                        Energy Reporting Framework
                     draft-nordman-eman-er-framework-02

Abstract

   Managing energy consumption of devices presents new challenges and
   issues. The EMAN Requirements draft identifies essential
   capabilities needed to accomplish this. This draft describes how an
   energy management system can use EMAN to gather and interpret data
   from individual devices, and how some of the Requirements are
   implemented in the model. This document focuses on Energy Reporting,
   though acknowledges and fully includes the limited control functions
   specified in the Requirements draft. Topics addressed in detail
   include the topology of power distribution, reporting mechanisms, and
   the various roles of devices, power interfaces, and components.

Table of Contents

1. Introduction

   Managing energy consumption of devices is different from typical
   network management functions because of the special nature of energy
   supply and use. The EMAN Requirements draft [RFC6988] identifies
   necessary capabilities for an energy management standard. This
   document describes how an energy management system (EnMS) uses EMAN
   and how to use and interpret EMAN data.

   In addition to the Requirements draft, this document derives content
   and inspiration from several now-expired drafts: Considerations for
   Power and Energy Management [I-D.norwin-energy-consider], Energy
   Perspective on Applicability [I-D.nordman-eman-energy-perspective],
   and most significantly, the Reference Model for Energy Management
   [I-D.quittek-eman-reference-model]. The EMAN Applicability statement
   [I-D.ietf-eman-applicability-statement] is also informative.

   The current EMAN Framework draft [I-D.ietf-eman-framework] addresses
   some of the requirements in ways suitable for this framework,
   particularly for Advanced capabilities (see Section 5). Those
   requirements are noted below.

   The EMAN requirements are overwhelmingly about the reporting of
   energy data, not about control. The only requirements that address
   control are setting power state (directly and by a proxy) and cutting
   power supply. Thus, while these are fully included in this
   framework, it is titled "Energy Reporting" to match the empirical
   facts about the content of the EMAN requirements.

1.1. Guiding Principles

   This presentation, in form and content, takes simplicity as an
   overarching goal. Complexity is burdensome for implementation of
   EMAN capabilities in individual devices, burdensome for
   implementation of management systems, and burdensome for readers and
   users of this standard to understand what they need to to accomplish
   their goals.

   Energy management involves people of many backgrounds and so
   documents about it should be accessible to a wide variety of
   audiences, from network professionals, to energy professionals, to
   building managers, or any person with an interest in energy use.

2. Concepts

   At the core of this framework are just a few key concepts. Energy is
   used by Devices. Devices have Power Interfaces, which are like
   network interfaces, through which power is transferred into or out of
   a device. Devices may have internal components with distinct power
   consumption or other characteristics. Measurement for devices occurs
   at interfaces so that the total or net consumption of a device can be
   determined from these. EMAN data are transferred from a device to an
   Energy Management System.

2.1. Energy Management System

   An Energy Management System (EnMS) is an entity that receives EMAN
   data from many devices and interprets it. An EnMS is often in the
   same building as the devices being monitored. A building can have
   zero, one, or many EnMS. A device can do both, acting as an EnMS and
   reporting EMAN data to another EnMS. While the EMAN Requirements
   draft does not identify requirements for an EnMS, it is necessary to
   understand some of EnMS operation to fully describe how the EMAN
   framework operates.

   Basic EnMS operations include: discovering relevant devices,
   acquiring static data about them, acquiring dynamic data about them
   on a periodic basis, processing the resulting data, and implementing
   control and/or reporting functions.

2.2. Device

   A Device is most commonly a complete product. At any given time, a
   device may be drawing electricity in, and may be sending electricity
   out. Combining these results in the net energy consumed by a device.

For devices with a single power interface, there is an identity
between data about the interface and about the device as a whole, so
the two can share an identity within EMAN. The power interface
reports some data that a device does not (see Section 2.5) so that in
most cases both will be needed.

2.3. Power Interface

A Power Interface (PI) is an interface on a device through which
power can flow into a device (an inlet) or out of it (an outlet).
Some PIs change over time from being an inlet to being an outlet and
vice versa, however most PIs never change. Most devices have a
single inlet. Devices with multiple inlets often have them connected
to separate power distribution trees. Most devices have no outlets,
but those that do often have many. The only distinction between an
inlet and outlet is the sign of the power value: positive for an
inlet and negative for an outlet. A PI can indicate whether it is
capable of being only an inlet, only an outlet, or can switch between
the two. PIs are all contained (in ENTITY MIB sense) in the device;
a PI is never contained in a component, and a PI cannot contain
anything within it. A PI consumes no power itself. It is always on
the border of a device, never internal.

The flow of electricity within a building is determined by which
power interfaces are connected to each other - the wiring topology.
Thus, a key attribute of a PI is a list of the other interfaces it is
connected to. The power interface term is not new. It is used
similarly by the Power over Ethernet (PoE) standard [IEEE-802.3af]
and [IEEE-802.3at] where a power interface denotes the interface
between a device and the Ethernet transmission medium.

2.4. Component

A component is a distinct part of a device. Components lack some of
the features of devices, such as having power interfaces; instead,
they simply have a net total consumption from the pool of power
available within a device. A component can contain other components,
that draw from the pool of power within the containing component.

2.5. Energy Object

Devices, PIs, and Components are all Energy Objects (EOs). The term
"entity" in the Requirements draft generally corresponds to Energy
Object. The kinds of data available for an EO depends on its type as
shown in Figure 1. Basic data will be implemented by many or most
devices. Most devices are unlikely to implement advanced data.

The total energy and power for a device must match the total of all
of its PIs. PIs dump power into or take power out of the pool of
power in a device. A component draws power from the pool.
Components do not have power interfaces. The sum of all components
in a device may be less than total device consumption as there may be
hardware consuming power that is not part of any modeled component.

```
Device PI Component  Kind
                  Basic
  X     X     X       Identification
  X     X     X       Local data
        X             Power Interfaces
        X             Power, static
  X     X     X       Power
  X     X     X       Power state
  X     X     X       Energy
  X                   Reporting
                  Advanced
  X     X             Identification, Advanced
        X             Power, Advanced
              X       Battery``````
  X     X     X       Energy, Advanced
  X     X     X       Time-series data
  X                   Reporting, advanced
  X                   Proxy control
```

                 Figure 1: Kinds of EMAN data

2.6. Battery

   A battery in a device is a special type of component. EMAN has
   battery-specific data such as battery chemistry and charge state, see
   [I-D.ietf-eman-battery-mib]. A device can report on a battery with
   the battery-specific data, the component data, or both.

3. Topologies

   EMAN covers several distinct topologies, particularly the
   distribution of electricity and the communication of information.
   The ability of EMAN to model power interfaces (PIs) of devices
   enables flexible and powerful capabilities. This section describes
   some of them. With this model an EnMS can query all devices in a
   building for their EMAN data and combine what it receives from each
   into a comprehensive picture of electricity flows and device state.

3.1. Power Distribution

   An EnMS combines all the data it acquires to create as complete a
   picture of power flows as possible; it is not necessary for each
   device to have complete information about its connectivity. For
   example, terminal devices that only consume power may only know the
   identity of the PI from which they draw power. The EnMS then sees
   many devices connected to a single outlet PI and can infer that they
   are all wired to each other, regardless of whether the supplying PI
   knows the identity of the devices it powers. Similarly, a supplying
   device may know the identify of the PI it supplies power to and so
   can create the map, even if the supplied device does not know where
   it draws power from.

   For each PI on a device, the group of PIs that are known to be
   directly wired to that PI is reported. In Figure 2, Device-A can
   report that its PI-1 is wired to PI-2, and Device-B can report that
   PI-2 is wired to PI-1. The EnMS knows that PI-1 is part of Device-A
   and that PI-2 is part of Device-B. Only one of the PIs needs to know
   of the wiring connection for the EnMS to fully understand it. The
   drawing shows how a PI is part of a device but is on its periphery.
   The line shown is of power flow only.

```
        +--------------+                    +--------------+
        |        +--------+          +--------+        |
        | Device-A | PI-1 |>-------->| PI-2  | Device-B |
        |        +--------+          +--------+        |
        +--------------+                    +--------------+
```
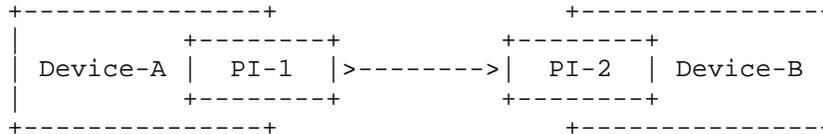
                 Figure 2: Simple wiring topology example

   Figure 3 shows how wiring is commonly done in AC supply systems, with
   Device A being a circuit breaker. In this case, four devices are
   wired together. It is possible that all four know of the identity of
   all others, but commonly less is known. PI-2, PI-3, and PI-4 may
   each know they are wired to PI-1, and PI-1 may know nothing, but the
   linkage of all four can be inferred by the EnMS. As another example,
   PI-3 and PI-4 may know of the connection to PI-1, with PI-1 knowing
   it is connected to PI-2; this also enables construction of the full
   set. Finally, PI-1 could know it is connected to PI-2 and PI-3, but
   have no knowledge of PI-4; however, it could observe that PI-2 and
   PI-3 do not account for all the energy leaving PI-1 and so infer that
   at least one other device is also wired to PI-1.

```
                              +--------+----------+
                       ---->| PI-2  | Device-B |
        +--------------+       |  +--------+----------+
        |        +--------+    |  +--------+----------+
        | Device-A | PI-1 |>-------->| PI-3  | Device-C |
```

```
    |             +--------+      |    +--------+----------+
    +--------------+             |    +--------+----------+
                                 ---->|  PI-4  | Device-D |
                                      +--------+----------+
```
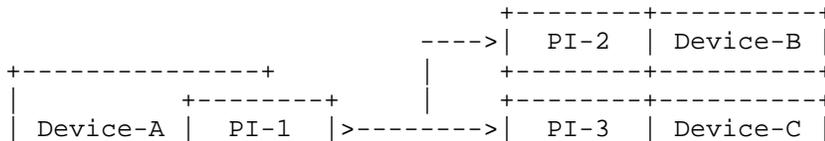
                 Figure 3: Typical AC wiring topology example

Figure 4 shows how wiring is commonly done in DC supply systems, with
Device-A being a PoE switch, USB hub, or similar device. For
technologies which inherently have communication, device identity can
be readily determined.

```
    +----------+--------+             +--------+----------+
    |          |  PI-1  |>-------->|  PI-4  | Device-B |
    |          +--------+             +--------+----------+
    |          +--------+             +--------+----------+
    | Device-A |  PI-2  |>-------->|  PI-5  | Device-C |
    |          +--------+             +--------+----------+
    |          +--------+             +--------+----------+
    |          |  PI-3  |>-------->|  PI-6  | Device-D |
    +----------+--------+             +--------+----------+
```
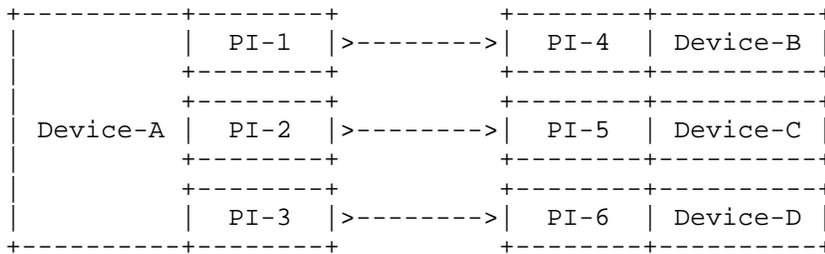
                 Figure 4: Typical DC wiring topology example

From the capability of each PI (or the direction of power flow on
first report) it is clear which PI is the source. For devices that
provide power (typically a power distribution unit, circuit breaker,
or PoE switch), one can determine which PIs are inlets and which are
outlets. Thus, mapping a traditional tree-structured power
distribution system is a simple process.

Some installations have more complex power distribution, as with more
than one device providing power to a circuit, or PIs that can and do
change the direction of power flow. These can be readily modeled
though require more sophisticated interpretation by the EnMS.

An EnMS may choose to put information it has inferred (principally
about PI connectivity) back into the individual devices, but is not
required to. Doing so is valuable when there is more than one EnMS
present.

With a power distribution map, a management system knows which
devices supply power to which other devices, so that the effect of
switching off a PI (usually at an outlet, but possibly at an inlet)
can be determined. The same applies to metering at PIs, which can
also occur at an outlet or inlet.

Power source control is accomplished by physically preventing power from flowing, or re-enabling it. In contrast, power state control is accomplished by communication protocols and not by power distribution control so that power state control mechanisms and capabilities have no required relation to power distribution systems. Power control for a PI is modeled with power state, with on corresponding with power able to flow, and off that power cannot flow.

3.2. Reporting

Only devices report data; PIs and components do not. An EMAN device may report on itself, or on a device other than itself. Self-reporting is the basic EMAN case and simply involves a device putting its internal data into the EMAN representation. The EnMS knows the identity of the device doing the reporting and the identity of the device being reported on, and these are the same. The left side of Figure 5 shows this.

```
     +----------+              +----------+
     |  EnMS    |              |  EnMS    |
     +----------+              +----------+
          ^                         ^
          |                         |
     +----------+              +----------+        +----------+
     | Device-A |              | Device-A |<-----| Device-B |
     +----------+              +----------+        +----------+
```
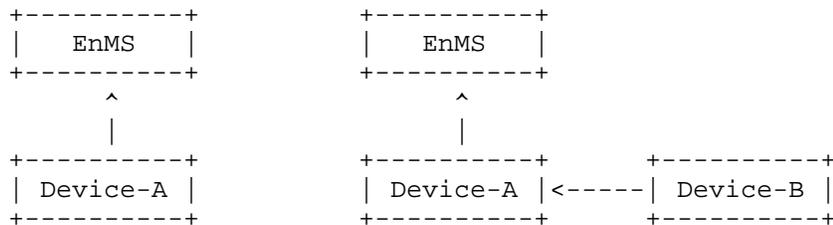
Figure 5: Two basic reporting scenarios

The other EMAN case is non-self-reporting, or other-reporting. The EnMS knows the identity of the device reporting and the identify of the device being reported on and can see that they are different. There are three types of other-reporting relationships, where Device-B is:

a) an IP device (like Device-A) that has the ability to report
   EMAN data to an EnMS,
b) an IP device that does not have the ability to report EMAN data
   to an EnMS, or
c) a non-IP device.

The right side of Figure 5 shows this where the Device-B to Device-A communication technology depends on which case applies.

Case a) is particularly useful when some amount of collection and/or summation is done. Summation can be across energy objects, and/or across time for an individual energy object. The reporting device

may retain the detailed data in case it becomes of interest to an
EnMS later.

Case c) includes circumstances in which some or all of the EMAN data
are communicated over non-IP mechanisms, as well as when the EMAN
device monitors power flowing to the monitored device and may have
access to other information, such as the identity of the device it
provides power to.

Case b) is similar to c) except that the device reported on is an IP
device that is unable to report data in EMAN format.

Practically, the EnMS is indifferent to the distinction between these
cases or even whether the data is self-reported or other-reported; it
only matters what device is being reported on.

The lines shown in Figure 5 are data transfer, not power flow. For
the right side of Figure 5, Device-B could be powered by Device-A, or
have no power relation at all with Device-A (only a reporting
connection). How Device-A obtains the data about Device-B has no
effect on how the data are reported.

4. Fulfilling Basic EMAN Requirements

   This section further elaborates the ER Framework and shows how its
   features fulfill many of the requirements in the EMAN Requirements
   draft. Each subsection includes the relevant requirements,
   abbreviated (for the full list see Section 7). Requirement numbers
   are noted in the text where the requirement is met. This section
   only covers basic requirements that many devices will implement.
   More advanced capabilities and requirements are covered in Section 5.

   When implementing EMAN capability on a device, only the basic
   identification data are required. All other data in Section 4 and
   all data in Section 5 are optional to report. In a few cases,
   features included that do not derive from specific requirements are
   included, and these are always identified as such.

4.1. Identification

   4.1. uniquely identifying entities.

                 Figure 6: EMAN Requirement for Identification

   There are only two truly mandatory items for an energy object to
   implement in EMAN. The first is a unique identity (UUID) (4.1). The
   second is an energy object type which can be device, component, power
   interface, or aggregate device (see Section 6.3). The EMAN Framework

specifies the UUID and includes an integer as an index for the energy
objects the device can report on.

In addition to the requirements, a widely useful feature for a device
is to report a URL for standard manufacturer data on the brand/model
of the device, represented as a string. Related to this, and also
not derivative of a requirement, is a string for each of the brand
(manufacturer) and model of the device.

4.2. Local Data

   5.1.1. configure, retrieve and report a textual name or a description
   5.1.2. retrieving and reporting context information ..., for example,
          tags associated with an entity that indicate the entity's role.
   5.1.3. retrieving and reporting the significance of entities within
          its context, for example, how important the entity is.
   5.1.4. retrieving and reporting Power priorities of entities. Power
          priorities indicate an order in which Power States of entities
          are changed.
   5.1.5. grouping entities. This can be achieved in multiple ways, for
          example, by providing means to tag entities, to assign them to
          domains, or to assign device types to them.

                Figure 7: EMAN Requirements for Local Data

Local data are those generated at the point of product use and so are
created for each building individually. The first item is a text
string for the name (5.1.1). The second is a list of keywords, each
of which will be a pair of strings, one representing a name or type
and the other one representing a value (5.1.2, .3, .4, .5). Since
none of these data types - role, priority, grouping, domain, or
device type - has been standardized, their representation and
encoding are determined locally. This flexible mechanism can be used
for various purposes, as the following examples illustrate using
different notation styles.

    role="switch"

    powerDownPriority: 6

    "lineofbusiness:Helpdesk"

    <location>South Wing</location>

    domain::office

4.3. Power Interfaces

      5.2.1. monitoring the list of Power Interfaces of a device.
      5.2.2. monitoring the operational mode of a Power Interface which is
             either "Power Inlet" or "Power Outlet".
      5.2.3. identifying the Power Outlet that provides the Power received
             at a Power Inlet.
      5.2.4. identifying the list of Power Inlets that receive the Power
             provided at a Power Outlet.
      5.2.5. monitoring the availability of Power at each Power Interface.
             ... indicates whether ... supply is switched on or off.
      5.2.6. monitoring for each Power Interface if it is in actual use.
             ... inlets ... device actually receives Power. ... outlets ...
             Power is actually provided.

                 Figure 8: EMAN Requirements for Power Interfaces

   PIs provide wiring topology and are the source of core energy and
   power data. A device provides PI information as a list of UUIDs of
   PIs it contains (5.2.1). Each PI provides a list of UUIDs of PIs
   that it knows it is wired to (5.2.3, .4). The power mode of a PI is
   indicated by the sign of the power value; positive for power flowing
   in, and negative for power flowing out (5.2.2). The availability of
   power at a PI is shown by its power state (see Section 4.6) (5.2.5).
   Actual flow of power is indicated by the power value being non-zero
   (5.2.6).

   Since wiring topology usually changes only infrequently, it would be
   burdensome for an EnMS to constantly refresh these data. To avoid
   this, a device can provide a timestamp of the last time there was a
   wiring topology change. Only when this changes does an EnMS need to
   rescan the topology. This feature does not arise from a listed
   requirement.

   Another capability not specified by any requirement is what
   directions of power flow a PI is capable of. This is indicated by an
   enumeration value of either inlet-only, outlet-only, or both.

4.4. Power, Static

      5.2.7. reporting the type of current (AC or DC) for each Power
             Interface as well as for a device.
      5.2.8. reporting the nominal voltage range for each Power Interface.
      5.2.9. reporting the nominal AC frequency for each Power Interface.
      5.2.10. reporting the number of AC phases for each Power Interface.

                  Figure 9: EMAN Requirements for Power, Static

These characteristics of power apply only to PIs and are static.
They do not change so can be set at the factory. They can be
represented by an enumeration (5.2.7), two strings (5.2.8, .9), and
an integers (5.2.10).

The requirements say that a device should also report its type of
current, but as some devices support both AC and DC simultaneously on
different PIs. There is not always a single type of current for a
whole device (as there is for a single PI). This is a shortcoming of
the requirements document, and the situation is covered by single PI
devices (see Section 2.2).

4.5. Power

   5.3.1. reporting the real power for each Power Interface as well as
      for an entity. ... includes ... the direction.
   5.3.2. reporting the corresponding time or time interval for which a
      Power value is reported.

                Figure 10: EMAN Requirements for Power

Power is represented as an integer and a power-of-ten multiplier and
the direction of power is by its sign (5.3.1; see Section 4.3). Any
kind of energy object can report power level data. The time of a
power reading is indicated by a corresponding timestamp.

4.6. Power State

   5.4.1. reporting the actual Power State of an entity.

                Figure 11: EMAN Requirement for Power State

The EMAN Framework [I-D.ietf-eman-framework] describes how EMAN
supports devices to have any number of power state series, and at any
time, a state can be reported for each. This representation is
suitable. Each energy object has a list of power state series it
supports and a corresponding state for each entry in the list
(5.4.1).

For PIs, the state indicates whether or not power can flow: 'on'
means power can flow, 'off' that power unable to flow, and 'sleep'
that only trickle power is available. Technologies which support
trickle power include PoE, USB, and UPAMD. Since the IEEE 1621 power
state series has these three states and only these, it is a natural
one to use for PIs.

4.7. Energy

5.5.1. reporting measured values of Energy and the direction of the
    Energy flow received or provided by an entity. ... report the
    Energy passing through each Power Interface.
5.5.2. reporting the time interval for which an Energy value is
    reported.

            Figure 12: EMAN Requirements for Energy

Energy is reported as an accumulated meter reading and a timestamp
(5.5.1). An EnMS subtracts one reading/timestamp from a later one to
get an interval of time and the energy flow during that time (5.5.2).
The sign of this energy value indicates whether net energy was
brought into a device or component (a positive value) or sent out of
it (a negative one) (5.5.1).

4.8. Control

6.1. setting Power States of entities.
6.2. switching Power supply off or turning Power supply on at Power
    Interfaces.

            Figure 13: EMAN Requirements for Control

A device or component may accept a command to set the power state
value to attempt a changing of the power state (6.1). For a PI, this
turns supply on and off (6.2; see Section 4.6). No new data elements
are introduced by these requirements.

4.9. Reporting

7.1. an entity to report information on another entity.
7.2. reporting the identity of other entities on which information is
    reported. ... For entities that report on one or more other
    entities.
7.4. reporting the complete list of all those entities on which
    Energy-related information can be reported. ... For entities that
    report on one or more other entities.

            Figure 14: EMAN Requirements for Reporting

Each device must be able to report the UUIDs of each device it can
report for, including itself (7.2). An EnMS can request data for any
listed device (7.1, .4). It can also request data for any PI or
component of any listed device.

4.10. Data Summary

The basic requirements above result in the following data elements
for a fully compliant implementatios. Note that a minimal
implementation would only require the first two data elements. In
the table below, the first column lists which type of energy objects
are relevant: device, PI, or component. The second column lists the
requirement level, mandatory, recommended, or optional.

```
EOs R Kind            Data
--- - -------------- ------------------------------------------------
DPC m Identification Index (within the device)
DPC m                UUID (for unique identification globally)
D   o                URL for brand/model specifications
D C o                Brand, model (strings)
DPC m Local Data     Text name
DPC o                Keyword list (strings)
D   r Power Interf.  List of PIs in device (UUIDs)
 P  r                List of PIs known to be connected to PI (UUIDs)
 P  r                Timestamp of last change to wiring topology
 P  m Power, Static  Type of current (AC or DC)
 P  m                Nominal voltage range, AC frequency, AC phases
DPC r Power          Timestamp of power reading
DPC r                Numeric value and exponent for power in W
DPC m Power State    List of supported power state series
DPC m                Current power state (for each supported series)
DPC m Energy         Timestamp of energy reading
DPC m                Numeric value and exponent for energy in Wh
D   r Reporting      List of devices that can be reported on (UUIDs)
```

Figure 15: Summary of Basic EMAN Data

A key point about the above table is that it is modest in size and
minimally burdensome for both implementers and those who use the EMAN
data in real buildings.

5. How the ER Framework Fulfills Advanced EMAN Requirements

   This section describes how the ER Framework fulfills the rest of the
   requirements in the EMAN Requirements draft. Each subsection begins
   with the requirements listed, abbreviated, as from Section 7. Most
   devices will not implement any of these, and many EnMS may not as
   well. That said, for those devices and buildings where these are
   relevant, these provide important capability. Many readers can skip
   this section entirely. For most of these, content from the EMAN
   Framework draft is suitable.

5.1. Identification, Advanced

   4.2. indicating whether identifiers ... are persistent across a
        re-start.
   4.3. indicate any change of entity identifiers.
   4.4. re-using entity identifiers from existing standards including
        ... the Entity MIB module ... the LLDP MIB module ... the
        LLDP-MED MIB module ... and the ... the Power Ethernet MIB. ...
        Generic means for re-using other entity identifiers must be
        provided.

          Figure 16: EMAN Requirements for Identification, Advanced

   These require a flag for identifier persistance (4.2), a timestamp of
   the last entity identifier change (4.3), and a list of identifiers
   each consisting of a pair of strings, one string representing a type
   and the other one representing a value (4.4).

5.2. Power, Advanced

   5.3.3. means to indicate the method how these values have been
          obtained.
   5.3.4. reporting the accuracy of reported Power and Energy values.
   5.3.5. reporting the actual voltage and actual current for each Power
          Interface as well as for a device. In case of AC Power supply ...
          per phase.
   5.3.6. creating notifications if Power values of an entity rise above
          or fall below given thresholds.
   5.3.7. reporting the complex power for each Power Interface and for
          each phase at a Power Interface.
   5.3.8. reporting the actual AC frequency for each Power Interface.
   5.3.9. reporting the Total Harmonic Distortion (THD) of voltage and
          current for each Power Interface.
   5.3.10. reporting the impedance of Power supply for each Power
           Interface. In case of AC Power supply ... per phase.

            Figure 17: EMAN Requirements for Power, Advanced

   The treatment of these requirements in the EMAN Framework draft is
   suitable. Note that 5.3.3, .4, and .6 apply to any energy object,
   but the rest only apply to PIs.

5.3. Power State, Advanced

   5.4.2. retrieving the list of all potential Power States of an
          entity.
   5.4.3. supporting multiple Power State sets simultaneously at an
          entity.
   5.4.4. retrieving the list of all Power State sets supported by an
          entity.

   5.4.5. retrieving the list of all potential Power States of an entity
          for each supported Power State set.
   5.4.6. retrieving the typical Power for each supported Power State.
   5.4.7. monitoring statistics per Power State including the total time
          spent in a Power State, the number of times each state was
          entered and the last time each state was entered.
   5.4.8. generating a notification when the actual Power State of an
          entity changes.

              Figure 18: EMAN Requirements for Power State

   The treatment of these requirements in the EMAN Framework draft is
   suitable.

5.4. Energy, Advanced

   5.5.3. reporting the received and provided Energy for each individual
          Power State.

              Figure 19: EMAN Requirement for Energy, Advanced

   The treatment of these requirements in the EMAN Framework draft is
   suitable, except for the notion of distinct values for consumed,
   provided, and net consumption (see Section 6.2).

5.5. Battery

   5.6.1. reporting the current charge ... in units of milliampere hours
   5.6.2. reporting the charging state (charging, discharging, etc.)
   5.6.3. reporting the number of completed charging cycles
   5.6.4. reporting the actual capacity
   5.6.5. reporting the actual temperature
   5.6.6. reporting static properties ... including the nominal
          capacity, the number of cells, the nominal voltage, and the ...
          technology.
   5.6.7. generating a notification when the charge ... decreases
          below a given threshold.
   5.6.8. generating a notification when the number of charging cycles
          ... exceeds a given threshold.
   5.6.9. meeting requirements 5.6.1 to 5.6.8 for each individual
          battery contained in a single entity.

               Figure 20: EMAN Requirements for Batteries

   The treatment of these requirements in the Battery MIB draft
   [I-D.ietf-eman-battery-mib] is suitable.

5.6. Time-series Data

   5.7.1. reporting time series of Energy values. If ... comparison ...
      between multiple entities ... is required, then time
      synchronization ... must be provided.
   5.7.2. supporting alternative interval types. Requirement 5.5.2
      applies to every reported time value.
   5.7.3. should ... reporting the number of values of a time series
      that can be stored.

           Figure 21: EMAN Requirements for Time-series data

   Time-series data is of the same form as single-value data for power
   and energy, notably a timestamp plus the relevant value. Thus,
   reporting is of a list of these (5.7.1). Every interval type can be
   derived from pairs of single-value data, by subtracting the time and
   energy values (5.7.2). For sequential windows, adjacent values are
   compared; for overlapping windows, non-adjacent values are compared.
   The energy object must be able to report the number of available time
   series data slots (5.7.3). A device must be able to accept a setting
   of time from an EnMS for time syncronization (5.7.1), though for
   security purposes, this might affect only EMAN reporting and not
   change the global system clock of the device. To implement all this,
   a device must be able to accept commands to initiate time-series data
   collection, including the requested time interval.

5.7. Reporting, Advanced

   7.3. reporting the list of all entities from which contributions are
      included in an accumulated value.
   7.5. indicating which Energy-related information can be reported for
      which of those entities. ... For entities that report on one or
      more other entities.

           Figure 22: EMAN Requirements for Reporting, Advanced

   This framework includes the concept of an aggregate device (7.3; see
   Section 6.3). No new dta elements are required for this. The method
   to implement 7.5 is TBD.

5.8. Proxy Control

   8.1.1. an Energy Management System to send Power State control
      commands to an entity that concern the Power States of [other]
      entities.
   8.1.2. reporting the identities of the entities for which the
      reporting entity has means to control their Power States.
   8.1.3. an entity to report the list of all entities for which it can

control the Power State.
8.1.4. an entity that receives commands controlling its Power State
        from other entities to report the list of all those entities.

        Figure 23: EMAN Requirements for Proxy Control

Only a device can control the power state of another entity, but any
energy object can be controlled this way. An energy object that can
be controlled by a proxy must be able to provide a list of
controllers (8.1.4). A device must be able to provide a list of
energy objects that it is capable to control the power state of
(8.1.2). 8.1.3 appears to be a duplicate of 8.1.2. To use this
feature, a device receives a power state set command for a UUID that
it is not itself.

5.9. Security

9.1. provide privacy, integrity, and authentication mechanisms for
        all actions addressed in Sections 5 - 8. ... must meet the
        security requirements elaborated in Section 1.4 of [RFC3411].
9.2. allow for isolation of entities that that are not sufficiently
        secure to operate on the public Internet.

        Figure 24: EMAN Requirements for Security

The treatment of these requirements in the EMAN Framework draft is
suitable.

6. EnMS Operation

The full utility and application of EMAN can only be understood with
some understanding of how an EnMS operates. This section covers
several of these. This section does not introduce new data items.

6.1. Incomplete data

Section 3 noted that data on PI connectivity need not be complete to
be useful, and need not be complete on every device to be
comprehensive. The ability to report and use incomplete data is
generic in EMAN. This is particularly valuable for including the
many legacy devices with few or no EMAN capabilities.

Note that for the case in the right side of Figure 5, the data
available from Device-A and Device-B need not be the same. There may
be a different combination of PIs and components available on them,
and for each energy object, one of the devices may know and report
more data than the other.

Devices may lack power or energy detail for several PIs but still be able to report data about the device total.

6.2. Separately Tracking Energy Flows by Direction

In some cases it may be desired to separately track the flows of energy into or out of a device, PI, or component for those that support bi-directional power flow. When the direction of power flow changes within a reporting period, the opposite flows cancel each other out, giving a correct indication of the net flow, but lacking the tracking of the total in each direction. Doing this is readily accomplished by having two PIs in place of one, or two components in place of one. Each is used only for a single direction of power flow. In this way, the separate directions are correctly tracked, and by summation, the net flow can be calculated. Implementations which desire this (batteries being a prime example) can do this, and those that do not simply use a single PI or component. This ability adds no data fields or complexity to the ER framework.

6.3. Aggregate Devices

An aggregate device is a not a real device, but an information construct to represent the summation of data across a set of energy objects. It has a UUID as any device does, has no power interfaces (since it does not interact with power in the real world), and has one or more components. Each component of an aggregate device can be any type of energy object. Commonly, all objects in an aggregate object will be of the same type (all devices, all PIs, or all components) but this is not required. A device may or may not report any data for the UUIDs in the aggregate object components; all that is required is to have the component UUIDs. The aggregate device concept introduces no new data items to the EMAN structure.

In producing an aggregate result, EMAN does not define how a reporting device should treat incomplete data, any mismatch in the alignment of timestamps among the objects to be aggregated, or whether power states should be aggregated or how such an aggregation would be interpreted. A device is free to do whatever it likes in this regard. The only requirement is that power and energy reported reflect a summation of all the constituent energy objects. Since this is an aggregate device and not an aggregate PI, advanced power data are not reported.

Common examples of aggregate objects might be all servers in a data center rack, all fans in a data center, all devices in a single room, all lights in a building, or all devices under the financial responsibility of a single tenant.

6.4. Proxy Control

   Control of the power state of one device by a second device is not
   common, but does occur. A prime example is the ability to wake (or
   turn on) a sleeping PC with the "Wake On LAN" technology. Another
   example would be when the device being controlled is not an IP
   device.

6.5. Cloud Devices

   Another use of EMAN to accomplish real-world needs is through the use
   of "cloud" devices. Like an aggregate device, these are not a single
   real-world object (or not necessarily one), but do correspond to real
   wiring topology. An example is a tree-structured wiring topology
   with circuit breaker panels. An individual circuit breaker, or
   collection of them, can be represented as a cloud device. The meter
   device shows a PI on it wired to an inlet PI on the cloud device, and
   individual end-use devices (or downstream circuit breakers) are wired
   to an outlet PI on the cloud device. The cloud device has a UUID,
   but is not a device on the network, and no data are reported from the
   cloud device. Since no data are reported, there is no cloud device
   of energy object.

   What this does is correctly represent the data known about the wiring
   topology, while hiding some potentially-existing (but unknown)
   complexity inside the cloud. It adds no complexity to the EMAN model
   and only adds the creation and use of a single UUID in PIs as
   appropriate. It is flexible in allowing for a wire variety of
   topologies of devices, clouds, and devices operating as meters.

6.6. External Power Meters

   A device which is only a power meter is modeled exactly as any other
   device. It is modeled as a device that has an inlet power interface
   receiving power and one or more outlet power interfaces providing
   power. The fact that a device may consume none of the energy that
   passes through it is not relevant to EMAN, and in this case, the
   quantitative values on both PIs will be identical except of opposite
   sign.

   A DC-powered blade server in a chassis has its own identity on the
   network and if it reports for itself, it is a separate device, not a
   component of the chassis. Similarly, a PoE-powered device can be
   modeled as either a separate device, or a component of the powering
   device.

7. EMAN Requirements Summary

This section summarizes the content of the EMAN Requirements
document, lists all specific requirements. The text is excerpted for
brevity, in one case text in [brackets] is added, and "..."
indicates text deleted within a requirement. The words "should" and
"must" occur in the introductory text; these are often redundant to
the actual requirements and so are not included in this list.
Requirements generally begin with "The standard must provide means
for ..."; this text is omitted. The headings below are not part of
the requirements draft.

Requirements: Basic

Identification
4.1. uniquely identifying entities.

Local Data
5.1.1. configure, retrieve and report a textual name or a description
5.1.2. retrieving and reporting context information ..., for example,
    tags associated with an entity that indicate the entity's role.
5.1.3. retrieving and reporting the significance of entities within
    its context, for example, how important the entity is.
5.1.4. retrieving and reporting Power priorities of entities. Power
    priorities indicate an order in which Power States of entities
    are changed.
5.1.5. grouping entities. This can be achieved in multiple ways, for
    example, by providing means to tag entities, to assign them to
    domains, or to assign device types to them.

Power Interfaces
5.2.1. monitoring the list of Power Interfaces of a device.
5.2.2. monitoring the operational mode of a Power Interface which is
    either "Power Inlet" or "Power Outlet".
5.2.3. identifying the Power Outlet that provides the Power received
    at a Power Inlet.
5.2.4. identifying the list of Power Inlets that receive the Power
    provided at a Power Outlet.
5.2.5. monitoring the availability of Power at each Power Interface.
    ... indicates whether ... supply is switched on or off.
5.2.6. monitoring for each Power Interface if it is in actual use.
    ... inlets ... device actually receives Power. ... outlets ...
    Power is actually provided.

Power, Static
5.2.7. reporting the type of current (AC or DC) for each Power
    Interface as well as for a device.
5.2.8. reporting the nominal voltage range for each Power Interface.
5.2.9. reporting the nominal AC frequency for each Power Interface.
5.2.10. reporting the number of AC phases for each Power Interface.

Power
5.3.1. reporting the real power for each Power Interface as well as
       for an entity. ... includes ... the direction.
5.3.2. reporting the corresponding time or time interval for which a
       Power value is reported.

Power State
5.4.1. reporting the actual Power State of an entity.

Energy
5.5.1. reporting measured values of Energy and the direction of the
       Energy flow received or provided by an entity. ... report the
       Energy passing through each Power Interface.
5.5.2. reporting the time interval for which an Energy value is
       reported.

Control
6.1. setting Power States of entities.
6.2. switching Power supply off or turning Power supply on at Power
     Interfaces.

Reporting
7.1. an entity to report information on another entity.
7.2. reporting the identity of other entities on which information is
     reported. ... For entities that report on one or more other
     entities.
7.4. reporting the complete list of all those entities on which
     Energy-related information can be reported. ... For entities that
     report on one or more other entities.

Requirements: Advanced

Identification, Advanced
4.2. indicating whether identifiers ... are persistent across a
     re-start.
4.3. indicate any change of entity identifiers.
4.4. re-using entity identifiers from existing standards including
     ... the Entity MIB module ... the LLDP MIB module ... the
     LLDP-MED MIB module ... and the ... the Power Ethernet MIB. ...
     Generic means for re-using other entity identifiers must be
     provided.

Power, Advanced
5.3.3. means to indicate the method how these values have been
       obtained.
5.3.4. reporting the accuracy of reported Power and Energy values.
5.3.5. reporting the actual voltage and actual current for each Power
       Interface as well as for a device. In case of AC Power supply ...

per phase.
5.3.6. creating notifications if Power values of an entity rise above
       or fall below given thresholds.
5.3.7. reporting the complex power for each Power Interface and for
       each phase at a Power Interface.
5.3.8. reporting the actual AC frequency for each Power Interface.
5.3.9. reporting the Total Harmonic Distortion (THD) of voltage and
       current for each Power Interface.
5.3.10. reporting the impedance of Power supply for each Power
        Interface. In case of AC Power supply ... per phase.

Power State
5.4.2. retrieving the list of all potential Power States of an
       entity.
5.4.3. supporting multiple Power State sets simultaneously at an
       entity.
5.4.4. retrieving the list of all Power State sets supported by an
       entity.
5.4.5. retrieving the list of all potential Power States of an entity
       for each supported Power State set.
5.4.6. retrieving the typical Power for each supported Power State.
5.4.7. monitoring statistics per Power State including the total time
       spent in a Power State, the number of times each state was
       entered and the last time each state was entered.
5.4.8. generating a notification when the actual Power State of an
       entity changes.

Energy, Advanced
5.5.3. reporting the received and provided Energy for each individual
       Power State.

Battery
5.6.1. reporting the current charge ... in units of milliampere hours
5.6.2. reporting the charging state (charging, discharging, etc.)
5.6.3. reporting the number of completed charging cycles
5.6.4. reporting the actual capacity
5.6.5. reporting the actual temperature
5.6.6. reporting static properties ... including the nominal
       capacity, the number of cells, the nominal voltage, and the ...
       technology.
5.6.7. generating a notification when the charge ... decreases
       below a given threshold.
5.6.8. generating a notification when the number of charging cycles
       ... exceeds a given threshold.
5.6.9. meeting requirements 5.6.1 to 5.6.8 for each individual
       battery contained in a single entity.

Time-series data

5.7.1. reporting time series of Energy values. If ... comparison ...
       between multiple entities ... is required, then time
       synchronization ... must be provided.
5.7.2. supporting alternative interval types. Requirement 5.5.2
       applies to every reported time value.
5.7.3. should ... reporting the number of values of a time series
       that can be stored.

   Reporting
7.3. reporting the list of all entities from which contributions are
     included in an accumulated value.
7.5. indicating which Energy-related information can be reported for
     which of those entities. ... For entities that report on one or
     more other entities.

   Proxy Control
8.1.1. an Energy Management System to send Power State control
       commands to an entity that concern the Power States of [other]
       entities.
8.1.2. reporting the identities of the entities for which the
       reporting entity has means to control their Power States.
8.1.3. an entity to report the list of all entities for which it can
       control the Power State.
8.1.4. an entity that receives commands controlling its Power State
       from other entities to report the list of all those entities.

   Security
9.1. provide privacy, integrity, and authentication mechanisms for
     all actions addressed in Sections 5 - 8. ... must meet the
     security requirements elaborated in Section 1.4 of [RFC3411].
9.2. allow for isolation of entities that that are not sufficiently
     secure to operate on the public Internet.

8. Outstanding Issues

   The following are questions that need further consideration by the
   EMAN working group.

8.1. How to Address Requirement 7.5

   "7.5. indicating which Energy-related information can be reported for
   which of those entities. ... For entities that report on one or more
   other entities." An EnMS could simply query all data for an energy
   object and observe what are available. A more efficient mechanism
   could be defined. Regardless, any value should have "unknown" as an
   option.

8.2. EMAN Framework Content

   Incorporate details from the EMAN Framework draft and the Battery MIB
   draft where they are to be included substantially as-is.

8.3. Metering

   Add text to section 3 explaining how measurement done at different
   power interfaces leads to conclusions about device metering.

8.4. PI Capability

   Per Section 4.3, should there be a data element to indicate if a PI
   can be only an inlet, only an outlet, or either?

8.5. Data Representation for Section 5.1

   Consider the syntax of the items to be represented and whether the
   proposal for a single text string is sufficient. Possibly provide
   examples.

8.6. Move Requirement Satisfaction Details to Appendix

   Consider whether the details in sections 4 and 5 on how specific
   requirements are satisfied should be moved to an appendix. Sections
   4 and 5 would retain, and possibly expand, their explanation of the
   framework content itself. This would make the document more readable
   (and shorter, not counting the appendix) while retaining
   documentation of the relationships of framework content to
   requirements in the appendix.

9. Security Considerations

   This memo currently does not impose any security considerations.

10. IANA Considerations

   This memo has no actions for IANA.

11. Acknowledgements

   This memo was inspired by discussions with many people, including
   (but not limited to) Juergen Quittek, Rolf Winter, Benoit Claise,
   John Parello, and Mouli Chandramouli.

12. Informative References

   [RFC3411] Harrington, D., Presuhn, R., and B. Wijnen, "An
              Architecture for Describing Simple Network Management
              Protocol (SNMP) Management Frameworks", STD 62, RFC 3411,
              December 2002.

   [RFC6988] Quittek, J., Chandramouli, M., Winter, R., Dietz, T., and
              B. Claise, "Requirements for Energy Management", RFC 6988,
              September 2013.

   [I-D.ietf-eman-framework]
              Parello, J., Claise, B., Schoening, B., and J. Quittek,
              "Energy Management Framework", draft-ietf-eman-
              framework-11 (work in progress), October 2013.

   [I-D.ietf-eman-applicability-statement]
              Schoening, B., Chandramouli, M., and B. Nordman, "Energy
              Management (EMAN) Applicability Statement", draft-ietf-
              eman-applicability-statement-04 (work in progress),
              October 2013.

   [I-D.ietf-eman-battery-mib]
              Quittek, J., Winter, R., and T. Dietz, "Definition of
              Managed Objects for Battery Monitoring", draft-ietf-eman-
              battery-mib-09 (work in progress), July 2013.

   [I-D.nordman-eman-energy-perspective]
              Nordman, B., "Energy perspective on applicability", draft-
              nordman-eman-energy-perspective-01 (work in progress),
              March 2011.

   [I-D.norwin-energy-consider]
              Nordman, B. and R. Winter, "Considerations for Power and
              Energy Management", draft-norwin-energy-consider-02 (work
              in progress), March 2011.

   [I-D.quittek-eman-reference-model]
              Quittek, J., Nordman, B., and R. Winter, "Reference Model
              for Energy Management", draft-quittek-eman-reference-
              model-03 (work in progress), October 2011.

   [IEEE-802.3af]

IEEE 802.3 Working Group, ., "IEEE Std 802.3af-2003 - IEEE
Standard for Information technology - Telecommunications
and information exchange between systems - Local and
metropolitan area networks - Specific requirements - Part
3: Carrier Sense Multiple Access with Collision Detection
(CSMA/CD) Access Method and Physical Layer Specifications
- Amendment: Data Terminal Equipment (DTE) - Power via
Media Dependent Interface (MDI)", July 2003.

[IEEE-802.3at]
IEEE 802.3 Working Group, ., "IEEE Std 802.3at-2009 - IEEE
Standard for Information technology - Telecommunications
and information exchange between systems - Local and
metropolitan area networks - Specific requirements - Part
3: Carrier Sense Multiple Access with Collision Detection
(CSMA/CD) Access Method and Physical Layer Specifications
- Amendment: Data Terminal Equipment (DTE) - Power via
Media Dependent Interface (MDI) Enhancements", October
2009.

Author's Address

Bruce Nordman
Lawrence Berkeley National Laboratory
1 Cyclotron Road
Berkeley 94720
US

Phone: +1 510 486 7089
Email: bnordman@lbl.gov