notion of Parent/Child). In the ENERGY-AWARE-MIB module tables, the Power Monitors are indexed by the pmIndex.

If an implementation of the ENERGY-AWARE-MIB module is available in the local SNMP context, for the same Power Monitor, the pmIndex value (EMAN-AWARE-MIB) MUST be assigned to the pmPowerIndex for The pmPowerIndex characterizes the Power Monitor in the powerMonitorMIB and powerQualityMIB MIB modules (this document).

From there, the NMS must poll the pmPowerStateTable (specified in the powerMonitorMIB module in this document), which enumerates, amongst other things, the maximum power usage. As the entries in pmPowerStateTable table are indexed by the Power Monitor (pmPowerIndex), by the Power State Set (pmPowerStateSetIndex), and by the Power State (pmPowerStateIndex), the maximum power usage is discovered per Power Monitor, per Power State Set, and per Power Usage. In other words, polling the pmPowerStateTable allows the discovery of each Power State within every Power State Set supported by the Power Monitor.

If the Power Monitor is an Aggregator or a Proxy, the MIB module would be populated with the Power Monitor Parent and Children information, which have their own Power Monitor index value (pmPowerIndex). However, the parent/child relationship must be discovered thanks to the ENERGY-AWARE-MIB module.

Finally, the NMS can monitor the Power Quality thanks to the powerQualityMIB MIB module, which reuses the pmPowerIndex to index the Power Monitor.

6.2. ENERGY-AWARE-MIB Module Not Implemented, ENTITY-MIB Implemented

When the ENERGY-AWARE-MIB module [EMAN-AWARE-MIB] is not implemented, the NMS must poll the ENTITY-MIB [RFC4133] in order to discover some more information about the Power Monitors. Indeed, the index for the Power Monitors in the MIB modules specified in this document is the pmPowerIndex, which specifies: "If there is no implementation of the ENERGY-AWARE-MIB module but one of the ENTITY MIB module is available in the local SNMP context, then the same index of an entity MUST be chosen as assigned to the entity by object entPhysicalIndex in the ENTITY MIB module."

As the Section 6.1. , the NMS must then poll the pmPowerStateTable (specified in the powerMonitorMIB module in this document), indexed by the Power Monitor (pmPowerIndex that inherited the entPhysicalIndex value), by the Power State Set (pmPowerStateSetIndex), and by the Power State (pmPowerStateIndex). Then the NMS has discovered every Power State within each Power State Set supported by the Power Monitor.

Note that, without the ENERGY-AWARE-MIB module, the Power Monitor acts as an standalone device, i.e. the notion of parent/child can't be specified.

6.3. ENERGY-AWARE-MIB Module and ENTITY-MIB Not Implemented

If neither the ENERGY-AWARE-MIB module [EMAN-AWARE-MIB] nor of the ENTITY MIB module [RFC4133] are available in the local SNMP context, then this MIB module may choose identity values from a further MIB module providing entity identities.

Note that, without the ENERGY-AWARE-MIB module, the Power Monitor acts as an standalone device, i.e. the notion of parent/child can't be specified.

7. Link with the other IETF MIBs

7.1. Link with the ENTITY MIB and the ENTITY-SENSOR MIB

RFC 4133 [RFC4133] defines the ENTITY MIB module that lists the physical entities of a networking device (router, switch, etc.)

and those physical entities indexed by entPhysicalIndex. From an energy-management standpoint, the physical entities that consume or produce energy are of interest.

RFC 3433 [RFC3433] defines the ENTITY-SENSOR MIB module that provides a standardized way of obtaining information (current value of the sensor, operational status of the sensor, and the data units precision) from sensors embedded in networking devices. Sensors are associated with each index of entPhysicalIndex of the ENTITY MIB [RFC4133]. While the focus of the Power and Energy Monitoring MIB is on measurement of power usage of networking equipment indexed by the ENTITY MIB, this MIB proposes a customized power scale for power measurement and different power state states of networking equipment, and functionality to configure the power state states.

When this MIB module is used to monitor the power usage of devices like routers and switches, the ENTITY MIB and ENTITY-SENSOR MIB SHOULD be implemented. In such cases, the Power Monitors are modeled by the entPhysicalIndex through the pmPhysicalEntity MIB object specified in the pmTable in the ENERGY-AWARE-MIB MIB module [EMAN-AWARE-MIB].

However, the ENTITY-SENSOR MIB [RFC3433] does not have the ANSI C12.x accuracy classes required for electricity (i.e., 1%, 2%, 0.5% accuracy classes). Indeed, entPhySensorPrecision [RFC3433] represents "The number of decimal places of precision in fixed-point sensor values returned by the associated entPhySensorValue object". The ANSI and IEC Standards are used for power measurement and these standards require that we use an accuracy class, not the scientific-number precision model specified in RFC3433. The pmPowerAccuracy MIB object models this accuracy. Note that pmPowerUnitMultiplier represents the scale factor per IEC 62053-21 [IEC.62053-21] and IEC 62053-22 [IEC.62053-22], which is a more logical representation for power measurements (compared to entPhySensorScale), with the mantissa and the exponent values $X * 10 ^ Y$.

Power measurements specifying the qualifier 'UNITS' for each measured value in watts are used in the LLDP-EXT-MED-MIB, POE [RFC3621], and UPS [RFC1628] MIBs. The same 'UNITS' qualifier is used for the power measurement values.

One cannot assume that the ENTITY MIB and ENTITY-SENSOR MIB are implemented for all Power Monitors that need to be monitored. A typical example is a converged building gateway, monitoring several other devices in the building, doing the proxy between SNMP and a protocol like BACNET. Another example is the home

Internet-Draft <Energy Monitoring MIB> July 2011
energy controller. In such cases, the pmPhysicalEntity value
contains the zero value, thanks to PhysicalIndexOrZero textual
convention.

The pmPowerIndex MIB object has been kept as the unique Power
Monitor index. The pmPower is similar to entPhySensorValue
[RFC3433] and the pmPowerUnitMultiplier is similar to
entPhySensorScale.

7.2. Link with the ENTITY-STATE MIB

For each entity in the ENTITY-MIB [RFC4133], the ENTITY-STATE
MIB [RFC4268] specifies the operational states (entStateOper:
unknown, enabled, disabled, testing), the alarm (entStateAlarm:
unknown, underRepair, critical, major, minor, warning,
indeterminate) and the possible values of standby states
(entStateStandby: unknown, hotStandby, coldStandby,
providingService).

From a power monitoring point of view, in contrast to the entity
operational states of entities, Power States are required, as
proposed in the Power and Energy Monitoring MIB module. Those
Power States can be mapped to the different operational states
in the ENTITY-STATE MIB, if a formal mapping is required. For
example, the entStateStandby "unknown", "hotStandby",
"coldStandby", states could map to the Power State "unknown",
"ready", "standby", respectively, while the entStateStandby
"providingService" could map to any "low" to "high" Power State.

7.3. Link with the POWER-OVER-ETHERNET MIB

Power-over-Ethernet MIB [RFC3621] provides an energy monitoring
and configuration framework for power over Ethernet devices.
The RFC introduces a concept of a port group on a switch to
define power monitoring and management policy and does not use
the entPhysicalIndex as the index. Indeed, the
pethMainPseConsumptionPower is indexed by the
pethMainPseGroupIndex, which has no mapping with the
entPhysicalIndex.

One cannot assume that the Power-over-Ethernet MIB is
implemented for all Power Monitors that need to be monitored. A
typical example is a converged building gateway, monitoring
several other devices in the building, doing the proxy between
SNMP and a protocol like BACNET. Another example is the home
energy controller. In such cases, the pmethPortIndex and

pmethPortGrpIndex values contain the zero value, thanks to new PethPsePortIndexOrZero and textual PethPsePortGroupIndexOrZero conventions.

However, if the Power-over-Ethernet MIB [RFC3621] is supported, the Power Monitor pmethPortIndex and pmethPortGrpIndex contain the pethPsePortIndex and pethPsePortGroupIndex, respectively.

As a consequence, the pmPowerIndex MIB object has been kept as the unique Power Monitor index.

Note that, even though the Power-over-Ethernet MIB [RFC3621] was created after the ENTITY-SENSOR MIB [RFC3433], it does not reuse the precision notion from the ENTITY-SENSOR MIB, i.e. the entPhySensorPrecision MIB object.

7.4. Link with the UPS MIB

To protect against unexpected power disruption, data centers and buildings make use of Uninterruptible Power Supplies (UPS). To protect critical assets, a UPS can be restricted to a particular subset or domain of the network. UPS usage typically lasts only for a finite period of time, until normal power supply is restored. Planning is required to decide on the capacity of the UPS based on output power and duration of probable power outage. To properly provision UPS power in a data center or building, it is important to first understand the total demand required to support all the entities in the site. This demand can be assessed and monitored via the Power and Energy Monitoring MIB.

UPS MIB [RFC1628] provides information on the state of the UPS network. Implementation of the UPS MIB is useful at the aggregate level of a data center or a building. The MIB module contains several groups of variables:

- upsIdent: Identifies the UPS entity (name, model, etc.).
- upsBattery group: Indicates the battery state (upsbatteryStatus, upsEstimatedMinutesRemaining, etc.)
- upsInput group: Characterizes the input load to the UPS (number of input lines, voltage, current, etc.).
- upsOutput: Characterizes the output from the UPS (number of output lines, voltage, current, etc.)
- upsAlarms: Indicates the various alarm events.

The measurement of power in the UPS MIB is in Volts, Amperes and Watts. The units of power measurement are RMS volts and RMS Amperes. They are not based on the EntitySensorDataScale and EntitySensorDataPrecision of Entity-Sensor MIB.

Both the Power and Energy Monitoring MIB and the UPS MIB may be implemented on the same UPS SNMP agent, without conflict. In this case, the UPS device itself is the Power Monitor Parent and any of the UPS meters or submeters are the Power Monitor Children.

7.5. Link with the LLDP and LLDP-MED MIBs

The LLDP Protocol is a Data Link Layer protocol used by network devices to advertise their identities, capabilities, and interconnections on a LAN network.

The Media Endpoint Discovery is an enhancement of LLDP, known as LLDP-MED. The LLDP-MED enhancements specifically address voice applications. LLDP-MED covers 6 basic areas: capability discovery, LAN speed and duplex discovery, network policy discovery, location identification discovery, inventory discovery, and power discovery.

Of particular interest to the current MIB module is the power discovery, which allows the endpoint device (such as a PoE phone) to convey power requirements to the switch. In power discovery, LLDP-MED has four Type Length Values (TLVs): power type, power source, power priority and power value. Respectively, those TLVs provide information related to the type of power (power sourcing entity versus powered device), how the device is powered (from the line, from a backup source, from external power source, etc.), the power priority (how important is it that this device has power?), and how much power the device needs.

The power priority specified in the LLDP-MED MIB [LLDP-MED-MIB] actually comes from the Power-over-Ethernet MIB [RFC3621]. If the Power-over-Ethernet MIB [RFC3621] is supported, the exact value from the pethPsePortPowerPriority [RFC3621] is copied over in the lldpXMedRemXPoEPDPowerPriority [LLDP-MED-MIB]; otherwise the value in lldpXMedRemXPoEPDPowerPriority is "unknown". From the Power and Energy Monitoring MIB, it is possible to identify the pethPsePortPowerPriority [RFC3621], thanks to the pmethPortIndex and pmethPortGrpIndex.

The `lldpXMedLocXPoEPDPowerSource` [LLDP-MED-MIB] is similar to `pmPowerOrigin` in indicating if the power for an attached device is local or from a remote device. If the LLDP-MED MIB is supported, the following mapping can be applied to the `pmPowerOrigin`: `lldpXMedLocXPoEPDPowerSource` from `PSE(2)` and `local(3)` can be mapped to `remote(2)` and `self(1)`, respectively.

8. Implementation Scenarios

This section provides an illustrative example scenario for the implementation of the Power Monitor, including Power Monitor Parent and Power Monitor Child relationships.

Example Scenario of a campus network: Switch with PoE Endpoints with further connected Devices

The campus network consists of switches that provide LAN connectivity. The switch with PoE ports is located in wiring closet. PoE IP phones are connected to the switch. The IP phones draw power from the PoE ports of the switch. In addition, a PC is daisy-chained from the IP phone for LAN connectivity.

The IP phone consumes power from the PoE switch, while the PC consumes power from the wall outlet.

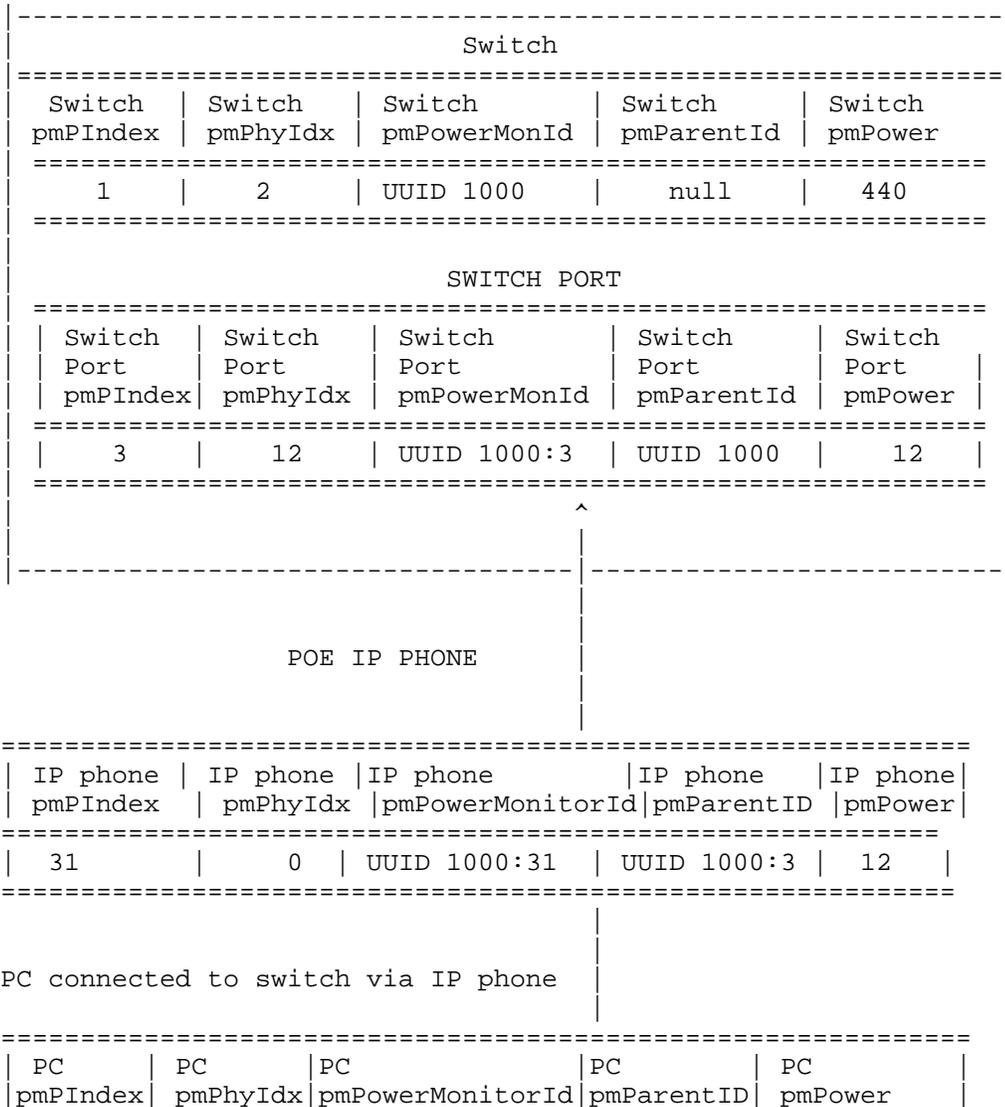
The switch has implementations of Entity MIB [RFC4133] and energy-aware MIB [EMAN-AWARE-MIB] while the PC does not have implementation of the Entity MIB, but has an implementation of energy-aware MIB. The switch has the following attributes, `pmPowerIndex "1"`, `pmPhysicalEntity "2"`, and `pmPowerMonitorId "UUID 1000"`. The power usage of the switch is "440 Watts". The switch does not have a Power Monitor Parent.

The PoE switch port has the following attributes: The switch port has `pmPowerIndex "3"`, `pmPhysicalEntity` is "12" and `pmPowerMonitorId` is "UUID 1000:3". The power metered at the POE switch port is "12 watts". In this example, the POE switch port has the switch as the Power Monitor Parent, with its `pmParentID` of "1000".

The attributes of the PC are given below. The PC does not have implementation of Entity MIB, and thus does not have `pmPhysicalEntity`. The `pmPowerIndex` (`pmPIndex`) of the PC is "57", the `pmPowerMonitorId` is "UUID 1000:57 ". The PC has a Power Monitor Parent, i.e. the switch port whose

pmPowerMonitorId is "UUID 1000:3". The power usage of the PC is "120 Watts" and is communicated to the switch port.

This example illustrates the important distinction between the Power Monitor Children: The IP phone draws power from the switch, while the PC has LAN connectivity from the phone, but is powered from the wall outlet. However, the Power Monitor Parent sends power control messages to both the Power Monitor Children (IP phone and PC) and the Children react to those messages.



```

Internet-Draft          <Energy Monitoring MIB>          July 2011
=====
| 57          |      0          |  UUID 1000:57 |  UUID 1000:3 | 120          |
=====

```

Figure 1: Example scenario

9. Structure of the MIB

The primary MIB object in this MIB module is the PowerMonitorMIBObject. The pmPowerTable table of PowerMonitorMibObject describes the power measurement attributes of a Power Monitor entity. The notion of identity of the device in terms of uniquely identification of the Power Monitor and its relationship to other entities in the network are addressed in [EMAN-AWARE-MIB].

The power measurement of Power Monitor contains information describing its power usage (pmPower) and its current power state (pmPowerOperState). In addition to power usage, additional information describing the units of measurement (pmPowerAccuracy, pmPowerUnitMultiplier), how power usage measurement was obtained (pmPowerMeasurementCaliber), the source of power (pmPowerOrigin) and the type of power (pmPowerCurrentTtype) are described.

A Power Monitor may contain an optional pmPowerQuality table that describes the electrical characteristics associated with the current power state and usage.

A Power Monitor may contain an optional pmEnergyTable to describe energy measurement information over time.

A Power Monitor may also contain optional battery information associated with this entity.

10. MIB Definitions

```

-- *****
--
--
-- This MIB is used to monitor power usage of network

```

Internet-Draft <Energy Monitoring MIB> July 2011
-- devices
--
-- *****

POWER-MONITOR-MIB DEFINITIONS ::= BEGIN

IMPORTS

MODULE-IDENTITY,
OBJECT-TYPE,
NOTIFICATION-TYPE,
mib-2,
Integer32, Counter64, TimeTicks
FROM SNMPv2-SMI
TEXTUAL-CONVENTION, DisplayString, RowStatus, TimeInterval
FROM SNMPv2-TC
MODULE-COMPLIANCE, NOTIFICATION-GROUP, OBJECT-GROUP
FROM SNMPv2-CONF
OwnerString
FROM RMON-MIB;

powerMonitorMIB MODULE-IDENTITY

LAST-UPDATED "201107080000Z" -- 8 July 2011
ORGANIZATION "IETF EMAN Working Group"
CONTACT-INFO
"WG charter:
<http://datatracker.ietf.org/wg/eman/charter/>

Mailing Lists:

General Discussion: eman@ietf.org

To Subscribe:

<https://www.ietf.org/mailman/listinfo/eman>

Archive:

<http://www.ietf.org/mail-archive/web/eman>

Editors:

Mouli Chandramouli
Cisco Systems, Inc.
Sarjapur Outer Ring Road
Bangalore,
IN
Phone: +91 80 4426 3947
Email: moulchan@cisco.com

Brad Schoening
44 Rivers Edge Drive
Little Silver, NJ 07739

US
Email: brad@bradschoening.com

Juergen Quittek
NEC Europe Ltd.
NEC Laboratories Europe
Network Research Division
Kurfuersten-Anlage 36
Heidelberg 69115
DE
Phone: +49 6221 4342-115
Email: quittek@neclab.eu

Thomas Dietz
NEC Europe Ltd.
NEC Laboratories Europe
Network Research Division
Kurfuersten-Anlage 36
69115 Heidelberg
DE
Phone: +49 6221 4342-128
Email: Thomas.Dietz@nw.neclab.eu

Benoit Claise
Cisco Systems, Inc.
De Kleetlaan 6a b1
Degem 1831
Belgium
Phone: +32 2 704 5622
Email: bclaise@cisco.com"

DESCRIPTION
"This MIB is used to monitor power and energy in
devices."
REVISION
"201107080000Z" -- 8 July 2011
DESCRIPTION
"Initial version, published as RFC XXXX."

::= { mib-2 xxx }

powerMonitorMIBNotifs OBJECT IDENTIFIER
 ::= { powerMonitorMIB 0 }

powerMonitorMIBObjects OBJECT IDENTIFIER
 ::= { powerMonitorMIB 1 }

Internet-Draft <Energy Monitoring MIB>
powerMonitorMIBConform OBJECT IDENTIFIER
 ::= { powerMonitorMIB 2 }

July 2011

-- Textual Conventions

PowerStateSet ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"PowerStateSet is a TC that describes the Power State Set a Power Monitor supports. IANA has created a registry of Power State Sets supported by a Power Monitor entity and IANA shall administer the list of Power State Sets.

One byte is used to represent the Power State Set.

field	octets	contents	range
----	-----	-----	-----
1	1	Power State Set	1..255

Note:

the value of Power State Set in network byte order

1 in the first byte indicates IEEE1621 Power State Set

2 in the first byte indicates DMTF Power State Set

3 in the first byte indicates EMAN Power State Set"

REFERENCE

"<http://www.iana.org/assignments/eman>

RFC EDITOR NOTE: please change the previous URL if this is not the correct one after IANA assigned it."

SYNTAX OCTET STRING (SIZE(1))

UnitMultiplier ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

<Claise, et. Al>

Expires January 8, 2012

[Page 34]

Internet-Draft <Energy Monitoring MIB> July 2011
"The Unit Multiplier is an integer value that represents
the IEEE 61850 Annex A units multiplier associated with
the integer units used to measure the power or energy.

For example, when used with pmPowerUnitMultiplier, -3
represents 10⁻³ or milliwatts."

REFERENCE

"The International System of Units (SI),
National Institute of Standards and Technology,
Spec. Publ. 330, August 1991."

SYNTAX INTEGER {

yocto(-24), -- 10⁻²⁴
zepto(-21), -- 10⁻²¹
atto(-18), -- 10⁻¹⁸
femto(-15), -- 10⁻¹⁵
pico(-12), -- 10⁻¹²
nano(-9), -- 10⁻⁹
micro(-6), -- 10⁻⁶
milli(-3), -- 10⁻³
units(0), -- 10⁰
kilo(3), -- 10³
mega(6), -- 10⁶
giga(9), -- 10⁹
tera(12), -- 10¹²
peta(15), -- 10¹⁵
exa(18), -- 10¹⁸
zetta(21), -- 10²¹
yotta(24) -- 10²⁴

}

-- Objects

pmPowerTable OBJECT-TYPE

SYNTAX SEQUENCE OF PmPowerEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"This table lists Power Monitors."
 ::= { powerMonitorMIBObjects 1 }

pmPowerEntry OBJECT-TYPE

SYNTAX PmPowerEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"An entry describes the power usage of a Power Monitor."

Internet-Draft <Energy Monitoring MIB> July 2011
INDEX { pmPowerIndex, pmPowerStateSetIndex}
 ::= { pmPowerTable 1 }

```
PmPowerEntry ::= SEQUENCE {  
    pmPowerIndex                    Integer32,  
    pmPowerStateSetIndex           PowerStateSet,  
    pmPower                        Integer32,  
    pmPowerNameplate               Integer32,  
    pmPowerUnitMultiplier           UnitMultiplier,  
    pmPowerAccuracy                Integer32,  
    pmPowerMeasurementCaliber      INTEGER,  
    pmPowerCurrentType             INTEGER,  
    pmPowerOrigin                  INTEGER,  
    pmPowerAdminState              Integer32,  
    pmPowerOperState               Integer32,  
    pmPowerStateEnterReason        OwnerString  
}
```

pmPowerIndex OBJECT-TYPE

```
SYNTAX                    Integer32 (0..2147483647)  
MAX-ACCESS                not-accessible  
STATUS                    current  
DESCRIPTION
```

"A unique value, for each Power Monitor.
If an implementation of the ENERGY-AWARE-MIB module is available in the local SNMP context, then the same index as the one in the ENERGY-AWARE-MIB MUST be assigned for the identical Power Monitor. In this case, entities without an assigned value for pmIndex cannot be indexed by the pmPowerStateTable.

If there is no implementation of the ENERGY-AWARE-MIB module but one of the ENTITY MIB module is available in the local SNMP context, then the same index of an entity MUST be chosen as assigned to the entity by object entPhysicalIndex in the ENTITY MIB module. In this case, entities without an assigned value for entPhysicalIndex cannot be indexed by the pmPowerStateTable.

If neither the ENERGY-AWARE-MIB module nor of the ENTITY MIB module are available in the local SNMP context, then this MIB module may choose identity values from a further MIB module providing entity identities. In this case the value for each pmPowerIndex must remain constant at least from one re-initialization of the entity's network management system to the next re-initialization.

In case that no other MIB modules have been chosen for providing entity identities, Power States can be reported exclusively for the local device on which this table is instantiated. Then this table will have a single entry only and an index value of 0 MUST be used."

```
::= { pmPowerEntry 1 }
```

```
pmPowerStateSetIndex      OBJECT-TYPE
    SYNTAX                  PowerStateSet
    MAX-ACCESS              not-accessible
    STATUS                  current
    DESCRIPTION
        "This object indicates the Power State Set supported by
        the Power Monitor. The list of Power State Sets and
        their numbering are administered by IANA"
 ::= { pmPowerEntry 2 }
```

```
pmPower OBJECT-TYPE
    SYNTAX                  Integer32
    UNITS                   "Watts"
    MAX-ACCESS              read-only
    STATUS                  current
    DESCRIPTION
        "This object indicates the 'instantaneous' RMS
        consumption for the Power Monitor. This value is
        specified in SI units of watts with the magnitude of
        watts (milliwatts, kilowatts, etc.) indicated separately
        in pmPowerUnitMultiplier. The accuracy of the measurement
        is specified in pmPowerAccuracy. The direction of power
        flow is indicated by the sign on pmPower. If the Power
        Monitor is consuming power, the pmPower value will be
        positive. If the Power Monitor is producing power, the
        pmPower value will be negative.

        The pmPower MUST be less than or equal to the maximum
        power that can be consumed at the power state specified
        by pmPowerState.

        The pmPowerMeasurementCaliber object specifies how the
        usage value reported by pmPower was obtained. The pmPower
        value must report 0 if the pmPowerMeasurementCaliber is
        'unavailable'. For devices that can not measure or
        report power, this option can be used."
 ::= { pmPowerEntry 3 }
```

```
pmPowerNameplate OBJECT-TYPE
```

```

Internet-Draft          <Energy Monitoring MIB>          July 2011
SYNTAX                 Integer32
UNITS                  "Watts"
MAX-ACCESS             read-only
STATUS                 current
DESCRIPTION
    "This object indicates the rated maximum consumption for
    the fully populated Power Monitor.  The nameplate power
    requirements are the maximum power numbers and, in almost
    all cases, are well above the expected operational
    consumption.  The pmPowerNameplate is widely used for
    power provisioning.  This value is specified in either
    units of watts or voltage and current.  The units are
    therefore SI watts or equivalent Volt-Amperes with the
    magnitude (milliwatts, kilowatts, etc.) indicated
    separately in pmPowerUnitMultiplier."
 ::= { pmPowerEntry 4 }

pmPowerUnitMultiplier OBJECT-TYPE
SYNTAX                 UnitMultiplier
MAX-ACCESS             read-only
STATUS                 current
DESCRIPTION
    "The magnitude of watts for the usage value in pmPower
    and pmPowerNameplate."
 ::= { pmPowerEntry 5 }

pmPowerAccuracy OBJECT-TYPE
SYNTAX                 Integer32 (0..10000)
UNITS                  "hundredths of percent"
MAX-ACCESS             read-only
STATUS                 current
DESCRIPTION
    "This object indicates a percentage value, in 100ths of a
    percent, representing the assumed accuracy of the usage
    reported by pmPower.  For example: The value 1010 means
    the reported usage is accurate to +/- 10.1 percent.  This
    value is zero if the accuracy is unknown or not
    applicable based upon the measurement method.

    ANSI and IEC define the following accuracy classes for
    power measurement:
        IEC 62053-22 60044-1 class 0.1, 0.2, 0.5, 1 3.
        ANSI C12.20 class 0.2, 0.5"
 ::= { pmPowerEntry 6 }

pmPowerMeasurementCaliber OBJECT-TYPE
SYNTAX                 INTEGER {

```

```

        unavailable(1) ,
        unknown(2),
        actual(3) ,
        estimated(4),
        presumed(5)
    }
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION

```

"This object specifies how the usage value reported by pmPower was obtained:

- unavailable(1): Indicates that the usage is not available. In such a case, the pmPower value must be 0. For devices that can not measure or report power this option can be used.

- unknown(2): Indicates that the way the usage was determined is unknown. In some cases, entities report aggregate power on behalf of another device. In such cases it is not known whether the usage reported is actual(2), estimated(3) or presumed (4).

- actual(3): Indicates that the reported usage was measured by the entity through some hardware or direct physical means. The usage data reported is not presumed (4) or estimated (3) but the real apparent current energy consumption rate.

- estimated(4): Indicates that the usage was not determined by physical measurement. The value is a derivation based upon the device type, state, and/or current utilization using some algorithm or heuristic. It is presumed that the entity's state and current configuration were used to compute the value.

- presumed(5): Indicates that the usage was not determined by physical measurement, algorithm or derivation. The usage was reported based upon external tables, specifications, and/or model information. For example, a PC Model X draws 200W, while a PC Model Y draws 210W"

```
 ::= { pmPowerEntry 7 }
```

```

pmPowerCurrentType OBJECT-TYPE
    SYNTAX      INTEGER {
                    ac(1),
                    dc(2),

```

```
        unknown(3)
    }
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
        "This object indicates whether the pmUsage for the Power
        Monitor reports alternative current AC(1), direct current
        DC(2), or that the current type is unknown(3)."
```

::= { pmPowerEntry 8 }

```
pmPowerOrigin OBJECT-TYPE
    SYNTAX          INTEGER {
                    self (1),
                    remote (2)
                    }
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "This object indicates the source of power measurement
        and can be useful when modeling the power usage of
        attached devices. The power measurement can be performed
        by the entity itself or the power measurement of the
        entity can be reported by another trusted entity using a
        protocol extension. A value of self(1) indicates the
        measurement is performed by the entity, whereas remote(2)
        indicates that the measurement was performed by another
        entity."
    ::= { pmPowerEntry 9 }
```

```
pmPowerAdminState OBJECT-TYPE
    SYNTAX          Integer32 (1..65535)
    MAX-ACCESS      read-write
    STATUS          current
    DESCRIPTION
        "This object specifies the desired Power State for the
        Power Monitor, in the context of the Power State Set
        specified by pmPowerStateSetIndex in this table.
        Possible values of pmPowerAdminState are registered at
        IANA, per Power States Set. A current list of
        assignments can be found at
        <http://www.iana.org/assignments/eman>
        RFC-EDITOR: please check the location after IANA"
    ::= { pmPowerEntry 10 }
```

```
pmPowerOperState OBJECT-TYPE
    SYNTAX          Integer32 (1..65535)
    MAX-ACCESS      read-only
```

STATUS

current

DESCRIPTION

"This object specifies the current operational Power State for the Power Monitor, in the context of the Power State Set specified by pmPowerStateSetIndex in this table. Possible values of pmPowerOperState are registered at IANA, per Power States Set. A current list of assignments can be found at <<http://www.iana.org/assignments/eman>> RFC-EDITOR: please check the list"

::= { pmPowerEntry 11 }

pmPowerStateEnterReason OBJECT-TYPE

SYNTAX OwnerString

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"This string object describes the reason for the pmPowerAdminState transition. Alternatively, this string may contain with the entity that configured this Power Monitor to this Power State."

DEFVAL { "" }

::= { pmPowerEntry 12 }

pmPowerStateTable OBJECT-TYPE

SYNTAX SEQUENCE OF PmPowerStateEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table enumerates the maximum power usage, in watts, for every single supported Power State of each Power Monitor.

This table has an expansion-dependent relationship on the pmPowerTable, containing rows describing each Power State for the corresponding Power Monitor. For every Power Monitor in the pmPowerTable, there is a corresponding entry in this table."

::= { powerMonitorMIBObjects 2 }

pmPowerStateEntry OBJECT-TYPE

SYNTAX PmPowerStateEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A pmPowerStateEntry extends a corresponding pmPowerEntry. This entry displays max usage values at

every single possible Power State supported by the Power Monitor.

For example, given the values of a Power Monitor corresponding to a maximum usage of 11W at the state 1 (mechoff), 6 (ready), 8 (mediumMinus), 12 (High):

State	MaxUsage	Units
1 (mechoff)	0	W
2 (softoff)	0	W
3 (hibernate)	0	W
4 (sleep)	0	W
5 (standby)	0	W
6 (ready)	8	W
7 (lowMinus)	8	W
8 (low)	11	W
9 (medimMinus)	11	W
10 (medium)	11	W
11 (highMinus)	11	W
12 (high)	11	W

Furthermore, this table extends to return the total time in each Power State, along with the number of times a particular Power State was entered."

```

INDEX {
    pmPowerIndex,
    pmPowerStateSetIndex,
    pmPowerStateIndex
}
 ::= { pmPowerStateTable 1 }

PmPowerStateEntry ::= SEQUENCE {
    pmPowerStateIndex          Integer32,
    pmPowerStateMaxPower      Integer32,
    pmPowerStatePowerUnitMultiplier  UnitMultiplier,
    pmPowerStateTotalTime     TimeTicks,
    pmPowerStateEnterCount    Counter64
}

pmPowerStateIndex OBJECT-TYPE
    SYNTAX          Integer32 (1..65535)
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "This object specifies the Power State for the Power
        Monitor, in the context of the Power State Set specified
        by pmPowerStateSetIndex in this table."

```

This object specifies the index of the Power State of the Power Monitor within a Power State Set. The semantics of the specific Power State can be obtained from the Power State Set definition."

::= { pmPowerStateEntry 1 }

pmPowerStateMaxPower OBJECT-TYPE

SYNTAX Integer32

UNITS "Watts"

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object indicates the maximum power for the Power Monitor at the particular Power State. This value is specified in SI units of watts with the magnitude of the units (milliwatts, kilowatts, etc.) indicated separately in pmPowerStatePowerUnitMultiplier. If the maximum power is not known for a certain Power State, then the value is encoded as 0xFFFF.

For Power States not enumerated, the value of pmPowerStateMaxPower might be interpolated by using the next highest supported Power State."

::= { pmPowerStateEntry 3 }

pmPowerStatePowerUnitMultiplier OBJECT-TYPE

SYNTAX UnitMultiplier

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The magnitude of watts for the usage value in pmPowerStateMaxPower."

::= { pmPowerStateEntry 4 }

pmPowerStateTotalTime OBJECT-TYPE

SYNTAX TimeTicks

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object indicates the total time in hundreds of seconds that the Power Monitor has been in this power state since the last reset, as specified in the sysUpTime."

::= { pmPowerStateEntry 5 }

pmPowerStateEnterCount OBJECT-TYPE

SYNTAX Counter64

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object indicates how often the Power Monitor has entered this power state, since the last reset of the device as specified in the sysUpTime."

::= { pmPowerStateEntry 6 }

pmEnergyParametersTable OBJECT-TYPE

SYNTAX SEQUENCE OF PmEnergyParametersEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table is used to configure the parameters for Energy measurement collection in the table pmEnergyTable."

::= { powerMonitorMIBObjects 4 }

pmEnergyParametersEntry OBJECT-TYPE

SYNTAX PmEnergyParametersEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"An entry controls an energy measurement in pmEnergyTable."

INDEX { pmPowerIndex }

::= { pmEnergyParametersTable 1 }

PmEnergyParametersEntry ::= SEQUENCE {

pmEnergyParametersIntervalLength TimeInterval,

pmEnergyParametersIntervalNumber Integer32,

pmEnergyParametersIntervalMode Integer32,

pmEnergyParametersIntervalWindow TimeInterval,

pmEnergyParametersSampleRate Integer32,

pmEnergyParametersStatus RowStatus

}

pmEnergyParametersIntervalLength OBJECT-TYPE

SYNTAX TimeInterval

UNITS "Seconds"

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"This object indicates the length of time in seconds over which to compute the average pmEnergyIntervalEnergyUsed measurement in the pmEnergyTable table. The computation is based on the Power Monitor's internal sampling rate of

Internet-Draft <Energy Monitoring MIB> July 2011
power consumed or produced by the Power Monitor. The
sampling rate is the rate at which the power monitor can
read the power usage and may differ based on device
capabilities. The average energy consumption is then
computed over the length of the interval."

DEFVAL { 900 }
 ::= { pmEnergyParametersEntry 1 }

pmEnergyParametersIntervalNumber OBJECT-TYPE

SYNTAX Integer32
MAX-ACCESS read-create
STATUS current

DESCRIPTION

"The number of intervals maintained in the pmEnergyTable.
Each interval is characterized by a specific
pmEnergyIntervalStartTime, used as an index to the table
pmEnergyTable . Whenever the maximum number of entries is
reached, the measurement over the new interval replaces
the oldest measurement , except if the oldest measurement
were to be the maximum pmEnergyIntervalMax, in which case
the measurement the measurement over the next oldest
interval is replaced."

DEFVAL { 10 }
 ::= { pmEnergyParametersEntry 2 }

pmEnergyParametersIntervalMode OBJECT-TYPE

SYNTAX INTEGER {
period(1),
sliding(2),
total(3)
}
MAX-ACCESS read-create
STATUS current

DESCRIPTION

"A control object to define the mode of interval calculation
for the computation of the average
pmEnergyIntervalEnergyUsed measurement in the pmEnergyTable
table.

A mode of period(1) specifies non-overlapping periodic
measurements.

A mode of sliding(2) specifies overlapping sliding windows
where the interval between the start of one interval and
the next is defined in pmEnergyParametersIntervalWindow.

A mode of total(3) specifies non-periodic measurement. In
this mode only one interval is used as this is a

Internet-Draft <Energy Monitoring MIB> July 2011
continuous measurement since the last reset. The value of
pmEnergyParametersIntervalNumber should be (1) one and
pmEnergyParametersIntervalLength is ignored. "
 ::= { pmEnergyParametersEntry 3 }

pmEnergyParametersIntervalWindow OBJECT-TYPE

SYNTAX TimeInterval

UNITS "Seconds"

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The length of the duration window between the starting
time of one sliding window and the next starting time in
seconds, in order to compute the average
pmEnergyIntervalEnergyUsed measurement in the pmEnergyTable
table. This is valid only when the
pmEnergyParametersIntervalMode is sliding(2). The
pmEnergyParametersIntervalWindow value should be a multiple
of pmEnergyParametersSampleRate."

::= { pmEnergyParametersEntry 4 }

pmEnergyParametersSampleRate OBJECT-TYPE

SYNTAX Integer32

UNITS "Milliseconds"

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The sampling rate, in milliseconds, at which the Power
Monitor should poll power usage in order to compute the
average pmEnergyIntervalEnergyUsed measurement in the
table pmEnergyTable. The Power Monitor should initially
set this sampling rate to a reasonable value, i.e., a
compromise between intervals that will provide good
accuracy by not being too long, but not so short that
they affect the Power Monitor performance by requesting
continuous polling. If the sampling rate is unknown, the
value 0 is reported. The sampling rate should be selected
so that pmEnergyParametersIntervalWindow is a multiple of
pmEnergyParametersSampleRate."

DEFVAL { 1000 }

::= { pmEnergyParametersEntry 5 }

pmEnergyParametersStatus OBJECT-TYPE

SYNTAX RowStatus

MAX-ACCESS read-create

STATUS current

DESCRIPTION

Internet-Draft <Energy Monitoring MIB> July 2011
"The status of this row. The pmEnergyParametersStatus is used to start or stop energy usage logging. An entry status may not be active(1) unless all objects in the entry have an appropriate value. If this object is not equal to active(1), all associated usage-data logged into the pmEnergyTable will be deleted. The data can be destroyed by setting up the pmEnergyParametersStatus to destroy(2)."
 ::= { pmEnergyParametersEntry 6 }

pmEnergyTable OBJECT-TYPE
SYNTAX SEQUENCE OF PmEnergyIntervalEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
 "This table lists Power Monitor energy measurements. Entries in this table are only created if the corresponding value of object pmPowerMeasurementCaliber is active(2), i.e., if the power is actually metered."
 ::= { powerMonitorMIBObjects 5 }

pmEnergyIntervalEntry OBJECT-TYPE
SYNTAX PmEnergyIntervalEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
 "An entry describing energy measurements."
INDEX { pmPowerIndex, pmEnergyParametersIntervalMode,
pmEnergyIntervalStartTime }
 ::= { pmEnergyTable 1 }

PmEnergyIntervalEntry ::= SEQUENCE {
 pmEnergyIntervalStartTime TimeTicks,
 pmEnergyIntervalEnergyUsed Integer32,
 pmEnergyIntervalEnergyUnitMultiplier UnitMultiplier,
 pmEnergyIntervalMax Integer32,
 pmEnergyIntervalDiscontinuityTime TimeTicks
}

pmEnergyIntervalStartTime OBJECT-TYPE
SYNTAX TimeTicks
UNITS "hundredths of seconds"
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION

"The time (in hundredths of a second) since the network management portion of the system was last re-initialized, as specified in the sysUpTime [RFC3418]. This object is useful for reference of interval periods for which the energy is measured."

::= { pmEnergyIntervalEntry 1 }

pmEnergyIntervalEnergyUsed OBJECT-TYPE

SYNTAX Integer32
UNITS "Watt-hours"
MAX-ACCESS read-only
STATUS current

DESCRIPTION

"This object indicates the energy used in units of watt-hours for the Power Monitor over the defined interval. This value is specified in the common billing units of watt-hours with the magnitude of watt-hours (kW-Hr, MW-Hr, etc.) indicated separately in pmEnergyIntervalEnergyUnitMultiplier."

::= { pmEnergyIntervalEntry 2 }

pmEnergyIntervalEnergyUnitMultiplier OBJECT-TYPE

SYNTAX UnitMultiplier
MAX-ACCESS read-only
STATUS current

DESCRIPTION

"This object is the magnitude of watt-hours for the energy field in pmEnergyIntervalEnergyUsed."

::= { pmEnergyIntervalEntry 3 }

pmEnergyIntervalMax OBJECT-TYPE

SYNTAX Integer32
UNITS "Watt-hours"
MAX-ACCESS read-only
STATUS current

DESCRIPTION

"This object is the maximum energy ever observed in pmEnergyIntervalEnergyUsed since the monitoring started. This value is specified in the common billing units of watt-hours with the magnitude of watt-hours (kW-Hr, MW-Hr, etc.) indicated separately in pmEnergyIntervalEnergyUnits."

::= { pmEnergyIntervalEntry 4 }

pmEnergyIntervalDiscontinuityTime OBJECT-TYPE

SYNTAX TimeTicks
MAX-ACCESS read-only

```

Internet-Draft          <Energy Monitoring MIB>          July 2011
STATUS                 current
DESCRIPTION
    "The value of sysUpTime [RFC3418] on the most recent
    occasion at which any one or more of this entity's energy
    consumption counters suffered a discontinuity. If no such
    discontinuities have occurred since the last re-
    initialization of the local management subsystem, then
    this object contains a zero value."
 ::= { pmEnergyIntervalEntry 5 }

-- Notifications

pmPowerStateChange NOTIFICATION-TYPE
    OBJECTS             {pmPowerAdminState, pmPowerOperState,
pmPowerStateEnterReason}
    STATUS              current
    DESCRIPTION
        "The SNMP entity generates the PmPowerStateChange when
        the value(s) of pmPowerAdminState or pmPowerOperState,
        in the context of the Power State Set, have changed for
        the Power Monitor represented by the pmPowerIndex."
 ::= { powerMonitorMIBNotifs 1 }

-- Conformance

powerMonitorMIBCompliances OBJECT IDENTIFIER
 ::= { powerMonitorMIB 3 }

powerMonitorMIBGroups OBJECT IDENTIFIER
 ::= { powerMonitorMIB 4 }

powerMonitorMIBFullCompliance MODULE-COMPLIANCE
STATUS              current
DESCRIPTION
    "When this MIB is implemented with support for
    read-create, then such an implementation can
    claim full compliance. Such devices can then
    be both monitored and configured with this MIB."
MODULE              -- this module
MANDATORY-GROUPS {
    powerMonitorMIBTableGroup,
    powerMonitorMIBStateTableGroup,
    powerMonitorMIBEnergyTableGroup,
    powerMonitorMIBEnergyParametersTableGroup,
    powerMonitorMIBNotifGroup
}
 ::= { powerMonitorMIBCompliances 1 }

```

```

Internet-Draft          <Energy Monitoring MIB>          July 2011
powerMonitorMIBReadOnlyCompliance MODULE-COMPLIANCE
  STATUS                current
  DESCRIPTION
    "When this MIB is implemented without support for
    read-create (i.e. in read-only mode), then such an
    implementation can claim read-only compliance.  Such a
    device can then be monitored but can not be configured
    with this MIB."
  MODULE                -- this module
  MANDATORY-GROUPS {
    powerMonitorMIBTableGroup,
    powerMonitorMIBStateTableGroup,
    powerMonitorMIBNotifGroup
  }

  OBJECT                pmPowerOperState
  MIN-ACCESS            read-only
  DESCRIPTION
    "Write access is not required."
    ::= { powerMonitorMIBCompliances 2 }

-- Units of Conformance

powerMonitorMIBTableGroup OBJECT-GROUP
  OBJECTS              {
    pmPower,
    pmPowerNameplate,
    pmPowerUnitMultiplier,
    pmPowerAccuracy,
    pmPowerMeasurementCaliber,
    pmPowerCurrentType,
    pmPowerOrigin,
    pmPowerAdminState,
    pmPowerOperState,
    pmPowerStateEnterReason
  }
  STATUS                current
  DESCRIPTION
    "This group contains the collection of all the objects
    related to the PowerMonitor."
    ::= { powerMonitorMIBGroups 1 }

powerMonitorMIBStateTableGroup OBJECT-GROUP
  OBJECTS              {
    pmPowerStateMaxPower,
    pmPowerStatePowerUnitMultiplier,
    pmPowerStateTotalTime,
    pmPowerStateEnterCount
  }

```

```

    }
    STATUS current
    DESCRIPTION
        "This group contains the collection of all the
        objects related to the Power State."
    ::= { powerMonitorMIBGroups 2 }

```

```

powerMonitorMIBEnergyParametersTableGroup OBJECT-GROUP
    OBJECTS {
        pmEnergyParametersIntervalLength,
        pmEnergyParametersIntervalNumber,
        pmEnergyParametersIntervalMode,
        pmEnergyParametersIntervalWindow,
        pmEnergyParametersSampleRate,
        pmEnergyParametersStatus
    }
    STATUS current
    DESCRIPTION
        "This group contains the collection of all the objects
        related to the configuration of the Energy Table."
    ::= { powerMonitorMIBGroups 3 }

```

```

powerMonitorMIBEnergyTableGroup OBJECT-GROUP
    OBJECTS {
        -- Note that object
        -- pmEnergyIntervalStartTime is not
        -- included since it is not-accessible

        pmEnergyIntervalEnergyUsed,
        pmEnergyIntervalEnergyUnitMultiplier,
        pmEnergyIntervalMax,
        pmEnergyIntervalDiscontinuityTime
    }
    STATUS current
    DESCRIPTION
        "This group contains the collection of all the objects
        related to the Energy Table."
    ::= { powerMonitorMIBGroups 4 }

```

```

powerMonitorMIBNotifGroup NOTIFICATION-GROUP
    NOTIFICATIONS {
        pmPowerStateChange
    }
    STATUS current

```

```
Internet-Draft          <Energy Monitoring MIB>          July 2011
  DESCRIPTION
    "This group contains the notifications for the power and
    energy monitoring MIB Module."
    ::= { powerMonitorMIBGroups 5 }
```

```
END
```

```
-- *****
--
-- This MIB module is used to monitor power quality of networked
-- devices with measurements.
--
-- This MIB module is an extension of powerMonitorMIB module.
--
-- *****
```

```
POWER-QUALITY-MIB DEFINITIONS ::= BEGIN
```

```
IMPORTS
  MODULE-IDENTITY,
  OBJECT-TYPE,
  mib-2,
  Integer32
    FROM SNMPv2-SMI
  MODULE-COMPLIANCE,
  OBJECT-GROUP
    FROM SNMPv2-CONF
  UnitMultiplier, pmPowerIndex
    FROM POWER-MONITOR-MIB
  OwnerString
    FROM RMON-MIB;
```

```
powerQualityMIB MODULE-IDENTITY
```

```
  LAST-UPDATED      "201107080000Z"          -- 8 July 2011
  ORGANIZATION      "IETF EMAN Working Group"
  CONTACT-INFO
    "WG charter:
    http://datatracker.ietf.org/wg/eman/charter/

    Mailing Lists:
    General Discussion: eman@ietf.org

    To Subscribe:
    https://www.ietf.org/mailman/listinfo/eman
```

Archive:

<http://www.ietf.org/mail-archive/web/eman>

Editors:

Mouli Chandramouli
Cisco Systems, Inc.
Sarjapur Outer Ring Road
Bangalore,
IN
Phone: +91 80 4426 3947
Email: moulchan@cisco.com

Brad Schoening
44 Rivers Edge Drive
Little Silver, NJ 07739
US
Email: brad@bradschoening.com

Juergen Quittek
NEC Europe Ltd.
NEC Laboratories Europe
Network Research Division
Kurfuersten-Anlage 36
Heidelberg 69115
DE
Phone: +49 6221 4342-115
Email: quittek@neclab.eu

Thomas Dietz
NEC Europe Ltd.
NEC Laboratories Europe
Network Research Division
Kurfuersten-Anlage 36
69115 Heidelberg
DE
Phone: +49 6221 4342-128
Email: Thomas.Dietz@nw.neclab.eu

Benoit Claise
Cisco Systems, Inc.
De Kleetlaan 6a b1
Degem 1831
Belgium
Phone: +32 2 704 5622
Email: bclaise@cisco.com

DESCRIPTION

Internet-Draft <Energy Monitoring MIB> July 2011
"This MIB is used to report AC power quality in devices. The table is a sparse augmentation of the pmPowerTable table from the powerMonitorMIB module. Both three-phase and single-phase power configurations are supported."

REVISION

"201107080000Z" -- 8 July 2011

DESCRIPTION

"Initial version, published as RFC YYY."

::= { mib-2 yyy }

powerQualityMIBConform OBJECT IDENTIFIER

::= { powerQualityMIB 0 }

powerQualityMIBObjects OBJECT IDENTIFIER

::= { powerQualityMIB 1 }

-- Objects

pmACPwrQualityTable OBJECT-TYPE

SYNTAX SEQUENCE OF PmACPwrQualityEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table defines power quality measurements for supported pmPowerIndex entities. It is a sparse extension of the pmPowerTable."

::= { powerQualityMIBObjects 1 }

pmACPwrQualityEntry OBJECT-TYPE

SYNTAX PmACPwrQualityEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This is a sparse extension of the pmPowerTable with entries for power quality measurements or configuration. Each measured value corresponds to an attribute in IEC 61850-7-4 for non-phase measurements within the object MMUX."

INDEX { pmPowerIndex }

::= { pmACPwrQualityTable 1 }

```

PmACPwrQualityEntry ::= SEQUENCE {
    pmACPwrQualityConfiguration      INTEGER,
    pmACPwrQualityAvgVoltage         Integer32,
    pmACPwrQualityAvgCurrent         Integer32,
    pmACPwrQualityFrequency          Integer32,
    pmACPwrQualityPowerUnitMultiplier UnitMultiplier,
    pmACPwrQualityPowerAccuracy      Integer32,
    pmACPwrQualityTotalActivePower   Integer32,
    pmACPwrQualityTotalReactivePower Integer32,
    pmACPwrQualityTotalApparentPower Integer32,
    pmACPwrQualityTotalPowerFactor   Integer32,
    pmACPwrQualityThdAmperes         Integer32,
    pmACPwrQualityThdVoltage         Integer32
}

```

pmACPwrQualityConfiguration OBJECT-TYPE

```

SYNTAX INTEGER {
    snl(1),
    del(2),
    wye(3)
}

```

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Configuration describes the physical configurations of the power supply lines:

- * alternating current, single phase (SNGL)
- * alternating current, three phase delta (DEL)
- * alternating current, three phase Y (WYE)

Three-phase configurations can be either connected in a triangular delta (DEL) or star Y (WYE) system. WYE systems have a shared neutral voltage, while DEL systems do not. Each phase is offset 120 degrees to each other."

```
 ::= { pmACPwrQualityEntry 1 }
```

pmACPwrQualityAvgVoltage OBJECT-TYPE

```

SYNTAX      Integer32
UNITS       "0.1 Volt AC"
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION

```

"A measured value for average 'instantaneous' RMS line voltage. For a 3-phase system, this is the average voltage (V1+V2+V3)/3. IEC 61850-7-4 measured value attribute 'Vol'"

::= { pmACPwrQualityEntry 2 }

pmACPwrQualityAvgCurrent OBJECT-TYPE

SYNTAX Integer32
UNITS "Amperes"
MAX-ACCESS read-only
STATUS current
DESCRIPTION

"A measured value of the current per phase. IEC 61850-7-4 attribute 'Amp' "

::= { pmACPwrQualityEntry 3 }

pmACPwrQualityFrequency OBJECT-TYPE

SYNTAX Integer32 (4500..6500) -- UNITS 0.01 Hertz
UNITS "hertz"
MAX-ACCESS read-only
STATUS current
DESCRIPTION

"A measured value for the basic frequency of the AC circuit. IEC 61850-7-4 attribute 'Hz'."

::= { pmACPwrQualityEntry 4 }

pmACPwrQualityPowerUnitMultiplier OBJECT-TYPE

SYNTAX UnitMultiplier
MAX-ACCESS read-only
STATUS current
DESCRIPTION

"The magnitude of watts for the usage value in pmACPwrQualityTotalActivePower, pmACPwrQualityTotalReactivePower and pmACPwrQualityTotalApparentPower measurements. For 3-phase power systems, this will also include pmACPwrQualityPhaseActivePower, pmACPwrQualityPhaseReactivePower and pmACPwrQualityPhaseApparentPower"

::= { pmACPwrQualityEntry 5 }

pmACPwrQualityPowerAccuracy OBJECT-TYPE

SYNTAX Integer32 (0..10000)
UNITS "hundredths of percent"
MAX-ACCESS read-only
STATUS current
DESCRIPTION

"This object indicates a percentage value, in 100ths of a percent, representing the presumed accuracy of active, reactive, and apparent power usage reporting. For example: 1010 means the reported usage is accurate

Internet-Draft <Energy Monitoring MIB> July 2011
to +/- 10.1 percent. This value is zero if the
accuracy is unknown.

ANSI and IEC define the following accuracy classes for
power measurement: IEC 62053-22 & 60044-1 class 0.1,
0.2, 0.5, 1 & 3.

ANSI C12.20 class 0.2 & 0.5"
::= { pmACPwrQualityEntry 6 }

pmACPwrQualityTotalActivePower OBJECT-TYPE
SYNTAX Integer32
UNITS "RMS watts"
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"A measured value of the actual power delivered to or
consumed by the load. IEC 61850-7-4 attribute 'TotW'."
::= { pmACPwrQualityEntry 7 }

pmACPwrQualityTotalReactivePower OBJECT-TYPE
SYNTAX Integer32
UNITS "volt-amperes reactive"
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"A measured value of the reactive portion of the
apparent power. IEC 61850-7-4 attribute 'TotVAR'."
::= { pmACPwrQualityEntry 8 }

pmACPwrQualityTotalApparentPower OBJECT-TYPE
SYNTAX Integer32
UNITS "volt-amperes"
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"A measured value of the voltage and current which
determines the apparent power. The apparent power is
the vector sum of real and reactive power.

Note: watts and volt-amperes are equivalent units and
may be combined. IEC 61850-7-4 attribute 'TotVA'."
::= { pmACPwrQualityEntry 9 }

pmACPwrQualityTotalPowerFactor OBJECT-TYPE
SYNTAX Integer32 (-10000..10000)
UNITS "hundredths of percent"
MAX-ACCESS read-only
STATUS current

DESCRIPTION

"A measured value ratio of the real power flowing to the load versus the apparent power. It is dimensionless and expressed here as a percentage value in 100ths of a percent. A power factor of 100% indicates there is no inductance load and thus no reactive power. Power Factor can be positive or negative, where the sign should be in lead/lag (IEEE) form. IEC 61850-7-4 attribute 'TotPF'."

```
::= { pmACPwrQualityEntry 10 }
```

pmACPwrQualityThdAmperes OBJECT-TYPE

SYNTAX Integer32 (0..10000)
UNITS "hundredths of percent"
MAX-ACCESS read-only
STATUS current

DESCRIPTION

"A calculated value for the current total harmonic distortion (THD). Method of calculation is not specified. IEC 61850-7-4 attribute 'ThdAmp'."

```
::= { pmACPwrQualityEntry 11 }
```

pmACPwrQualityThdVoltage OBJECT-TYPE

SYNTAX Integer32 (0..10000)
UNITS "hundredths of percent"
MAX-ACCESS read-only
STATUS current

DESCRIPTION

"A calculated value for the voltage total harmonic distortion (THD). Method of calculation is not specified. IEC 61850-7-4 attribute 'ThdVol'."

```
::= { pmACPwrQualityEntry 12 }
```

pmACPwrQualityPhaseTable OBJECT-TYPE

SYNTAX SEQUENCE OF PmACPwrQualityPhaseEntry
MAX-ACCESS not-accessible
STATUS current

DESCRIPTION

"This table describes 3-phase power quality measurements. It is a sparse extension of the pmACPwrQualityTable."

```
::= { powerQualityMIBObjects 2 }
```

pmACPwrQualityPhaseEntry OBJECT-TYPE

SYNTAX PmACPwrQualityPhaseEntry
MAX-ACCESS not-accessible
STATUS current

DESCRIPTION

"An entry describes common 3-phase power quality measurements.

This optional table describes 3-phase power quality measurements, with three entries for each supported pmPowerIndex entity. Entities having single phase power shall not have any entities.

This table describes attributes common to both WYE and DEL. Entities having single phase power shall not have any entries here. It is a sparse extension of the pmACPwrQualityTable.

These attributes correspond to IEC 61850-7.4 MMXU phase measurements."

```
INDEX { pmPowerIndex, pmPhaseIndex }
 ::= { pmACPwrQualityPhaseTable 1 }
```

```
PmACPwrQualityPhaseEntry ::= SEQUENCE {
    pmPhaseIndex                Integer32,
    pmACPwrQualityPhaseAvgCurrent Integer32,
    pmACPwrQualityPhaseActivePower Integer32,
    pmACPwrQualityPhaseReactivePower Integer32,
    pmACPwrQualityPhaseApparentPower Integer32,
    pmACPwrQualityPhasePowerFactor Integer32,
    pmACPwrQualityPhaseImpedance Integer32
}
```

```
pmPhaseIndex OBJECT-TYPE
    SYNTAX          Integer32 (0..359)
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "A phase angle typically corresponding to 0, 120, 240."
    ::= { pmACPwrQualityPhaseEntry 1 }
```

```
pmACPwrQualityPhaseAvgCurrent OBJECT-TYPE
    SYNTAX          Integer32
    UNITS           "Amperes"
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "A measured value of the current per phase. IEC 61850-
        7-4 attribute 'A'"
    ::= { pmACPwrQualityPhaseEntry 2 }
```

```
pmACPwrQualityPhaseActivePower OBJECT-TYPE
    SYNTAX          Integer32
```

Internet-Draft <Energy Monitoring MIB> July 2011
UNITS "RMS watts"
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"A measured value of the actual power delivered to or
consumed by the load. IEC 61850-7-4 attribute 'W'"
::= { pmACPwrQualityPhaseEntry 3 }

pmACPwrQualityPhaseReactivePower OBJECT-TYPE
SYNTAX Integer32
UNITS "volt-amperes reactive"
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"A measured value of the reactive portion of the
apparent power. IEC 61850-7-4 attribute 'VAR'"
::= { pmACPwrQualityPhaseEntry 4 }

pmACPwrQualityPhaseApparentPower OBJECT-TYPE
SYNTAX Integer32
UNITS "volt-amperes"
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"A measured value of the voltage and current determines
the apparent power. Active plus reactive power equals
the total apparent power.

Note: Watts and volt-amperes are equivalent units and
may be combined. IEC 61850-7-4 attribute 'VA'."
::= { pmACPwrQualityPhaseEntry 5 }

pmACPwrQualityPhasePowerFactor OBJECT-TYPE
SYNTAX Integer32 (-10000..10000)
UNITS "hundredths of percent"
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"A measured value ratio of the real power flowing to
the load versus the apparent power for this phase. IEC
61850-7-4 attribute 'PF'. Power Factor can be positive
or negative where the sign should be in lead/lag (IEEE)
form."
::= { pmACPwrQualityPhaseEntry 6 }

pmACPwrQualityPhaseImpedance OBJECT-TYPE
SYNTAX Integer32
UNITS "volt-amperes"

```

Internet-Draft          <Energy Monitoring MIB>          July 2011
    MAX-ACCESS          read-only
    STATUS              current
    DESCRIPTION
    "A measured value of the impedance.  IEC 61850-7-4 attribute
    'Z'."
    ::= { pmACPwrQualityPhaseEntry 7 }

pmACPwrQualityDelPhaseTable OBJECT-TYPE
    SYNTAX              SEQUENCE OF PmACPwrQualityDelPhaseEntry
    MAX-ACCESS          not-accessible
    STATUS              current
    DESCRIPTION
    "This table describes DEL configuration phase-to-phase
    power quality measurements.  This is a sparse extension
    of the pmACPwrQualityPhaseTable."
    ::= { powerQualityMIBObjects 3 }

pmACPwrQualityDelPhaseEntry OBJECT-TYPE
    SYNTAX              PmACPwrQualityDelPhaseEntry
    MAX-ACCESS          not-accessible
    STATUS              current
    DESCRIPTION
    "An entry describes quality attributes of a phase in a
    DEL 3-phase power system.  Voltage measurements are
    provided both relative to each other and zero.

    Measured values are from IEC 61850-7-2 MMUX and THD from
    MHAI objects.

    For phase-to-phase measurements, the pmPhaseIndex is
    compared against the following phase at +120 degrees.
    Thus, the possible values are:

            pmPhaseIndex          Next Phase Angle
            0                      120
            120                    240
            240                    0
    "
    INDEX { pmPowerIndex, pmPhaseIndex}
    ::= { pmACPwrQualityDelPhaseTable 1}

PmACPwrQualityDelPhaseEntry ::= SEQUENCE {
    pmACPwrQualityDelPhaseToNextPhaseVoltage          Integer32,
    pmACPwrQualityDelThdPhaseToNextPhaseVoltage       Integer32,
    pmACPwrQualityDelThdCurrent                       Integer32
}

pmACPwrQualityDelPhaseToNextPhaseVoltage OBJECT-TYPE

```

```

Internet-Draft          <Energy Monitoring MIB>          July 2011
SYNTAX                  Integer32
UNITS                   "0.1 Volt AC"
MAX-ACCESS              read-only
STATUS                  current
DESCRIPTION
    "A measured value of phase to next phase voltages, where
    the next phase is IEC 61850-7-4 attribute 'PPV'."
 ::= { pmACPwrQualityDelPhaseEntry 2 }

pmACPwrQualityDelThdPhaseToNextPhaseVoltage OBJECT-TYPE
SYNTAX                  Integer32 (0..10000)
UNITS                   "hundredths of percent"
MAX-ACCESS              read-only
STATUS                  current
DESCRIPTION
    "A calculated value for the voltage total harmonic
    distortion for phase to next phase. Method of calculation
    is not specified. IEC 61850-7-4 attribute 'ThdPPV'."
 ::= { pmACPwrQualityDelPhaseEntry 3 }

pmACPwrQualityDelThdCurrent OBJECT-TYPE
SYNTAX                  Integer32 (0..10000)
UNITS                   "hundredths of percent"
MAX-ACCESS              read-only
STATUS                  current
DESCRIPTION
    "A calculated value for the voltage total harmonic
    distortion (THD) for phase to phase. Method of
    calculation is not specified.
    IEC 61850-7-4 attribute 'ThdPPV'."
 ::= { pmACPwrQualityDelPhaseEntry 4 }

pmACPwrQualityWyePhaseTable OBJECT-TYPE
SYNTAX                  SEQUENCE OF PmACPwrQualityWyePhaseEntry
MAX-ACCESS              not-accessible
STATUS                  current
DESCRIPTION
    "This table describes WYE configuration phase-to-neutral
    power quality measurements. This is a sparse extension
    of the pmACPwrQualityPhaseTable."
 ::= { powerQualityMIBObjects 4 }

pmACPwrQualityWyePhaseEntry OBJECT-TYPE
SYNTAX                  PmACPwrQualityWyePhaseEntry
MAX-ACCESS              not-accessible
STATUS                  current
DESCRIPTION

```

"This table describes measurements of WYE configuration with phase to neutral power quality attributes. Three entries are required for each supported pmPowerIndex entry. Voltage measurements are relative to neutral.

This is a sparse extension of the pmACPwrQualityPhaseTable.

Each entry describes quality attributes of one phase of a WYE 3-phase power system.

Measured values are from IEC 61850-7-2 MMUX and THD from MHAI objects."

```
INDEX { pmPowerIndex, pmPhaseIndex }
 ::= { pmACPwrQualityWyePhaseTable 1 }
```

```
PmACPwrQualityWyePhaseEntry ::= SEQUENCE {
    pmACPwrQualityWyePhaseToNeutralVoltage      Integer32,
    pmACPwrQualityWyePhaseCurrent              Integer32,
    pmACPwrQualityWyeThdPhaseToNeutralVoltage  Integer32
}
```

```
pmACPwrQualityWyePhaseToNeutralVoltage OBJECT-TYPE
    SYNTAX      Integer32
    UNITS       "0.1 Volt AC"
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "A measured value of phase to neutral voltage. IEC
        61850-7-4 attribute 'PhV'."
    ::= { pmACPwrQualityWyePhaseEntry 1 }
```

```
pmACPwrQualityWyePhaseCurrent OBJECT-TYPE
    SYNTAX      Integer32
    UNITS       "0.1 amperes AC"
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "A measured value of phase currents. IEC 61850-7-4
        attribute 'A'."
    ::= { pmACPwrQualityWyePhaseEntry 2 }
```

```
pmACPwrQualityWyeThdPhaseToNeutralVoltage OBJECT-TYPE
    SYNTAX      Integer32 (0..10000)
    UNITS       "hundredths of percent"
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
```

```

Internet-Draft          <Energy Monitoring MIB>          July 2011
    "A calculated value of the voltage total harmonic
    distortion (THD) for phase to neutral. IEC 61850-7-4
    attribute 'ThdPhV'."
    ::= { pmACPwrQualityWyePhaseEntry 3 }

-- Conformance

powerQualityMIBCompliances OBJECT IDENTIFIER
    ::= { powerQualityMIB 2 }

powerQualityMIBGroups OBJECT IDENTIFIER
    ::= { powerQualityMIB 3 }

powerQualityMIBFullCompliance MODULE-COMPLIANCE
    STATUS current
    DESCRIPTION
        "When this MIB is implemented with support for read-
        create, then such an implementation can claim full
        compliance. Such devices can then be both monitored and
        configured with this MIB."
    MODULE -- this module
    MANDATORY-GROUPS {
        powerACPwrQualityMIBTableGroup,
        powerACPwrQualityPhaseMIBTableGroup
    }

    GROUP powerACPwrQualityDelPhaseMIBTableGroup
    DESCRIPTION
        "This group must only be implemented for a DEL phase
        configuration."

    GROUP powerACPwrQualityWyePhaseMIBTableGroup
    DESCRIPTION
        "This group must only be implemented for a WYE phase
        configuration."
    ::= { powerQualityMIBCompliances 1 }

-- Units of Conformance

powerACPwrQualityMIBTableGroup OBJECT-GROUP
    OBJECTS {
        -- Note that object pmPowerIndex is NOT
        -- included since it is not-accessible
        pmACPwrQualityConfiguration,
        pmACPwrQualityAvgVoltage,
        pmACPwrQualityAvgCurrent,
        pmACPwrQualityFrequency,

```

```

    pmACPwrQualityPowerUnitMultiplier,
    pmACPwrQualityPowerAccuracy,
    pmACPwrQualityTotalActivePower,
    pmACPwrQualityTotalReactivePower,
    pmACPwrQualityTotalApparentPower,
    pmACPwrQualityTotalPowerFactor,
    pmACPwrQualityThdAmperes,
    pmACPwrQualityThdVoltage
}      STATUS      current

DESCRIPTION
    "This group contains the collection of all the power
    quality objects related to the Power Monitor."
 ::= { powerQualityMIBGroups 1 }

powerACPwrQualityPhaseMIBTableGroup OBJECT-GROUP
OBJECTS      {
    -- Note that object pmPowerIndex is NOT
    -- included since it is not-accessible
    pmACPwrQualityPhaseAvgCurrent,
    pmACPwrQualityPhaseActivePower,
    pmACPwrQualityPhaseReactivePower,
    pmACPwrQualityPhaseApparentPower,
    pmACPwrQualityPhasePowerFactor,
    pmACPwrQualityPhaseImpedance
}
STATUS      current
DESCRIPTION
    "This group contains the collection of all 3-phase power
    quality objects related to the Power State."
 ::= { powerQualityMIBGroups 2 }

powerACPwrQualityDelPhaseMIBTableGroup OBJECT-GROUP
OBJECTS      {
    -- Note that object pmPowerIndex and
    -- pmPhaseIndex are NOT included
    -- since they are not-accessible
    pmACPwrQualityDelPhaseToNextPhaseVoltage ,
    pmACPwrQualityDelThdPhaseToNextPhaseVoltage,
    pmACPwrQualityDelThdCurrent
}
STATUS      current
DESCRIPTION
    "This group contains the collection of all quality
    attributes of a phase in a DEL 3-phase power system."
 ::= { powerQualityMIBGroups 3 }

powerACPwrQualityWyePhaseMIBTableGroup OBJECT-GROUP

```

```

Internet-Draft      <Energy Monitoring MIB>      July 2011
OBJECTS             {
                    -- Note that object pmPowerIndex and
                    -- pmPhaseIndex are NOT included
                    -- since they are not-accessible
                    pmACPwrQualityWyePhaseToNeutralVoltage,
                    pmACPwrQualityWyePhaseCurrent,
                    pmACPwrQualityWyeThdPhaseToNeutralVoltage
                    }
STATUS              current
DESCRIPTION
    "This group contains the collection of all WYE
    configuration phase-to-neutral power quality
    measurements."
 ::= { powerQualityMIBGroups 4 }

```

END

11. Security Considerations

Some of the readable objects in these MIB modules (i.e., objects with a MAX-ACCESS other than not-accessible) may be considered sensitive or vulnerable in some network environments. It is thus important to control even GET and/or NOTIFY access to these objects and possibly to even encrypt the values of these objects when sending them over the network via SNMP.

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

- Unauthorized changes to the pmPowerOperState (via thepmPowerAdminState) MAY disrupt the power settings of the different Power Monitors, and therefore the state of functionality of the respective Power Monitors.
- Unauthorized changes to the pmEnergyParametersTable MAY disrupt energy measurement in the pmEnergyTable table.

SNMP versions prior to SNMPv3 did not include adequate security. Even if the network itself is secure (for example, by using IPsec), there is still no secure control over who on the secure network is allowed to access and GET/SET (read/change/create/delete) the objects in these MIB modules.

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

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

12. IANA Considerations

12.1. IANA Considerations for the MIB Modules

The MIB modules in this document uses the following IANA-assigned OBJECT IDENTIFIER values recorded in the SMI Numbers registry:

Descriptor -----	OBJECT IDENTIFIER value -----
PowerMonitorMIB	{ mib-2 xxx }
powerQualityMIB	{ mib-2 yyy }

Additions to the MIB modules are subject to Expert Review [RFC5226], i.e., review by one of a group of experts designated by an IETF Area Director. The group of experts MUST check the requested MIB objects for completeness and accuracy of the description. Requests for MIB objects that duplicate the functionality of existing objects SHOULD be declined. The smallest available OIDs SHOULD be assigned to the new MIB objects. The specification of new MIB objects SHOULD follow the structure specified in Section 10. and MUST be published using a well-established and persistent publication medium.

12.2. IANA Registration of new Power State Set

This document specifies an initial set of Power State Sets. The list of these Power State Sets with their numeric identifiers is given in Section 5.2.1. The Internet Assigned Numbers Authority (IANA) has created a new registry for Power State Sets numeric

Internet-Draft <Energy Monitoring MIB> July 2011
identifiers and filled it with the initial list as in Section
5.2.1. New Assignments to Power State Sets shall be
administered by IANA and the guidelines and procedures are
listed in this Section.

New assignments in Power State Sets require a Standards Action [RFC5226], i.e., they are to be made via Standards Track RFCs approved by the IESG. The new Power State Set based on the following guidelines; firstly check if there are devices or entities that have implementations of the proposed Power State Set or secondly, if the new Power State Set has been adopted or approved by the respective energy management standards organizations. A pure vendor specific implementation of Power State Set shall not be adopted; since it would lead to proliferation of Power State Sets.

12.2.1. IANA Registration of the IEEE1621 Power State Set

This document specifies a set of values for the IEEE1621 Power State Set [IEEE1621]. The list of these values with their identifiers is given in Section 5.2.1. The Internet Assigned Numbers Authority (IANA) created a new registry for IEEE1621 Power State Set identifiers and filled it with the initial list in Section 5.2.2.

New assignments (or potentially deprecation) for IEEE1621 Power State Set will be administered by IANA through Expert Review [RFC5226], i.e., review by one of a group of experts designated by an IETF Area Director. The group of experts MUST check the requested state for completeness and accuracy of the description.

12.2.2. IANA Registration of the DMTF Power State Set

This document specifies a set of values for the DMTF Power State Set. The list of these values with their identifiers is given in Section 5.2.1. The Internet Assigned Numbers Authority (IANA) has created a new registry for DMTF Power State Set identifiers and filled it with the initial list in Section 5.2.1.

New assignments (or potentially deprecation) for DMTF Power State Set will be administered by IANA through Expert Review [RFC5226], i.e., review by one of a group of experts designated by an IETF Area Director. The group of experts MUST check the conformance with the DMTF standard [DMTF], on the top of

12.2.3. IANA Registration of the EMAN Power State Set

This document specifies a set of values for the EMAN Power State Set. The list of these values with their identifiers is given in Section 5.2.1. The Internet Assigned Numbers Authority (IANA) has created a new registry for EMAN Power State Set identifiers and filled it with the initial list in Section 5.2.1.

New assignments (or potentially deprecation) for EMAN Power State Set New assignments in Power State Set require a Standards Action , i.e., they are to be made via Standards Track RFCs approved by the IESG.

12. Contributors

This document results from the merger of two initial proposals. The following persons made significant contributions either in one of the initial proposals or in this document.

John Parello

Rolf Winter

Dominique Dudkowski

13. Acknowledgment

The authors would like to thank Shamita Pisal for her prototype of this MIB module, and her valuable feedback. The authors would like to Michael Brown for improving the text dramatically.

14. Open Issues

OPEN ISSUE : double-check all the IEC references in the draft.

OPEN ISSUE: Description clause of pmPowerIndex Do we need this text Juergen Quittek to comment:

"The identity provisioning method that has been chosen can be retrieved by reading the value of powerStateEnergyConsumerOid. In case of identities provided by the ENERGY-AWARE-MIB module, this OID points to an existing instance of pmPowerIndex, in case of the ENTITY MIB, the object points to a valid instance of entPhysicalIndex, and in a similar way, it points to a value of another MIB module if this is used for identifying entities. If no other MIB module has been chosen for providing entity identities, then the value of powerStateEnergyConsumerOid MUST be 0.0 (zeroDotZero).

OPEN ISSUE : Juergen Schoenwalder review comments email May 25, 2011

PowerStateSeries ::= TEXTUAL-CONVENTION

Why is this an OCTET STRING (SIZE(1)) and not simply an enumerated INTEGER? And if this is to be maintained by IANA, why not create a IANA-POWER-SERIES-TC MIB module so that one can simply fetch the latest version from IANA?

New assignments in Power State Series require a Standards Action [RFC5226], i.e., they are to be made via Standards Track RFCs approved by the IESG.

This raises the bar pretty high. If some future organization defines popular power states, do you think someone is going to go through the trouble of producing a standards-track specification for this?

I also do not see why all objects in the pmPowerEntry are necessarily indexed by power series - some appear to me to be rather a property of the monitor and not the power state series the monitor happens to support.

Since I started looking at the IANA considerations, I believe this text needs to be removed:

OPEN ISSUE : Michael Schroff email comments Feb 24, 2011

TimeStamps for Power measurements

AC Power, Voltage, current measurement terminology

3-phase WYE or Delta or hybrid of WYE and Delta

Circuit breakers in scope of EMAN

Response sent to mailing list requesting for more information
and Clarification June 29, 2011.

15. References

15.2. Normative References

- [RFC2119] S. Bradner, Key words for use in RFCs to Indicate Requirement Levels, BCP 14, RFC 2119, March 1997.
- [RFC2578] McCloghrie, K., Ed., Perkins, D., Ed., and J. Schoenwaelder, Ed., "Structure of Management Information Version 2 (SMIv2)", STD 58, RFC 2578, April 1999.
- [RFC2579] McCloghrie, K., Ed., Perkins, D., Ed., and J. Schoenwaelder, Ed., "Textual Conventions for SMIv2", STD 58, RFC 2579, April 1999.
- [RFC2580] McCloghrie, K., Perkins, D., and J. Schoenwaelder, "Conformance Statements for SMIv2", STD 58, RFC 2580, April 1999.
- [RFC3621] Berger, A., and D. Romascanu, "Power Ethernet MIB", RFC3621, December 2003.
- [RFC4133] Bierman, A. and K. McCloghrie, "Entity MIB (Version 3)", RFC 4133, August 2005.
- [LLDP-MED-MIB] ANSI/TIA-1057, "The LLDP Management Information Base extension module for TIA-TR41.4 media endpoint discovery information", July 2005.
- [EMAN-AWARE-MIB] J. Parello, and B. Claise, "draft-ietf-eman-energy-aware-mib-02 ", work in progress, July 2011.

15.3. Informative References

- [RFC1628] S. Bradner, "UPS Management Information Base", RFC 1628, May 1994

- Internet-Draft <Energy Monitoring MIB> July 2011
- [RFC3410] Case, J., Mundy, R., Partain, D., and B. Stewart,
"Introduction and Applicability Statements for Internet
Standard Management Framework ", RFC 3410, December
2002.
- [RFC3418] Presun, R., Case, J., McCloghrie, K., Rose, M, and S.
Waldbusser, "Management Information Base (MIB) for the
Simple Network Management Protocol (SNMP)", RFC3418,
December 2002.
- [RFC3433] Bierman, A., Romascanu, D., and K. Norseth, "Entity
Sensor Management Information Base", RFC 3433, December
2002.
- [RFC4268] Chisholm, S. and D. Perkins, "Entity State MIB", RFC
4268, November 2005.
- [RFC5226] Narten, T. Alverstrand, H., A. and K. McCloghrie,
"Guidelines for Writing an IANA Considerations Section
in RFCs ", BCP 26, RFC 5226, May 2008.
- [EMAN-REQ] Quittek, J., Winter, R., Dietz, T., Claise, B., and
M. Chandramouli, " Requirements for Energy Management
", draft-ietf-eman-requirements-03 (work in
progress), July 2011. .
- [EMAN-FRAMEWORK] Claise, B., Parello, J., Schoening, B., and J.
Quittek, "Energy Management Framework", draft-ietf-
eman-framework-02 , July 2011.
- [EMAN-MONITORING-MIB] M. Chandramouli, Schoening, B., Dietz, T.,
Quittek, J. and B. Claise "Energy and Power Monitoring
MIB ", draft-claise-energy-monitoring-mib-09, July
2011.
- [EMAN-AS] Tychon, E., Laherty, M., and B. Schoening, "Energy
Management (EMAN) Applicability Statement", draft-
tychon-eman-applicability-statement-02, work in
progress, June 2011.
- [ACPI] "Advanced Configuration and Power Interface
Specification", <http://www.acpi.info/DOWNLOADS/ACPIspec30b.pdf>
- [DMTF] "Power State Management Profile DMTF DSP1027 Version
2.0" December 2009

[IEEE1621] "Standard for User Interface Elements in Power Control of Electronic Devices Employed in Office/Consumer Environments", IEEE 1621, December 2004.

[IEC.61850-7-4] International Electrotechnical Commission, "Communication networks and systems for power utility automation Part 7-4: Basic communication structure Compatible logical node classes and data object classes", 2010.

[IEC.62053-21] International Electrotechnical Commission, "Electricity metering equipment (a.c.) Particular requirements Part 22: Static meters for active energy (classes 1 and 2)", 2003.

[IEC.62053-22] International Electrotechnical Commission, "Electricity metering equipment (a.c.) Particular requirements Part 22: Static meters for active energy (classes 0,2 S and 0,5 S)", 2003.

Authors' Addresses

Mouli Chandramouli
Cisco Systems, Inc.
Sarjapur Outer Ring Road
Bangalore,
IN

Phone: +91 80 4426 3947
Email: moulchan@cisco.com

Brad Schoening
44 Rivers Edge Drive
Little Silver, NJ 07739
US
Email: brad@bradschoening.com

Juergen Quittek
NEC Europe Ltd.

Internet-Draft <Energy Monitoring MIB>
NEC Laboratories Europe
Network Research Division
Kurfuersten-Anlage 36
Heidelberg 69115
DE

July 2011

Phone: +49 6221 4342-115
Email: quittek@neclab.eu

Thomas Dietz
NEC Europe Ltd.
NEC Laboratories Europe
Network Research Division
Kurfuersten-Anlage 36
Heidelberg 69115
DE

Phone: +49 6221 4342-128
Email: Thomas.Dietz@neclab.eu

Benoit Claise
Cisco Systems, Inc.
De Kleetlaan 6a b1
Diegem 1813
BE

Phone: +32 2 704 5622
Email: bclaise@cisco.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 28, 2011

J. Quittek, Ed.
R. Winter
T. Dietz
D. Dudkowski
NEC Europe Ltd.
October 25, 2010

Definition of Managed Objects for Energy Management
draft-quittek-power-mib-02.txt

Abstract

This memo defines a portion of the Management Information Base (MIB) for use with network management protocols in the Internet community. In particular, it describes managed objects providing information about the energy consumption, the power states, and the battery status of managed devices and their components.

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79.

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/ietf/lid-abstracts.txt>.

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

This Internet-Draft will expire on April 28, 2011.

Copyright Notice

Copyright (c) 2010 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal

Provisions Relating to IETF Documents
(<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the BSD License.

Table of Contents

1. The Internet-Standard Management Framework	3
2. Introduction	3
3. Identifying Monitored Devices and Components	4
4. Power State MIB	5
4.1. Current Power State Table	5
4.2. Power State Table	6
5. Energy MIB	7
5.1. Energy Consumption Table	7
5.2. Energy Consumption Per Power State Table	8
5.3. Power History Table	9
6. Battery MIB	9
7. Relationship to Other MIB Modules	10
8. Definitions	11
8.1. Power State MIB	11
8.2. Energy MIB	20
8.3. Battery MIB	39
9. Security Considerations	51
10. IANA Considerations	52
11. References	52
11.1. Normative References	52
11.2. Informative References	53
Authors' Addresses	54

1. The Internet-Standard Management Framework

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

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. MIB objects are generally accessed through the Simple Network Management Protocol (SNMP). Objects in the MIB are defined using the mechanisms defined in the Structure of Management Information (SMI). This memo specifies MIB modules that is compliant to the SMIV2, which is described in STD 58, RFC 2578 [RFC2578], STD 58, RFC 2579 [RFC2579] and STD 58, RFC 2580 [RFC2580].

2. Introduction

Energy management in communication networks is a topic that has been neglected for many years when energy was cheap and global warming not recognized. This has changed recently. Energy management is becoming a significant component of network planning, operations and management and new energy management strategies are currently being explored.

An essential requirement for energy management is collecting information on energy consumption and energy storage at managed devices.

An elementary step into this direction is monitoring power states. A power state defines a limitations of services provided by a device and implicitly limits energy consumption. Examples for commonly implemented power states include 'on', 'full power', 'low power', 'sleep', 'stand-by', and 'off'. There is no commonly agreed convention for power states naming and semantics. Therefore power states with the same names may have different semantics and different names may be in use for the same power state.

But the actual energy consumption of a device depends on more than just its power state. Also the current load, the kind of load, and many other factors influence energy consumption. If instrumentation is available, it is very helpful to receive information on the actual energy consumption of a device and its component. Providing this information requires much more effort than reporting power states, because a probe that measures (electrical) power is required. Typically this means not just adding several lines of software to a device, but also adding costly sensor hardware to it.

A third aspect to be considered for energy management is energy storage in batteries. It is helpful, for example, to monitor which device is running on batteries and which is charging its battery. Fortunately, the problem of instrumentation is often an easy one for devices with rechargeable batteries. Controlling the charging cycles needs instrumentation anyway and this instrumentation can also be used for providing battery status information.

This document defines a portion of the Management Information Base (MIB) that serves the three purposes sketched above:

- o monitoring power states of managed entities,
- o monitoring energy consumption of managed entities,
- o monitoring the status of batteries contained in or controlled by managed devices.

Supporting all three monitoring task will not make sense for every device. Many networked devices do not have batteries to be monitored and thus it would not make sense for them to implement managed objects for this purpose.

As mentioned above, instrumentation for measuring actual energy consumption is relatively expensive and it will not make sense for every managed device to provide sufficient instrumentation. In such a case it would not be appropriate to still implement managed objects for energy consumption monitoring.

This leads to the conclusion that the portions of the MIB for the three monitoring tasks listed above should be rather independent of each other and not combined in a single one. This document contains three MIB modules called Power State MIB, Energy MIB, and Battery MIB. The Energy MIB module uses an object defined in the Power State MIB module, but beyond that there is no dependency between the three modules. Obviously, any combination of the three modules is possible.

The definitions in this document are based on the requirements outlined in [I-D.quittek-power-monitoring-requirements].

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 RFC 2119 [RFC2119].

3. Identifying Monitored Devices and Components

As argued in [I-D.quittek-power-monitoring-requirements] it is often required or at least desirable to not just monitor energy consumption and power state of an entire devices, but also of its contained

individual components. Furthermore it is argued in [I-D.quittek-power-monitoring-requirements] that there are cases where it is required that a managed device reports about energy consumption of one or more other, potentially remote devices. An example is a power strip reporting actual power and accumulated energy consumption of devices plugged into it.

It is not the purpose of MIB modules in this document to solve the problem of identifying components of the managed device that implements these modules or of components remote to this managed device. The task of identifying the entity that is subject of monitoring is left to other MIB modules, such as the ENERGY AWARE MIB module [I-D.parello-eman-energy-aware-mib], and the Entity MIB module [RFC4133].

As an open and flexible way of identifying the monitored entity, the MIB modules in this document use an OID as index that points into a MIB module used for identifying the monitored entity. For simplifying the trivial case that the monitored entity is identical with the device that implements the MIB module, an empty OID may be used.

4. Power State MIB

A number of devices today can operate in a number of different power states by reducing performance or going into standby mode or sleep mode. The Power State MIB module can be used for monitoring these states. Typically, not much instrumentation is needed for supporting the power state MIB module, because most devices with different power states are already equipped with means for controlling their these.

The Power State MIB module is structured into two tables, the powerCurrentStateTable reporting the current power state per entity and the powerStateTable providing statistics per power state. In addition, the Power State MIB module defines a notification that can be sent for informing the receiver about a change of an entity's current power state. For identifying the entity for which power state information is provided, OIDs are used, as explained in the previous section. Both tables use such an OID as their first index.

4.1. Current Power State Table

For basic monitoring of the actual power state of an entity, there is already a MIB module available: the Entity State MIB [RFC4268]. It reports the power state of an entity in object entStateStandby. It can have four different values: unknown(0), off(1), nonOperational(2), operational(3), see ENTITY-STATE-TC-MIB in

[RFC4268].

If this was considered to be sufficient, there would be no need for replicating this object in the power state MIB module. However, there is a concern that the three "known" states are too few for reflecting the variety of power saving states available today. For PCs, for example, there are several more states defined for the Advanced Configuration & Power Interface (ACPI). It might be useful to support several or all of these power states as suggested by [I-D.claise-energy-monitoring-mib].

The powerCurrentStateTable contains just a two objects per row:

```
powerStateTable(1)
+--powerStateEntry(1) [powerStateEnergyConsumerId]
  +-- --- Integer32      powerStateEnergyConsumerId(1)
  +-- --- ObjectIdentifier powerStateEnergyConsumerOid(2)
  +-- r-n SnmpAdminString powerStateOperationalState(3)
  +-- rwn SnmpAdminString powerStateAdminState(4)
```

Object powerStateOperationalState reports the actual power state of an entity at the time the object's value is retrieved. Object powerStateAdminState indicates a desired power state that the entity has been requested to enter, for example, by a network management system.

4.2. Power State Table

The second table called powerStateTable provides more detailed statistics for each power state. For this purpose it uses the power state name as another index object next to the entity index. This way, statistics can be reported per entity and per power state. The second index has the syntax of a SnmpAdminString and can be defined by the manufacturer of the device or MIB. In this way the index can fit many devices because the characteristics of the power state can be defined per device. The characteristics of the power state SHOULD be described as closely as possible in the object powerStateDescription.

```

powerStateAllStatesTable(2)
+--powerStateAllStatesEntry(1)
  [powerStateEnergyConsumerId,powerStateName]
  +-- --- SnmpAdminString powerStateName(1)
  +-- r-n Enumeration      powerStateType(2)
  +-- r-n SnmpAdminString powerStateDescription(3)
  +-- r-n Integer32        powerStateAveragePower(4)
  +-- r-n Integer32        powerStateMaximumPower(5)
  +-- r-n TimeTicks        powerStateTotalTime(6)
  +-- r-n TimeStamp        powerStateLastEnterTime(7)
  +-- r-n SnmpAdminString powerStateLastEnterReason(8)
  +-- r-n Counter64        powerStateEnterCount(9)

```

The offered statistics include the total time that the entity spent in a certain power state (`powerStateTotalTime`), the last time at which the entity entered a power state (`powerStateLastEnterTime`), the reason for entering it at the last time (`powerStateLastEnterReason`), the number of times a certain state has been entered (`powerStateEnterCount`), the average power consumed by the entity (`powerStateAveragePower`) and the maximum power consumed by the entity (`powerStateMaximumPower`).

5. Energy MIB

Devices that have instrumentation for measuring electrical energy consumption of entities can implement the Energy MIB module. Entities for which energy consumption is reported can be the entire devices, a component thereof or even an external entity for which the reporting devices observes the energy consumption.

The Energy MIB module defines two tables, the `energyTable` and the `energyPerStateTable`. The first one provides information on the instrumentations and on measured energy consumption of the entity. The second one provides energy consumption information for each individual power state.

5.1. Energy Consumption Table

The first set of managed objects in the `energyTable` are needed to help interpreting the energy consumption readings. These include the power supply type and voltage.

```

energyTable(1)
+--energyEntry(1) [energyConsumerId]
  +-- --- Integer32          energyConsumerId(1)
  +-- --- ObjectIdentifier  energyConsumerOid(2)
  +-- r-n EntitySensorStatus energySensorOperStatus(3)
  +-- r-n Unsigned32        energyNominalSupplyVoltage(4)
  +-- r-n Enumeration       energyElectricSupplyType(5)
  +-- r-n Unsigned32        energyTotalEnergy(6)
  +-- r-n UnitMultiplier    energyEnergyUnitMultiplier(7)
  +-- r-n Integer32         energyEnergyPrecision(8)
  +-- r-n Enumeration       energyMeasurementMethod(9)
  +-- r-n TimeStamp         energyDiscontinuityTime(10)
  +-- r-n Unsigned32        energySampleInterval(11)
  +-- r-n Unsigned32        energyMaxHistory(12)
  +-- r-n UnitMultiplier    energyPowerUnitMultiplier(13)
  +-- r-n Integer32         energyPowerPrecision(14)
  +-- r-n Unsigned32        energyRealPower(15)
  +-- r-n Unsigned32        energyPeakRealPower(16)
  +-- r-n Unsigned32        energyReactivePower(17)
  +-- r-n Unsigned32        energyApparentPower(18)
  +-- r-n Integer32         energyPhaseAngle(19)
  +-- r-n Integer32         energyPhaseAnglePrecision(20)

```

The main measured values provided by the table are the total energy consumed by the device and the current power (energy consumption rate). For entities supplied with alternating current (AC) there are also objects defined for reporting apparent power, reactive power and phase angle.

Provided energy and power values need to be multiplied by a unit multiplier given by a corresponding unit multiplier object in order to determine a measured value.

Measurements of the total energy consumed by an entity may suffer from interruptions in the continuous measurement of the current energy consumption. In order to indicate such interruptions, object `energyDiscontinuityTime` is provided for indicating the time of the last interruption of total energy measurement.

Time series of energy consumption values for past points in time are stored in the `energyHistoryTable`. Objects `energySampleInterval` and `energyMaxHistory` control the generation of entries in this table, see below.

5.2. Energy Consumption Per Power State Table

The second table in this module is called `energyPerStateTable` and it provides values of total energy consumption per power state in a way

similar to the powerStateTable in the Power State MIB module.

```
energyPerStateTable(2)
+--energyPerStateEntry(1) [energyConsumerId,powerStateName]
  +-- r-n Unsigned32 energyPerStateTotalEnergy(1)
```

5.3. Power History Table

The third table in this module is the energyHistoryTable. It stores total energy consumption values for past points in time.

```
energyHistoryTable(3)
+--energyHistoryEntry(1) [energyConsumerId,energyHistoryIndex]
  +-- --- Unsigned32 energyHistoryIndex(1)
  +-- r-n TimeStamp energyHistoryTimestamp(2)
  +-- r-n Unsigned32 energyHistoryTotalEnergy(3)
```

Creation of entries in this table is controlled by the values of corresponding objects energySampleInterval and energyMaxHistory in the energyTable.

Entries are indexed by the the entity (energyConsumerId) and by energyHistoryIndex. The first entry created for a certain entity in the table always has an energyHistoryIndex with a value of 1. Further entries for the same entity get increasing consecutive indices until the maximum index value given by object energyMaxHistory is reached. Then, no further indices will be used, but the entry with the oldest timestamp will be overwritten each time a new entry needs to be created.

A new entry is created with a time difference given by object energySampleInterval after creation of the previous entry. Hence, the difference between timestamps energyHistoryTimestamp of two consecutive entries SHOULD be equal to the value of object energySampleInterval.

6. Battery MIB

Editor's note: The Battery MIB module still uses the entPhysicalIndex from the ENTITY MIB. This will be changed in the next revision.

The third MIB module defined in this document defines objects for reporting information about batteries. The batteryTable contained in the Battery MIB module is again a sparse augment of the Entity MIB module [RFC4133]. It uses one row per battery and require that every battery for which information is provided has its own entry in the entPhysicalTable of the Entity MIB module.

The kind of entity in the entPhysicalTable is indicated by the value of enumeration object entPhysicalClass. Since there is no value called 'battery' defined for this object, it is RECOMMENDED that for batteries the value of this object is chosen to be powerSupply(6).

The batteryTable contains three groups of objects. The first group describes the battery in more detail than the generic objects in the entPhysicalTable. The second group of objects report on the current battery state, if it is charging or discharging, how much it is charged, its remaining capacity, the number of experienced charging cycles, etc.

```

batteryTable(1)
+--batteryEntry(1) [entPhysicalIndex]
  +-- r-n Enumeration batteryType(1)
  +-- r-n Enumeration batteryTechnology(2)
  +-- r-n Unsigned32 batteryNominalVoltage(3)
  +-- r-n Unsigned32 batteryNumberOfCells(4)
  +-- r-n Unsigned32 batteryNominalCapacity(5)
  +-- r-n Unsigned32 batteryRemainingCapacity(6)
  +-- r-n Counter32 batteryChargingCycleCount(7)
  +-- r-n DateAndTime batteryLastChargingCycleTime(8)
  +-- r-n Enumeration batteryState(9)
  +-- r-n Unsigned32 batteryCurrentCharge(10)
  +-- r-n Unsigned32 batteryCurrentChargePercentage(11)
  +-- r-n Unsigned32 batteryCurrentVoltage(12)
  +-- r-n Integer32 batteryCurrentCurrent(13)
  +-- r-n Unsigned32 batteryLowAlarmPercentage(14)
  +-- r-n Unsigned32 batteryLowAlarmVoltage(15)
  +-- r-n Unsigned32 batteryReplacementAlarmCapacity(16)
  +-- r-n Unsigned32 batteryReplacementAlarmCycles(17)

```

The third group of objects in this table indicates thresholds which can be used to raise an alarm if a property of the battery exceeds one of them. Raising an alarm may include sending a notification. The Battery MIB defines two notifications, one indicating a low battery charging state and one indicating an aged battery that may need to be replaced.

7. Relationship to Other MIB Modules

The three MIB modules described above relate to a number of existing standard MIB modules and complements them where necessary.

This section needs to be revised.

8. Definitions

8.1. Power State MIB

```
POWER-STATE-MIB DEFINITIONS ::= BEGIN

IMPORTS
    MODULE-IDENTITY, OBJECT-TYPE, NOTIFICATION-TYPE,
    mib-2, Integer32, Counter64, TimeTicks
        FROM SNMPv2-SMI -- RFC2578
    TimeStamp
        FROM SNMPv2-TC -- RFC2579
    MODULE-COMPLIANCE, OBJECT-GROUP, NOTIFICATION-GROUP
        FROM SNMPv2-CONF -- RFC2580
    SnmpAdminString
        FROM SNMP-FRAMEWORK-MIB; -- RFC3411

powerStateMIB MODULE-IDENTITY
    LAST-UPDATED "201010231200Z" -- 23 October 2010
    ORGANIZATION "IETF OPSAWG Working Group"
    CONTACT-INFO
        "General Discussion: opsawg@ietf.org
        To Subscribe: https://www.ietf.org/mailman/listinfo/opsawg
        Archive: http://www.ietf.org/mail-archive/web/opsawg

        Co-editor:
            Juergen Quittek
            NEC Europe Ltd.
            NEC Laboratories Europe
            Kurfuersten-Anlage 36
            69115 Heidelberg
            Germany
            Tel: +49 6221 4342-115
            Email: quittek@neclab.eu

        Co-editor:
            Thomas Dietz
            NEC Europe Ltd.
            NEC Laboratories Europe
            Kurfuersten-Anlage 36
            69115 Heidelberg
            Germany
            Phone: +49 6221 4342-128
            Email: Thomas.Dietz@neclab.eu"

DESCRIPTION
    "This MIB module defines a set of objects for monitoring
    the power state of managed entitites.
```

Copyright (c) 2010 IETF Trust and the persons identified as authors of the code. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, is permitted pursuant to, and subject to the license terms contained in, the Simplified BSD License set forth in Section 4.c of the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info).

This version of this MIB module is part of RFC yyyy; see the RFC itself for full legal notices."

-- replace yyyy with actual RFC number & remove this notice

-- Revision history

REVISION "201010231200Z" -- 23 October 2010
DESCRIPTION

"Initial version, published as RFC yyyy."

-- replace yyyy with actual RFC number & remove this notice

::= { mib-2 9991 }

-- xxx to be assigned by IANA.

--*****
-- Top Level Structure of the MIB module
--*****

powerStateNotifications OBJECT IDENTIFIER ::= { powerStateMIB 0 }
powerStateObjects OBJECT IDENTIFIER ::= { powerStateMIB 1 }
powerStateConformance OBJECT IDENTIFIER ::= { powerStateMIB 2 }

-- 1. Object Definitions

-- 1.1. Actual Power State Table

powerStateTable OBJECT-TYPE
SYNTAX SEQUENCE OF PowerStateEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"This table provides information on the current power state of managed entities.

The table is indexed by an ID of the entity on which power

state information is provided. IDs can be provided by another MIB module, such as the ENERGY AWARE MIB module or the ENTITY MIB module. If not ID provisioning from other MIB modules is available, the table can only have one entry for reporting the local power state of the device that tuns an instance of this table."

```
::= { powerStateObjects 1 }
```

```
powerStateEntry OBJECT-TYPE
```

```
SYNTAX          PowerStateEntry
```

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

```
STATUS         current
```

```
DESCRIPTION
```

"An entry providing information on the current power state of an entity."

```
INDEX { powerStateEnergyConsumerId }
```

```
::= { powerStateTable 1 }
```

```
PowerStateEntry ::=
```

```
SEQUENCE {
```

```
    powerStateEnergyConsumerId  Integer32,
```

```
    powerStateEnergyConsumerOid OBJECT IDENTIFIER,
```

```
    powerStateOperationalState  SnmpAdminString,
```

```
    powerStateAdminState        SnmpAdminString
```

```
}
```

```
powerStateEnergyConsumerId OBJECT-TYPE
```

```
SYNTAX          Integer32 (0..2147483647)
```

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

```
STATUS         current
```

```
DESCRIPTION
```

"An integer that identifies an entity that is subject of power state monitoring. Index values MUST be locally unique for each identified entity.

If an implementation of the ENERGY AWARE MIB module is available in the local SNMP context, then the same index of an entity MUST be chosen as assigned to the entity by object pmIndex in the ENERGY AWARE MIB module. In this case, entities without an assigned value for pmIndex cannot be indexed by the powerCurrentStateTable.

If there is no implementation of the ENERGY AWARE MIB module but one of the ENTITY MIB module is available in the local SNMP context, then the same index of an entity MUST be chosen as assigned to the entity by object entPhysicalIndex in the ENTITY MIB module. In this case, entities without an assigned value for pmIndex cannot be indexed by the

powerCurrentStateTable.

If neither the ENERGY AWARE MIB module nor of the ENTITY MIB module is available in the local SNMP context, then this MIB module may choose identity values from a further MIB module providing entity identities. In this case the value for each pmIndex must remain constant at least from one re-initialization of the entity's network management system to the next re-initialization.

In case that no other MIB module has been chosen for providing entity identities, power state can be reported exclusively for the local device on which this table is instantiated. Then this table will have a single entry only and an index value of 0 MUST be used.

The identity provisioning method that has been chosen can be retrieved by reading the value of powerStateEnergyConsumerOid. In case of identities provided by the ENERGY AWARE MIB module, this OID points to an existing instance of pmIndex, in case of the ENTITY MIB, the object points to a valid instance of entPhysicalIndex, and in a similar way, it points to a value of another MIB module if this is used for identifying entities. If no other MIB module has been chosen for providing entity identities, then the value of powerStateEnergyConsumerOid MUST be 0.0 (zeroDotZero)."

```
::= { powerStateEntry 1 }
```

powerStateEnergyConsumerOid OBJECT-TYPE

SYNTAX OBJECT IDENTIFIER

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"An OID that identifies an entity that is subject of power state monitoring. The value MUST be an OID that points to an existing managed object or 0.0 (zeroDotZero).

If another MIB module is chosen for providing identities for managed entities, then the value of this object points to an existing instance of an entity identifier, such as an instance of pmIndex in the ENERGY AWARE MIB or an instance of entPhysicalIndex in the ENTITY MIB module.

If power state information is provided only for the local device on which this table is instantiated, then the value of this object MUST be 0.0 (zeroDotZero)."

```
::= { powerStateEntry 2 }
```

```
powerStateOperationalState OBJECT-TYPE
    SYNTAX      SnmpAdminString (SIZE(1..32))
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "This object indicates the current power state of the
        entity. The given SnmpAdminString MUST match the
        powerStateName object of an entry in the
        powerStateAllStatesTable."
    ::= { powerStateEntry 3 }

powerStateAdminState OBJECT-TYPE
    SYNTAX      SnmpAdminString (SIZE(0..32))
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "This object indicates the desired power state of the
        entity. This object may be set by a network management
        system in order to request changing the actual power state
        to the desired one.

        If this object has not been set by an administrative action
        requesting a certain power state, then its value is an
        empty string of length 0."
    ::= { powerStateEntry 4 }

-----
-- 1.2. All Power States Table
-----

powerStateAllStatesTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF PowerStateAllStatesEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "This table provides information on all available power
        states of managed entities.

        The table extends the powerStateTable by sharing the first
        index. The first index serves for identifying an entity for
        which power state information is provided. The second index
        identifies a single power state by its name."
    ::= { powerStateObjects 2 }

powerStateAllStatesEntry OBJECT-TYPE
    SYNTAX      PowerStateAllStatesEntry
    MAX-ACCESS  not-accessible
    STATUS      current
```

DESCRIPTION

"Power state information about this physical entity."

```
INDEX      { powerStateEnergyConsumerId, powerStateName }
 ::= { powerStateAllStatesTable 1 }
```

```
PowerStateAllStatesEntry ::=
```

```
SEQUENCE {
    powerStateName          SnmpAdminString,
    powerStateType          INTEGER,
    powerStateDescription   SnmpAdminString,
    powerStateAveragePower  Integer32,
    powerStateMaximumPower  Integer32,
    powerStateTotalTime     TimeTicks,
    powerStateLastEnterTime TimeStamp,
    powerStateLastEnterReason SnmpAdminString,
    powerStateEnterCount    Counter64
}
```

```
powerStateName OBJECT-TYPE
```

```
SYNTAX      SnmpAdminString (SIZE(1..32))
```

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

```
STATUS      current
```

DESCRIPTION

"This index should only be created for power states that are actually implemented by the entity that is identified by the first index powerStateEnergyConsumerOid.

This index is the name of the power state and is limited to 32 characters.

If possible the name SHOULD already give a rough idea of the characteristic of this power state."

```
::= { powerStateAllStatesEntry 1 }
```

```
powerStateType OBJECT-TYPE
```

```
SYNTAX      INTEGER {
                unknown(0),
                off(1),
                nonOperational(2),
                operational(3)
            }
```

```
-----
-- Open issue: Shall we replace the syntax by textual convention
```

```
-- PowerMonitorLevel from draft-claise-energy-monitoring-mib?
-----
```

```
MAX-ACCESS  read-only
```

```
STATUS      current
```

DESCRIPTION

"Object classifies the power state. It helps to clearly distinguish non-operational power states (sleep, standby, etc.) from operational ones. In a nonOperational(2) state an entity provides non of its primary services except for bringing it into operational(3) states or off(1) states.

A device in state off(1) cannot report its state on its own. But state off(1) may be reported by managed devices reporting on the power state of other managed devices."

```
::= { powerStateAllStatesEntry 2 }
```

powerStateDescription OBJECT-TYPE

SYNTAX SnmpAdminString

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Power states are identified by their names. However, semantics of power states may vary between different entities. Reasons for variations can be different hardware and software architectures of managed devices.

Object powerStateDescription SHOULD describe the power state and its characteristics as closely as possible."

```
::= { powerStateAllStatesEntry 3 }
```

powerStateAveragePower OBJECT-TYPE

SYNTAX Integer32

UNITS "milliwatt"

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object indicates the average power (energy consumption rate) in milliwatt at the electrical power supply of the entity in the power state indicated by powerStateName.

A value of -1 indicates that the average power in this state is unknown."

```
::= { powerStateAllStatesEntry 4 }
```

powerStateMaximumPower OBJECT-TYPE

SYNTAX Integer32

UNITS "milliwatt"

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object indicates the maximum power (energy consumption rate) in milliwatt at the electrical power supply of the

entity in the power state indicated by powerStateName.

A value of -1 indicates that the maximum power in this state is unknown."

```
::= { powerStateAllStatesEntry 5 }
```

```
powerStateTotalTime OBJECT-TYPE
```

```
SYNTAX      TimeTicks
```

```
MAX-ACCESS  read-only
```

```
STATUS      current
```

```
DESCRIPTION
```

"This object indicates the total time in hundreds of seconds that the entity has been in the state indicated by index powerStateName."

```
::= { powerStateAllStatesEntry 6 }
```

```
-----
```

```
-- Open issue: Shall we use DateAndTime instead of timeTicks?
```

```
-----
```

```
powerStateLastEnterTime OBJECT-TYPE
```

```
SYNTAX      TimeStamp
```

```
MAX-ACCESS  read-only
```

```
STATUS      current
```

```
DESCRIPTION
```

"This time stamp object indicates the last time a which the entity entered the state indicated by index powerStateName."

```
::= { powerStateAllStatesEntry 7 }
```

```
powerStateLastEnterReason OBJECT-TYPE
```

```
SYNTAX      SnmpAdminString
```

```
MAX-ACCESS  read-only
```

```
STATUS      current
```

```
DESCRIPTION
```

"This string object describes the reason for the last power state transition into the power state indicated by index powerStateName."

```
::= { powerStateAllStatesEntry 8 }
```

```
powerStateEnterCount OBJECT-TYPE
```

```
SYNTAX      Counter64
```

```
MAX-ACCESS  read-only
```

```
STATUS      current
```

```
DESCRIPTION
```

"This object indicates how often the entity indicated by index entPhysicalIndex entered the power state indicated by index powerStateName."

```
::= { powerStateAllStatesEntry 9 }
```

```
=====
-- 2. Notifications
=====

powerStateChangeEvent NOTIFICATION-TYPE
  OBJECTS      { powerStateLastEnterReason }
  STATUS       current
  DESCRIPTION
    "This notification can be generated when the power state of
    an entity changes.

    Note that the state that has been entered is indicated by
    the OID of object powerStateLastEnterReason."
  ::= { powerStateNotifications 1 }

-----

-- 3. Conformance Information
-----

powerStateCompliances OBJECT IDENTIFIER
  ::= { powerStateConformance 1 }
powerStateGroups      OBJECT IDENTIFIER
  ::= { powerStateConformance 2 }

-----

-- 3.1. Compliance Statements
-----

powerCompliance MODULE-COMPLIANCE
  STATUS       current
  DESCRIPTION
    "The compliance statement for implementations of the
    POWER-STATE-MIB module.

    A compliant implementation MUST implement the objects
    defined in the mandatory group powerRequiredGroup."
  MODULE      -- this module
  MANDATORY-GROUPS { powerStateRequiredGroup }
  GROUP       powerStateNotificationsGroup
  DESCRIPTION
    "A compliant implementation does not have to implement
    the powerNotificationsGroup."
  ::= { powerStateCompliances 1 }

-----

-- 3.2. MIB Grouping
-----
```

```

powerStateRequiredGroup OBJECT-GROUP
  OBJECTS {
    powerStateOperationalState,
    powerStateAdminState,
    powerStateType,
    powerStateDescription,
    powerStateTotalTime,
    powerStateLastEnterTime,
    powerStateLastEnterReason,
    powerStateEnterCount,
    powerStateAveragePower,
    powerStateMaximumPower
  }
  STATUS      current
  DESCRIPTION
    "A compliant implementation MUST implement the objects
    contained in this group."
  ::= { powerStateGroups 1 }

powerStateNotificationsGroup NOTIFICATION-GROUP
  NOTIFICATIONS { powerStateChangeEvent }
  STATUS      current
  DESCRIPTION
    "A compliant implementation does not have to implement the
    notification contained in this group."
  ::= { powerStateGroups 2 }
END

```

8.2. Energy MIB

```

ENERGY-MIB DEFINITIONS ::= BEGIN

IMPORTS
  MODULE-IDENTITY, OBJECT-TYPE, mib-2,
  Unsigned32, Integer32
    FROM SNMPv2-SMI -- RFC2578
  TimeStamp
    FROM SNMPv2-TC -- RFC2579
  MODULE-COMPLIANCE, OBJECT-GROUP
    FROM SNMPv2-CONF -- RFC2580
  EntitySensorStatus
    FROM ENTITY-SENSOR-MIB -- RFC3433
  powerStateName
    FROM POWER-STATE-MIB
  UnitMultiplier
    FROM POWER-MONITOR-MIB;

energyMIB MODULE-IDENTITY

```

LAST-UPDATED "201010231200Z" -- 23 October 2010
ORGANIZATION "IETF OPSAWG Working Group"
CONTACT-INFO
"General Discussion: opsawg@ietf.org
To Subscribe: <https://www.ietf.org/mailman/listinfo/opsawg>
Archive: <http://www.ietf.org/mail-archive/web/opsawg>

Co-editor:

Juergen Quittek
NEC Europe Ltd.
NEC Laboratories Europe
Kurfuersten-Anlage 36
69115 Heidelberg
Germany
Tel: +49 6221 4342-115
Email: quittek@neclab.eu

Co-editor:

Thomas Dietz
NEC Europe Ltd.
NEC Laboratories Europe
Kurfuersten-Anlage 36
69115 Heidelberg
Germany
Phone: +49 6221 4342-128
Email: Thomas.Dietz@neclab.eu

DESCRIPTION

"This MIB module defines a set of objects for monitoring the energy consumption of networked devices and their components.

Copyright (c) 2010 IETF Trust and the persons identified as authors of the code. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, is permitted pursuant to, and subject to the license terms contained in, the Simplified BSD License set forth in Section 4.c of the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>).

This version of this MIB module is part of RFC yyyy; see the RFC itself for full legal notices."

-- replace yyyy with actual RFC number & remove this notice

-- Revision history

```
REVISION      "201010231200Z"          -- 23 October 2010
DESCRIPTION
  "Initial version, published as RFC yyyy."
-- replace yyyy with actual RFC number & remove this notice

 ::= { mib-2 9992 }
-- yyy to be assigned by IANA.

--*****
-- Top Level Structure of the MIB module
--*****

energyObjects      OBJECT IDENTIFIER ::= { energyMIB 1 }
energyConformance OBJECT IDENTIFIER ::= { energyMIB 2 }

-----
-- 1. Object Definitions
-----

-----
-- 1.1. Energy Consumption Table
-----

energyTable OBJECT-TYPE
  SYNTAX      SEQUENCE OF EnergyEntry
  MAX-ACCESS  not-accessible
  STATUS      current
  DESCRIPTION
    "This table provides information on the current and
    accumulated energy consumption of entities.

    The table is indexed by an ID of the entity on which
    energy information is provided. IDs can be provided by
    another MIB module, such as the ENERGY AWARE MIB module
    or the ENTITY MIB module. If not ID provisioning from
    other MIB modules is available, the table can only have
    one entry for reporting the local power state of the
    device that runs an instance of this table."
  ::= { energyObjects 1 }

energyEntry OBJECT-TYPE
  SYNTAX      EnergyEntry
  MAX-ACCESS  not-accessible
  STATUS      current
  DESCRIPTION
    "An entry providing information on the energy consumption
    of a physical entity."
  INDEX      { energyConsumerId }
  ::= { energyTable 1 }
```

```

EnergyEntry ::=
  SEQUENCE {
    energyConsumerId          Integer32,
    energyConsumerOid        OBJECT IDENTIFIER,
    energySensorOperStatus   EntitySensorStatus,
    energyNominalSupplyVoltage Unsigned32,
    energyElectricSupplyType INTEGER,
    energyTotalEnergy        Unsigned32,
    energyEnergyUnitMultiplier UnitMultiplier,
    energyEnergyPrecision    Integer32,
    energyMeasurementMethod  INTEGER,
    energyDiscontinuityTime  TimeStamp,
    energySampleInterval     Unsigned32,
    energyMaxHistory         Unsigned32,
    energyPowerUnitMultiplier UnitMultiplier,
    energyPowerPrecision    Integer32,
    energyRealPower          Unsigned32,
    energyPeakRealPower      Unsigned32,
    energyReactivePower      Unsigned32,
    energyApparentPower      Unsigned32,
    energyPhaseAngle         Integer32,
    energyPhaseAnglePrecision Integer32
  }

```

energyConsumerId OBJECT-TYPE

SYNTAX Integer32 (0..2147483647)

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"An integer that identifies an entity that is subject of energy monitoring. Index values MUST be locally unique for each identified entity.

If an implementation of the ENERGY AWARE MIB module is available in the local SNMP context, then the same index of an entity MUST be chosen as assigned to the entity by object pmIndex in the ENERGY AWARE MIB module. In this case, entities without an assigned value for pmIndex cannot be indexed by the powerCurrentStateTable.

If there is no implementation of the ENERGY AWARE MIB module but one of the ENTITY MIB module is available in the local SNMP context, then the same index of an entity MUST be chosen as assigned to the entity by object entPhysicalIndex in the ENTITY MIB module. In this case, entities without an assigned value for pmIndex cannot be indexed by the powerCurrentStateTable.

If neither the ENERGY AWARE MIB module nor of the ENTITY MIB module is available in the local SNMP context, then this MIB module may choose identity values from a further MIB module providing entity identities. In this case the value for each pmIndex must remain constant at least from one re-initialization of the entity's network management system to the next re-initialization.

In case that no other MIB module has been chosen for providing entity identities, power state can be reported exclusively for the local device on which this table is instantiated. Then this table will have a single entry only and an index value of 0 MUST be used.

The identity provisioning method that has been chosen can be retrieved by reading the value of object powerStateEnergyConsumerOid. In case of identities provided by the ENERGY AWARE MIB module, this OID points to an existing instance of pmIndex, in case of the ENTITY MIB, the object points to a valid instance of entPhysicalIndex, and in a similar way, it points to a value of another MIB module if this is used for identifying entities. If no other MIB module has been chosen for providing entity identities, then the value of powerStateEnergyConsumerOid MUST be 0.0 (zeroDotZero)."

```
::= { energyEntry 1 }
```

```
energyConsumerOid OBJECT-TYPE
```

```
SYNTAX          OBJECT IDENTIFIER
```

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

```
STATUS         current
```

```
DESCRIPTION
```

```
"An OID that identifies an entity that is subject of energy monitoring. The value MUST be an OID that points to an existing managed object or 0.0 (zeroDotZero)."
```

If another MIB module is chosen for providing identities for managed entities, then the value of this object points to an existing instance of an entity identifier, such as an instance of pmIndex in the ENERGY AWARE MIB or an instance of entPhysicalIndex in the ENTITY MIB module.

If power state information is provided only for the local device on which this table is instantiated, then the value of this object MUST be 0.0 (zeroDotZero)."

```
::= { energyEntry 2 }
```

```
energySensorOperStatus OBJECT-TYPE
```

```
SYNTAX      EntitySensorStatus
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "This object provides the operational status of the
    sensor that is used for measuring the energy consumption
    of the entity indicated by energyConsumerId."
 ::= { energyEntry 3 }
```

energyNominalSupplyVoltage OBJECT-TYPE

```
SYNTAX      Unsigned32
UNITS       "millivolt"
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "This object provides the nominal voltage of the power
    supply of the entity. It is provided in units of
    millivolt (mV).

    The nominal voltage actual of an entity is assumed to be
    fixed, while the actual power supply voltage may vary over
    time, for example, caused by changing load conditions.

    A value of 0 indicates that the nominal supply voltage
    is unknown."
 ::= { energyEntry 4 }
```

energyElectricSupplyType OBJECT-TYPE

```
SYNTAX      INTEGER {
                alternatingCurrent(1),
                directCurrent(2),
                unknown(3)
            }
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "This object indicates the type of electrical power
    supply for the entity. It is used for distinguishing
    between alternating current (AC) supply and direct
    current (DC) supply."
 ::= { energyEntry 5 }
```

energyTotalEnergy OBJECT-TYPE

```
SYNTAX      Unsigned32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "This object indicates the total consumed energy measured
```

at the electrical power supply of the entity.

In order to determine the measured value in watt hours, the value of this object needs to be multiplied by a unit multiplier given by the value of object energyEnergyUnitMultiplier.

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of energyDiscontinuityTime."

```
::= { energyEntry 6 }
```

energyEnergyUnitMultiplier OBJECT-TYPE

SYNTAX UnitMultiplier

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object provides unit multiplier for measured energy values. Reported values need to be multiplied with this multiplier in order to determine the measured value in watt hours.

This object serves as unit multiplier for objects energyTotalEnergy, energyPSTotalEnergy,

..."

```
::= { energyEntry 7 }
```

energyEnergyPrecision OBJECT-TYPE

SYNTAX Integer32 (0..10000)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object indicates the precision of a measured energy value. The precision is indicated as a percentage value, in 100ths of a percent. A value of 0 indicates that the precision is unknown or not applicable to the measured value.

This object serves precision indicator for the values provided by objects energyTotalEnergy, energyPSTotalEnergy, ..."

```
::= { energyEntry 8 }
```

energyMeasurementMethod OBJECT-TYPE

```
SYNTAX INTEGER {
    directEnergyMeasurement(1),
    powerOversampling(2),
```

```

        powerSampling(3),
        loadBasedEstimation(4),
        deviceBasedEstimation(5),
        unknown(6)
    }
MAX-ACCESS read-only
STATUS current
DESCRIPTION
    "This object indicates the method used for measuring energy
    consumption. A device may not be equipped with capabilities
    to measure its energy consumption directly, but rather
    relies on other input in order to conduct more or less
    precise estimations of its power consumption.

    The measurement methods concerns values of objects
    energyTotalEnergy, energyPSTotalEnergy, and
    energyPowerHistoryAverageValue.

    Five different measurement methods are specified.

    - directEnergyMeasurement(1) indicates that the entity is
      instrumented to directly measure its energy consumption.

    - powerOversampling(2) indicates that energy is measured
      by sampling power values more frequently than indicated
      by the value of object energySampleInterval.

    - powerSampling(3) indicates that energy is measured
      by sampling power values according to the value of object
      energySampleInterval.

    - loadBasedEstimation(4) indicates that power is estimated
      based on measurements of the load of the entity.

    - deviceBasedEstimation(5) indicates that power is estimated
      based on static properties of the entity. In this case,
      reported power only depends on the power state of the
      devices as indicated by object powerCurrentState in the
      powerCurrentStateTable of the Power State MIB module."
 ::= { energyEntry 9 }

```

energyDiscontinuityTime OBJECT-TYPE

```

SYNTAX TimeStamp
MAX-ACCESS read-only
STATUS current
DESCRIPTION
    "The value of sysUpTime on the most recent occasion at which
    any one or more of this entity's energy consumption counters

```

suffered a discontinuity. The relevant counters are energyTotalEnergy and energyPerStateTotalEnergy. If no such discontinuities have occurred since the last re-initialization of the local management subsystem, then this object contains a zero value."

::= { energyEntry 10 }

energySampleInterval OBJECT-TYPE

SYNTAX Unsigned32
UNITS "milliseconds"
MAX-ACCESS read-only
STATUS current
DESCRIPTION

"This object indicates is the difference of time stamps between two consecutive entries in the energyHistoryTable for this entity.

The interval length provided by this object indicates the or maximum interval length (or minimal sampling rate) at which the power sensor measures values of the current power. Implementations of the Energy MIB module may choose higher sampling rates (or shorter sampling intervals) in order to provide higher precision of the measurement. Preferably, shorter intervals may be chosen such that the sampling interval indicated by this object is a multiple of the actual sampling interval.

The sampling interval is provided in units of microseconds.

A value of 0 indicates that the sampling interval applied by the sensor is unknown or not constant."

::= { energyEntry 11 }

energyMaxHistory OBJECT-TYPE

SYNTAX Unsigned32
MAX-ACCESS read-only
STATUS current
DESCRIPTION

"This object indicates is the maximum number of corresponding entries in the energyPowerHistoryTable. An entry in the energyHistoryTable is corresponding if it has the same value for object energyConsumerId as index.

An implementation of the Energy MIB module will remove the oldest corresponding entry in the energyHistoryTable to allow the addition of a new entry once the number of corresponding entries in the energyHistoryTable

reaches this value.

Entries are added to the energyHistoryTable until energyMaxHistory is reached before entries begin to be removed.

A value of 0 for this object disables creation of corresponding energyHistoryTable entries."

```
DEFVAL      { 0 }
 ::= { energyEntry 12 }
```

energyPowerUnitMultiplier OBJECT-TYPE

```
SYNTAX      UnitMultiplier
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
```

"This object provides unit multiplier for measured energy values. Reported values need to be multiplied with this multiplier in order to determine the measured value in watt hours.

This object serves as unit multiplier for the values provided by objects energyRealPower, energyPeakRealPower, energyReactivePower, and energyApparentPower."

```
::= { energyEntry 13 }
```

energyPowerPrecision OBJECT-TYPE

```
SYNTAX      Integer32 (0..10000)
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
```

"This object indicates a the precision of a measured power value. The precision is indicated as a percentage value, in 100ths of a percent. A value of 0 indicates that the precision is unknown or not applicable to the measured value.

This object serves precision indicator for the values provided by objects energyRealPower, energyPeakRealPower, energyReactivePower, and energyApparentPower."

```
::= { energyEntry 14 }
```

energyRealPower OBJECT-TYPE

```
SYNTAX      Unsigned32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
```

"This object indicates the current real power value

at the electrical supply of the entity indicated by index energyConsumerId.

In order to determine the measured value in watts, the value of this object needs to be multiplied by a unit multiplier given by the value of object energyEnergyUnitMultiplier.

Measured values of this object are stored in the energyPowerTable with a rate determined by object energySampleInterval."

::= { energyEntry 15 }

energyPeakRealPower OBJECT-TYPE

SYNTAX Unsigned32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object indicates the highest observed value for object energyRealPower since the last re-initialization of the management system.

In order to determine the measured value in watts, the value of this object needs to be multiplied by a unit multiplier given by the value of object energyEnergyUnitMultiplier."

::= { energyEntry 16 }

energyReactivePower OBJECT-TYPE

SYNTAX Unsigned32

UNITS "volt-amperes reactive"

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object indicates the current reactive power value at the electrical supply of the entity indicated by index energyConsumerId.

In order to determine the measured value in volt-amperes (var), the value of this object needs to be multiplied by a unit multiplier given by the value of object energyEnergyUnitMultiplier.

The value provided by this object is only useful if the value of object energySupplyType is alternatingCurrent(1). In this case it is RECOMMENDED that at least one of the three values energyReactivePower, energyApparentPowerScale, and energyPhaseAngle

are provided.

If object `energyElectricSupplyType` of this row has a value other than `alternatingCurrent(1)`, then the value of this object MUST be 0.

If object `energyElectricSupplyType` of this row has the value `alternatingCurrent(1)` and if no value for the current reactive power is provided, then the value of this object MUST be `0xFFFF`."

::= { energyEntry 17 }

`energyApparentPower` OBJECT-TYPE

SYNTAX Unsigned32

UNITS "volt-amperes"

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object indicates the current apparent power value measured in volt-ampere (VA) at the electrical supply of the entity for a time interval indicated by object `energySampleInterval`.

The value provided by this object is only useful if the value of object `energySupplyType` is `alternatingCurrent(1)`. In this case it is RECOMMENDED that at least one of the three values `energyReactivePower`, `energyApparentPowerScale`, and `energyPhaseAngle` are provided.

Scale and precision of the value are indicated by objects `energyPowerScale` and `energyPowerPrecision`.

If object `energyElectricSupplyType` of this row has a value other than `alternatingCurrent(1)`, then the value of this object MUST be equal to the value of object `energyRealPower`.

If object `energyElectricSupplyType` of this row has the value `alternatingCurrent(1)` and if no value for the current apparent power is provided, then the value of this object MUST be `-10000000000`."

::= { energyEntry 18 }

`energyPhaseAngle` OBJECT-TYPE

SYNTAX Integer32 (-1..360000)

UNITS "millidegrees"

MAX-ACCESS read-only

STATUS current
DESCRIPTION

"This object indicates the current phase angle value measured at the electrical supply of the entity for a time interval indicated by object energySampleInterval.

The value provided by this object is only useful if the value of object energySupplyType is alternatingCurrent(1). In this case it is RECOMMENDED that at least one of the three values energyReactivePower, energyApparentPowerScale, and energyPhaseAngle are provided.

The value is provided in units of millidegree (one thousands of a degree. This is equivalent to an associated object of type EntitySensorDataScale with the value of milli(8) and an associated object of type EntitySensorPrecision with a value of 0.

The minimum value for this object when indicating an actual angle is 0, the maximum value is 360000.

The maximum error of of the value is indicated by object energyPhaseAngleMaxError.

If object energyElectricSupplyType of this row has a value other than alternatingCurrent(1), then the value of this object MUST be 0.

If object energyElectricSupplyType of this row has the value alternatingCurrent(1) and if no value for the phase angle is provided, then the value of this object MUST be -1."

::= { energyEntry 19 }

energyPhaseAnglePrecision OBJECT-TYPE

SYNTAX Integer32 (0..10000)

UNITS "millidegrees"

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object indicates a the precision of a measured phase angle value. The precision is indicated as a percentage value, in 100ths of a percent. A value of 0 indicates that the precision is unknown or not applicable to the measured value.

This object serves precision indicator for the values

```

    provided by object energyPhaseAngle."
 ::= { energyEntry 20 }

```

```

-----
-- 1.2. Energy Consumption Per Power State Table
-----

```

```

energyPerStateTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF EnergyPerStateEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "This table provides information on the accumulated energy
        consumption of an entity.

        This table extends the energyTable by sharing the
        first index. The first index serves for identifying an
        entity for which energy information is provided. The second
        index identifies a single power state by its name."
 ::= { energyObjects 2 }

```

```

energyPerStateEntry OBJECT-TYPE
    SYNTAX      EnergyPerStateEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Energy consumption information per power state for a
        physical entity."
    INDEX { energyConsumerId, powerStateName }
 ::= { energyPerStateTable 1 }

```

```

EnergyPerStateEntry ::=
    SEQUENCE {
        energyPerStateTotalEnergy      Unsigned32
    }

```

```

energyPerStateTotalEnergy OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "This object indicates the total consumed energy value
        at the electrical supply of the entity indicated by index
        energyConsumerId while being in a specific power state
        indicated by index powerStateName.

        In order to determine the measured value in watts, the value
        of this object needs to be multiplied by a unit multiplier

```

given by the value of object
energyEnergyUnitMultiplier of table
energyTable.

Discontinuities in the value of this counter can occur at
re-initialization of the management system, and at other
times as indicated by the value of
energyDiscontinuityTime."
 ::= { energyPerStateEntry 1 }

-- 1.3. Energy Power History Table

energyHistoryTable OBJECT-TYPE
SYNTAX SEQUENCE OF EnergyHistoryEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION

"This table stores results of energy consumption
measurements for multiple entities.

This table extends the energyTable by sharing the
first index. The first index serves for identifying an
entity for which energy information is provided. The second
index energyHistoryIndex identifies a single measurement
consisting of an energy consumption value and a timestamp.

Creation of entries in this row is controlled individually
for each entity by two parameters: energyMaxHistory and
energySamplingInterval.

The energySamplingInterval controls the difference in time
between the creation of two consecutive entries in this
table. Object energyMaxHistory limits the number of entries
in this table that can be created for the corresponding
entity.

An implementation of the Energy MIB module will remove the
oldest entry for an entity in the energyHistoryTable to
allow the addition of a new entry once the number of
entries for this entity reaches the value indicated by
object energyMaxHistory.

Entries for a specific entity are added to this table
until energyMaxHistory is reached before
entries begin to be removed.

Entries for the same entity are indexed by energyHistoryIndex. The first entry for an entity MUST have an index value of 1. Further new entries MUST be indexed by consecutive numbers in the order in which they are created until the value of energyMaxHistory is reached. Then no further new indices will be assigned, but existing ones will be re-used."

```
::= { energyObjects 3 }
```

energyHistoryEntry OBJECT-TYPE

SYNTAX EnergyHistoryEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"An entry indicating consumed energy for an entity at a certain point in time."

INDEX { energyConsumerId, energyHistoryIndex }

```
::= { energyHistoryTable 1 }
```

EnergyHistoryEntry ::=

SEQUENCE {

energyHistoryIndex Unsigned32,

energyHistoryTimestamp TimeStamp,

energyHistoryTotalEnergy Unsigned32

}

energyHistoryIndex OBJECT-TYPE

SYNTAX Unsigned32 (1..4294967295)

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The index for this entry per entity.

Values of this index MUST be unique per entity used as first index."

```
::= { energyHistoryEntry 1 }
```

energyHistoryTimestamp OBJECT-TYPE

SYNTAX TimeStamp

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object indicates the time at which the energy consumption value provided by object energyHistoryTotalEnergy was measured."

```
::= { energyHistoryEntry 2 }
```

energyHistoryTotalEnergy OBJECT-TYPE

SYNTAX Unsigned32

```

MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
    "This object indicates the total consumed energy measured
    at the electrical power supply of the entity.

    In order to determine the measured value in watt hours,
    the value of this object needs to be multiplied by a unit
    multiplier given by the value of object
    energyEnergyUnitMultiplier in the corresponding entry
    for this entity in table energyTable.

    Discontinuities in the value of this counter can occur at
    re-initialization of the management system, and at other
    times as indicated by the value of
    energyDiscontinuityTime in the corresponding entry
    for this entity in table energyTable."
 ::= { energyHistoryEntry 3 }

```

```

-----
-- 2. Conformance Information
-----

```

```

energyCompliances OBJECT IDENTIFIER ::= { energyConformance 1 }
energyGroups      OBJECT IDENTIFIER ::= { energyConformance 2 }

```

```

-----
-- 2.1. Compliance Statements
-----

```

```

energyCompliance MODULE-COMPLIANCE
STATUS        current
DESCRIPTION
    "The compliance statement for implementations of the
    ENERGY-MIB module.

    A compliant implementation MUST implement the objects
    defined in the mandatory group energyRequiredGroup.

    If one of the entities for which energy consumption is
    reported are supplied by alternating current (AC) then it
    is recommended that not just real power is reported
    (REQUIRED) but it is also RECOMMENDED that at least one
    of three other related values (reactive power, apparent
    power, and phase angle) is reported by implementing at least
    one of the three groups energyReactivePowerGroup,
    energyApparentPowerGroup, and energyPhaseAngleGroup."

```

```
MODULE -- this module
MANDATORY-GROUPS { energyRequiredGroup }
```

```
GROUP energyPowerHistoryGroup
DESCRIPTION
```

"This group is only needed for implementations that support storing time series of measured power values in the energyPowerHistoryTable."

```
GROUP energyACGroup
DESCRIPTION
```

"This group is only needed for implementations that report consumption of electric energy provided by alternating current (AC) supply.

Implementations for devices supplied with direct current (DC) only and implementations that do only report real power reporting for alternative current do not need to implement objects in this group."

```
GROUP energyReactivePowerGroup
DESCRIPTION
```

"Information provided by elements in this group is redundant to information provided by elements in the energyApparentPowerGroup and the energyPhaseAngleGroup.

For compliant implementations that report consumption of electric energy provided by alternating current (AC) supply it is RECOMMENDED to at least one of the three groups energyReactivePowerGroup, energyApparentPowerGroup, and energyPhaseAngleGroup."

```
GROUP energyApparentPowerGroup
DESCRIPTION
```

"Information provided by elements in this group is redundant to information provided by elements in the energyReactivePowerGroup and the energyPhaseAngleGroup.

For compliant implementations that report consumption of electric energy provided by alternating current (AC) supply it is RECOMMENDED to at least one of the three groups energyReactivePowerGroup, energyApparentPowerGroup, and energyPhaseAngleGroup."

```
GROUP energyPhaseAngleGroup
DESCRIPTION
```

"Information provided by elements in this group is redundant to information provided by elements in the

energyReactivePowerGroup and the energyApparentPowerGroup.

For compliant implementations that report consumption of electric energy provided by alternating current (AC) supply it is RECOMMENDED to at least one of the three groups energyReactivePowerGroup, energyApparentPowerGroup, and energyPhaseAngleGroup."

::= { energyCompliances 1 }

-- 2.2. Object Grouping

energyRequiredGroup OBJECT-GROUP

OBJECTS {
 energySensorOperStatus,
 energyNominalSupplyVoltage,
 energyElectricSupplyType,
 energyTotalEnergy,
 energyEnergyUnitMultiplier,
 energyEnergyPrecision,
 energyMeasurementMethod,
 energyDiscontinuityTime,
 energyPowerUnitMultiplier,
 energyPowerPrecision,
 energyRealPower,
 energyPeakRealPower,
 energyPerStateTotalEnergy
}

STATUS current

DESCRIPTION

"A compliant implementation MUST implement the objects contained in this group."

::= { energyGroups 1 }

energyPowerHistoryGroup OBJECT-GROUP

OBJECTS {
 energySampleInterval,
 energyMaxHistory,
 energyHistoryTimestamp,
 energyHistoryTotalEnergy
}

STATUS current

DESCRIPTION

"The group of object for reporting details of AC power measurement."

::= { energyGroups 2 }

```
energyACGroup OBJECT-GROUP
  OBJECTS {
    energyReactivePower,
    energyApparentPower,
    energyPhaseAngle,
    energyPhaseAnglePrecision
  }
  STATUS      current
  DESCRIPTION
    "The group of object for reporting details of
    AC power measurement."
  ::= { energyGroups 3 }

energyReactivePowerGroup OBJECT-GROUP
  OBJECTS {
    energyReactivePower
  }
  STATUS      current
  DESCRIPTION
    "The group of object for reporting the reactive power
    measured for AC supply."
  ::= { energyGroups 4 }

energyApparentPowerGroup OBJECT-GROUP
  OBJECTS {
    energyApparentPower
  }
  STATUS      current
  DESCRIPTION
    "The group of object for reporting the apparent power
    measured for AC supply."
  ::= { energyGroups 5 }

energyPhaseAngleGroup OBJECT-GROUP
  OBJECTS {
    energyPhaseAngle,
    energyPhaseAnglePrecision
  }
  STATUS      current
  DESCRIPTION
    "The group of object for reporting the phase angler
    measured for AC supply."
  ::= { energyGroups 6 }

END
```

8.3. Battery MIB

```
BATTERY-MIB DEFINITIONS ::= BEGIN

IMPORTS
    MODULE-IDENTITY, OBJECT-TYPE, NOTIFICATION-TYPE,
    mib-2, Integer32, Unsigned32, Counter32
        FROM SNMPv2-SMI -- RFC2578
    DateAndTime
        FROM SNMPv2-TC -- RFC2579
    MODULE-COMPLIANCE, OBJECT-GROUP, NOTIFICATION-GROUP
        FROM SNMPv2-CONF -- RFC2580
    entPhysicalIndex
        FROM ENTITY-MIB; -- RFC4133
```

```
batteryMIB MODULE-IDENTITY
    LAST-UPDATED "201001291200Z" -- 29 January 2010
    ORGANIZATION "IETF OPSAWG Working Group"
    CONTACT-INFO
        "General Discussion: opsawg@ietf.org
        To Subscribe: https://www.ietf.org/mailman/listinfo/opsawg
        Archive: http://www.ietf.org/mail-archive/web/opsawg
```

```
Co-editor:
    Juergen Quittek
    NEC Europe Ltd.
    NEC Laboratories Europe
    Kurfuersten-Anlage 36
    69115 Heidelberg
    Germany
    Tel: +49 6221 4342-115
    Email: quittek@neclab.eu
```

```
Co-editor:
    Thomas Dietz
    NEC Europe Ltd.
    NEC Laboratories Europe
    Kurfuersten-Anlage 36
    69115 Heidelberg
    Germany
    Phone: +49 6221 4342-128
    Email: Thomas.Dietz@neclab.eu"
```

```
DESCRIPTION
    "This MIB module defines a set of objects for monitoring
    batteries of networked devices and of their components.
```

```
Copyright (c) 2010 IETF Trust and the persons identified as
authors of the code. All rights reserved.
```

Redistribution and use in source and binary forms, with or without modification, is permitted pursuant to, and subject to the license terms contained in, the Simplified BSD License set forth in Section 4.c of the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>).

This version of this MIB module is part of RFC yyyy; see the RFC itself for full legal notices."

-- replace yyyy with actual RFC number & remove this notice

-- Revision history

REVISION "201001291200Z" -- 29 January 2010
DESCRIPTION

"Initial version, published as RFC yyyy."

-- replace yyyy with actual RFC number & remove this notice

::= { mib-2 zzz }

-- zzz to be assigned by IANA.

--*****
-- Top Level Structure of the MIB module
--*****

batteryNotifications OBJECT IDENTIFIER ::= { batteryMIB 0 }
batteryObjects OBJECT IDENTIFIER ::= { batteryMIB 1 }
batteryConformance OBJECT IDENTIFIER ::= { batteryMIB 2 }

=====
-- 1. Object Definitions
=====

-- 1.1. Battery Table

batteryTable OBJECT-TYPE
SYNTAX SEQUENCE OF BatteryEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
"This table provides information on batteries in networked devices. It is designed as a sparse augment of the entPhysicalTable defined in the ENTITY-MIB module and assumes that each battery is represented by an individual row in the entPhysicalTable with an individual value for the index entPhysicalIndex.

Entries appear in this table only for entities that represent a battery. An entry in this table SHOULD be created at the same time as the associated entPhysicalEntry. An entry SHOULD be destroyed if the associated entPhysicalEntry is destroyed."

```
::= { batteryObjects 1 }
```

```
batteryEntry OBJECT-TYPE
```

```
SYNTAX      BatteryEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
```

```
"An entry providing information on a battery."
```

```
INDEX { entPhysicalIndex } -- SPARSE-AUGMENTS
::= { batteryTable 1 }
```

```
BatteryEntry ::=
```

```
SEQUENCE {
    batteryType          INTEGER,
    batteryTechnology    INTEGER,
    batteryNominalVoltage Unsigned32,
    batteryNumberOfCells Unsigned32,
    batteryNominalCapacity Unsigned32,
    batteryRemainingCapacity Unsigned32,
    batteryChargingCycleCount Counter32,
    batteryLastChargingCycleTime DateAndTime,
    batteryState         INTEGER,
    batteryCurrentCharge Unsigned32,
    batteryCurrentChargePercentage Unsigned32,
    batteryCurrentVoltage Unsigned32,
    batteryCurrentCurrent Integer32,
    batteryLowAlarmPercentage Unsigned32,
    batteryLowAlarmVoltage Unsigned32,
    batteryReplacementAlarmCapacity Unsigned32,
    batteryReplacementAlarmCycles Unsigned32
}
```

```
batteryType OBJECT-TYPE
```

```
SYNTAX      INTEGER {
                primary(1),
                rechargeable(2),
                capacitor(3),
                other(4),
                unknown(5)
            }
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
```

"This object indicates the type of battery. It distinguishes between one-way primary batteries, rechargeable secondary batteries and capacitors which are not really batteries but often used in the same way as a battery.

The value other(4) can be used if the battery type is known but none of the ones above. Value unknown(5) is to be used if the type of battery cannot be determined."

```
::= { batteryEntry 1 }
```

batteryTechnology OBJECT-TYPE

```
SYNTAX          INTEGER {
                zincCarbon(1),
                zincChloride(2),
                oxyNickelHydroxide(3),
                lithiumCopper(4),
                lithiumIron(5),
                lithiumManganese(6),
                zincAir(7),
                silverOxide(8),
                alkaline(9),
                leadAcid(10),
                nickelCadmium(12),
                nickelMetalHybride(13),
                nickelZinc(14),
                lithiumIon(15),
                lithiumPolymer(16),
                doubleLayerCapacitor(17),
                other(18),
                unknown(19)
                }
```

```
MAX-ACCESS      read-only
```

```
STATUS          current
```

```
DESCRIPTION
```

"This object indicates the technology used by the battery. Values 1-8 are primary battery technologies, values 10-16 are rechargeable battery technologies and value alkaline(9) is used for primary batteries as well as for rechargeable batteries.

The value other(18) can be used if the battery type is known but none of the ones above. Value unknown(19) is to be used if the type of battery cannot be determined."

```
::= { batteryEntry 2 }
```

batteryNominalVoltage OBJECT-TYPE

```
SYNTAX          Unsigned32
```

```
UNITS          "millivolt"
```

MAX-ACCESS read-only
STATUS current
DESCRIPTION
"This object provides the nominal voltage of the battery
in units of millivolt (mV).

Note that the nominal voltage is a constant value and
typically different from the actual voltage of the battery.

A value of 0 indicates that the nominal voltage is unknown."
 ::= { batteryEntry 3 }

batteryNumberOfCells OBJECT-TYPE

SYNTAX Unsigned32
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"This object indicates the number of cells contained in the
battery.

A value of 0 indicates that the number of cells is unknown."
 ::= { batteryEntry 4 }

batteryNominalCapacity OBJECT-TYPE

SYNTAX Unsigned32
UNITS "milliampere hours"
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"This object provides the nominal capacity of the battery
in units of milliampere hours (mAh).

Note that the nominal capacity is a constant value and
typically different from the actual capacity of the battery.

A value of 0 indicates that the nominal capacity is unknown."
 ::= { batteryEntry 5 }

batteryRemainingCapacity OBJECT-TYPE

SYNTAX Unsigned32
UNITS "milliampere hours"
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"This object provides the ACTUAL REMAINING capacity of the
battery in units of milliampere hours (mAh).

Note that the actual capacity needs to be measured and is

typically an estimate based on observed discharging and charging cycles of the battery.

A value of 'ffffffff'H indicates that the actual capacity cannot be determined."

```
::= { batteryEntry 6 }
```

batteryChargingCycleCount OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object indicates the number of charging cycles that that the battery underwent. Please note that the precise definition of a recharge cycle varies for different kinds of batteries and of devices containing batteries.

For batteries of type primary(1) the value of this object is always 0.

A value of 'ffffffff'H indicates that the number of charging cycles cannot be determined."

```
::= { batteryEntry 7 }
```

batteryLastChargingCycleTime OBJECT-TYPE

SYNTAX DateAndTime

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The date and time of the last charging cycle. The value '0000000000000000'H is returned if the battery has not been charged yet or if the last charging time cannot be determined.

For batteries of type primary(1) the value of this object is always '0000000000000000'H."

```
::= { batteryEntry 8 }
```

batteryState OBJECT-TYPE

```
SYNTAX INTEGER {
    full(1),
    partiallyCharged(2),
    empty(3),
    charging(4),
    discharging(5),
    unknown(6)
}
```

MAX-ACCESS read-only

```

STATUS      current
DESCRIPTION
    "This object indicates the current state of the battery.
    Value full(1) indicates a full battery with a capacity
    given by onject batteryRemainingCapacity. Value empty(3)
    indicates a battery that cannot be used for providing
    electric power before charging it. Value partiallyCharged(2)
    is provided if the battery is neither empty nor full and if
    no charging or discharging is in progress. Charging or
    discharging of hte battery is indicated by values charging(3)
    or discharging(4), respectively.

    Value unknown(6) is to be used if the state of the battery
    cannot be determined."
 ::= { batteryEntry 9 }

```

```

batteryCurrentCharge OBJECT-TYPE
    SYNTAX      Unsigned32
    UNITS       "milliampere hours"
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "This object provides the current charge of the battery
        in units of milliampere hours (mAh).

        Note that the current charge needs to be measured and is
        typically an estimate based on observed discharging and
        charging cycles of the battery.

        A value of 'ffffffff'H indicates that the current charge
        cannot be determined."
 ::= { batteryEntry 10 }

```

```

batteryCurrentChargePercentage OBJECT-TYPE
    SYNTAX      Unsigned32 (0..10000)
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "This object provides the current charge of the battery
        relative to the nominal capacity in units of a hundreds
        of a percent.

```

```

-----
-- Open issue:
--   Should it be the percentage of the nominal capacity
--   or of the current capacity?
-----

```

Note that this value needs to be measured and is typically an estimate based on observed discharging and charging cycles of the battery.

A value of 'ffffffff'H indicates that the relative current charge cannot be determined."

::= { batteryEntry 11 }

batteryCurrentVoltage OBJECT-TYPE

SYNTAX Unsigned32

UNITS "millivolt"

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object provides the current voltage of the battery in units of millivolt (mV).

A value of 'ffffffff'H indicates that the current voltage cannot be determined."

::= { batteryEntry 12 }

batteryCurrentCurrent OBJECT-TYPE

SYNTAX Integer32

UNITS "milliampere"

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object provides the current charging or discharging current of the battery in units of milliampere (mA). Charging current is indicated by positive values, discharging current is indicated by negative values.

A value of '7ffffffff'H indicates that the current current cannot be determined."

::= { batteryEntry 13 }

batteryLowAlarmPercentage OBJECT-TYPE

SYNTAX Unsigned32 (0..10000)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"This object provides the lower threshold value for object batteryCurrentChargePercentage. If the value of object batteryCurrentChargePercentage falls below this threshold, a low battery alarm will be raised. The alarm procedure may include generating a batteryLowNotification.

A value of 0 indicates that the no alarm will be raised for

any value of object batteryCurrentChargePercentage."
 ::= { batteryEntry 14 }

batteryLowAlarmVoltage OBJECT-TYPE

SYNTAX Unsigned32
UNITS "millivolt"
MAX-ACCESS read-only
STATUS current
DESCRIPTION

"This object provides the lower threshold value for object batteryCurrentVoltage. If the value of object batteryCurrentVoltage falls below this threshold, a low battery alarm will be raised. The alarm procedure may include generating a batteryLowNotification.

A value of 0 indicates that the no alarm will be raised for any value of object batteryCurrentVoltage."

::= { batteryEntry 15 }

batteryReplacementAlarmCapacity OBJECT-TYPE

SYNTAX Unsigned32
UNITS "milliampere hours"
MAX-ACCESS read-only
STATUS current
DESCRIPTION

"This object provides the lower threshold value for object batteryRemainingCapacity. If the value of object batteryRemainingCapacity falls below this threshold, a battery aging alarm will be raised. The alarm procedure may include generating a batteryAgingNotification.

A value of 0 indicates that the no alarm will be raised for any value of object batteryRemainingCapacity."

::= { batteryEntry 16 }

batteryReplacementAlarmCycles OBJECT-TYPE

SYNTAX Unsigned32
UNITS "milliampere hours"
MAX-ACCESS read-only
STATUS current
DESCRIPTION

"This object provides the upper threshold value for object batteryChargingCycleCount. If the value of object batteryChargingCycleCount rises above this threshold, a battery aging alarm will be raised. The alarm procedure may include generating a batteryAgingNotification.

A value of 0 indicates that the no alarm will be raised for

```
any value of object batteryChargingCycleCount."  
 ::= { batteryEntry 17 }
```

```
-----  
-- 2. Notifications  
-----
```

```
batteryLowNotification NOTIFICATION-TYPE  
  OBJECTS      {  
    batteryCurrentChargePercentage,  
    batteryCurrentVoltage  
  }  
  STATUS       current  
  DESCRIPTION  
    "This notification can be generated when the current charge  
    (batteryCurrentChargePercentage) or the current voltage  
    (batteryCurrentVoltage) of the battery falls below a  
    threshold defined by object batteryLowAlarmPercentage or  
    object batteryLowAlarmVoltage, respectively."  
  ::= { batteryNotifications 1 }
```

```
batteryAgingNotification NOTIFICATION-TYPE  
  OBJECTS      {  
    batteryRemainingCapacity,  
    batteryChargingCycleCount  
  }  
  STATUS       current  
  DESCRIPTION  
    "This notification can be generated when the remaining  
    capacity (batteryRemainingCapacity) falls below a threshold  
    defined by object batteryReplacementAlarmCapacity  
    or when the charging cycle count of the battery  
    (batteryChargingCycleCount) exceeds the threshold defined  
    by object batteryLowAlarmPercentage."  
  ::= { batteryNotifications 2 }
```

```
-----  
-- 3. Conformance Information  
-----
```

```
batteryCompliances OBJECT IDENTIFIER ::= { batteryConformance 1 }  
batteryGroups      OBJECT IDENTIFIER ::= { batteryConformance 2 }
```

```
-----  
-- 3.1. Compliance Statements  
-----
```

```
batteryCompliance MODULE-COMPLIANCE
  STATUS      current
  DESCRIPTION
    "The compliance statement for implementations of the
    POWER-STATE-MIB module.

    A compliant implementation MUST implement the objects
    defined in the mandatory group psmRequiredGroup."
  MODULE -- this module
  MANDATORY-GROUPS {
    batteryDescriptionGroup,
    batteryStatusGroup,
    batteryAlarmThresholdsGroup
  }
  GROUP      batteryNotificationsGroup
  DESCRIPTION
    "A compliant implementation does not have to implement
    the psmNotificationsGroup."
  ::= { batteryCompliances 1 }
```

```
-----
-- 3.2. MIB Grouping
-----
```

```
batteryDescriptionGroup OBJECT-GROUP
  OBJECTS {
    batteryType,
    batteryTechnology,
    batteryNominalVoltage,
    batteryNumberOfCells,
    batteryNominalCapacity
  }
  STATUS      current
  DESCRIPTION
    "A compliant implementation MUST implement the objects
    contained in this group."
  ::= { batteryGroups 1 }
```

```
batteryStatusGroup OBJECT-GROUP
  OBJECTS {
    batteryRemainingCapacity,
    batteryChargingCycleCount,
    batteryLastChargingCycleTime,
    batteryState,
    batteryCurrentCharge,
    batteryCurrentChargePercentage,
    batteryCurrentVoltage,
    batteryCurrentCurrent
  }
```

```
    }
    STATUS          current
    DESCRIPTION
        "A compliant implementation MUST implement the objects
        contained in this group."
    ::= { batteryGroups 2 }

batteryAlarmThresholdsGroup OBJECT-GROUP
    OBJECTS {
        batteryLowAlarmPercentage,
        batteryLowAlarmVoltage,
        batteryReplacementAlarmCapacity,
        batteryReplacementAlarmCycles
    }
    STATUS          current
    DESCRIPTION
        "A compliant implementation MUST implement the objects
        contained in this group."
    ::= { batteryGroups 3 }

batteryNotificationsGroup NOTIFICATION-GROUP
    NOTIFICATIONS {
        batteryLowNotification,
        batteryAgingNotification
    }
    STATUS          current
    DESCRIPTION
        "A compliant implementation does not have to implement the
        notification contained in this group."
    ::= { batteryGroups 4 }
END
```

9. Security Considerations

There are no management objects defined in this MIB module that have a MAX-ACCESS clause of read-write and/or read-create. So, if this MIB module is implemented correctly, then there is no risk that an intruder can alter or create any management objects of this MIB module via direct SNMP SET operations.

Some of the readable objects in this MIB module (i.e., objects with a MAX-ACCESS other than not-accessible) may be considered sensitive or vulnerable in some network environments. It is thus important to control even GET and/or NOTIFY access to these objects and possibly to even encrypt the values of these objects when sending them over the network via SNMP. These are the tables and objects and their sensitivity/vulnerability:

- o This list is still to be done.

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

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

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

10. IANA Considerations

The MIB modules in this document uses the following IANA-assigned OBJECT IDENTIFIER values recorded in the SMI Numbers registry:

Descriptor	OBJECT IDENTIFIER value
-----	-----
powerStateMIB	{ mib-2 xxx }
energyMIB	{ mib-2 yyy }
batteryMIB	{ mib-2 zzz }

Other than that this document does not impose any IANA considerations.

11. References

11.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2578] McCloghrie, K., Ed., Perkins, D., Ed., and J. Schoenwaelder, Ed., "Structure of Management Information Version 2 (SMIV2)", STD 58, RFC 2578, April 1999.

- [RFC2579] McCloghrie, K., Ed., Perkins, D., Ed., and J. Schoenwaelder, Ed., "Textual Conventions for SMIV2", STD 58, RFC 2579, April 1999.
- [RFC2580] McCloghrie, K., Perkins, D., and J. Schoenwaelder, "Conformance Statements for SMIV2", STD 58, RFC 2580, April 1999.
- [RFC4268] Chisholm, S. and D. Perkins, "Entity State MIB", RFC 4268, November 2005.
- [RFC3621] Berger, A. and D. Romascanu, "Power Ethernet MIB", RFC 3621, December 2003.
- [RFC3433] Bierman, A., Romascanu, D., and K. Norseth, "Entity Sensor Management Information Base", RFC 3433, December 2002.
- [RFC4133] Bierman, A. and K. McCloghrie, "Entity MIB (Version 3)", RFC 4133, August 2005.
- [I-D.quittek-power-monitoring-requirements]
Quittek, J., Winter, R., Dietz, T., Claise, B., and M. Chandramouli, "Requirements for Power Monitoring", draft-quittek-power-monitoring-requirements-01 (work in progress), July 2010.
- [I-D.claise-energy-monitoring-mib]
Claise, B., Chandramouli, M., Parello, J., and B. Schoening, "Power and Energy Monitoring MIB", draft-claise-energy-monitoring-mib-06 (work in progress), October 2010.
- [I-D.parello-eman-energy-aware-mib]
Claise, B. and J. Parello, "Energy-aware Networks and Devices MIB", draft-parello-eman-energy-aware-mib-00 (work in progress), October 2010.

11.2. Informative References

- [RFC1628] Case, J., "UPS Management Information Base", RFC 1628, May 1994.
- [RFC3410] Case, J., Mundy, R., Partain, D., and B. Stewart, "Introduction and Applicability Statements for Internet-Standard Management Framework", RFC 3410, December 2002.

Authors' Addresses

Juergen Quittek (editor)
NEC Europe Ltd.
NEC Laboratories Europe
Network Research Division
Kurfuersten-Anlage 36
Heidelberg 69115
DE

Phone: +49 6221 4342-115
Email: quittek@nw.neclab.eu

Rolf Winter
NEC Europe Ltd.
NEC Laboratories Europe
Network Research Division
Kurfuersten-Anlage 36
Heidelberg 69115
DE

Phone: +49 6221 4342-121
Email: Rolf.Winter@nw.neclab.eu

Thomas Dietz
NEC Europe Ltd.
NEC Laboratories Europe
Network Research Division
Kurfuersten-Anlage 36
Heidelberg 69115
DE

Phone: +49 6221 4342-128
Email: Thomas.Dietz@nw.neclab.eu

Dominique Dudkowski
NEC Europe Ltd.
NEC Laboratories Europe
Network Research Division
Kurfuersten-Anlage 36
Heidelberg 69115
DE

Phone: +49 6221 4342-233
Email: Dominique.Dudkowski@nw.neclab.eu

