



## Energy Monitoring and Management of Networked Objects using SNMP and Role-Specific Sparse Tables

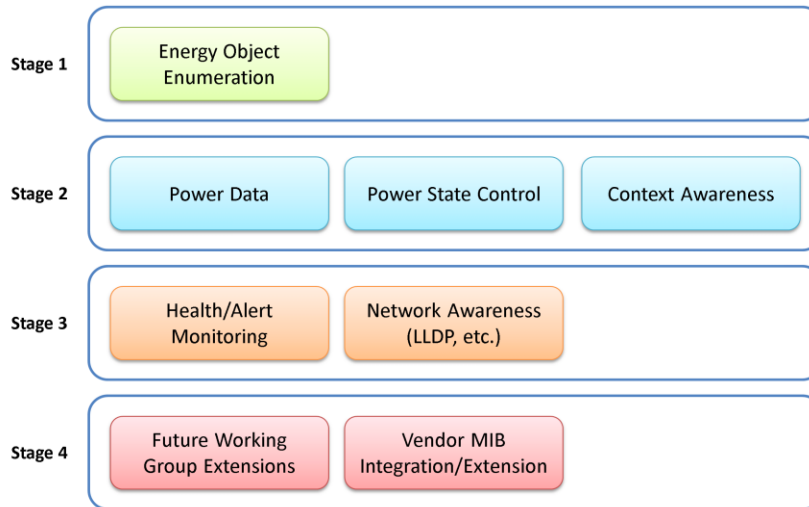
Benoit Claise  
(on behalf of Chris Verges)

IETF 80 - Prague, Czech Republic  
March 30, 2011

The following presentation describes how management of energy objects can be accomplished using role-specific sparse tables. This model leverages object-oriented design patterns to increase flexibility and decrease maintenance overhead.

Before we begin, I (Chris Verges) would like to give a special thanks to Benoit Claise for presenting this document in my absence.

# Vision of EMAN Eco-System



EMAN is more than just this one working group. The work currently being structure has the potential to become an overarching energy management framework beyond just servers and the use cases we are considering today. My goal in today's proposal is to setup a flexible structure that is generic to anything that utilizes power and can be extended by other working groups and vendors into the future.

There are four basic stages in my mind for EMAN maturity. The first stage is identifying the objects that are to be managed. The second stage is to add basic behaviors and information around them. The third stage adds more "frills" and complexity, but enables a wider range of applications. The fourth stage hands the combined schemas off to the world for adoption and expansion.

The next slides will briefly touch on the core blocks in Stages 1 thru 3 that are in EMAN's scope today and will describe how they are segmented.

# Elements



Energy Object Enumeration

*Power Data*

*Power State Control*

*Context Awareness*

*Health/Alert Monitoring*

*Network Awareness*

- **The object definition itself**
- A table/list of objects
- **Independent of any “decorations”**
- **Structural relationships between objects**
- Three Phase (3PH) objects modeled as aggregates of Single Phase (1PH) objects, one “child” per phase
- Minimalist approach intended to increase flexibility to extend through sparse tables
- High cohesion, low coupling

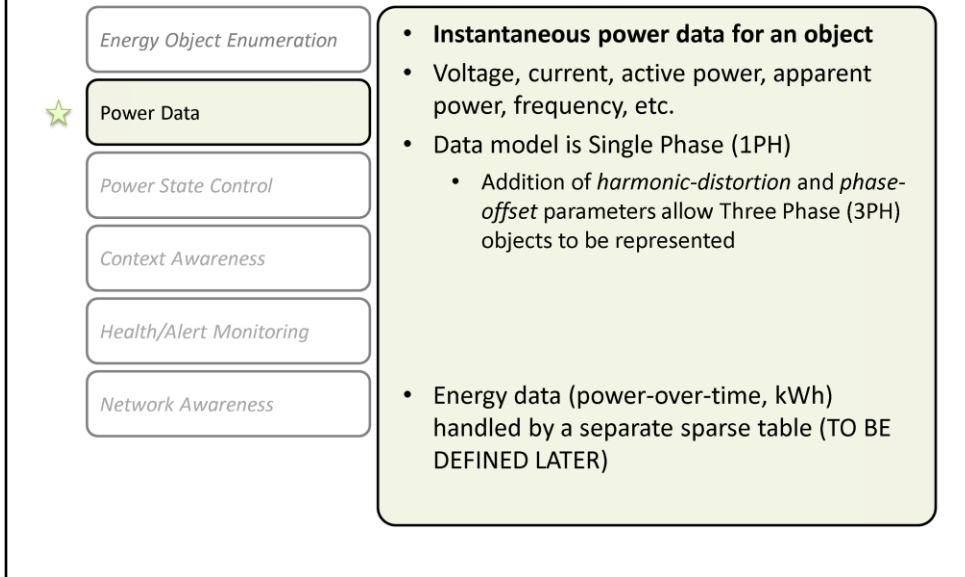
Listing the objects that will be managed is the core problem to be tackled. Some drafts have based this on ENTITY-MIB; others create a monolithic structure. We will get into the specific differences in a later slide, but the primary difference is scope.

In this proposal, a list of objects will be created and given a unique index on the system. These objects can be physical in nature (power supplies, outlets, servers, batteries) or logical in nature (outlet “gangs”, virtual machines, software processes). The relationship between objects can be described by a graph rather than a tree, allowing for various combinations and arrangements. By accepting objects of any type, the framework is generic enough to grow into use cases not yet considered.

All “attributes” or “properties” that are not critical to the identity of an object are pushed to sparse tables. This keeps the core structure clean and increases extensibility.

Complex objects, such as Three Phase meters, are represented as the parent object (the Three Phase meter aggregate object) and a series of child objects (three or four separate Single Phase meter objects). This allows for a simplified data model in sparse table extensions later.

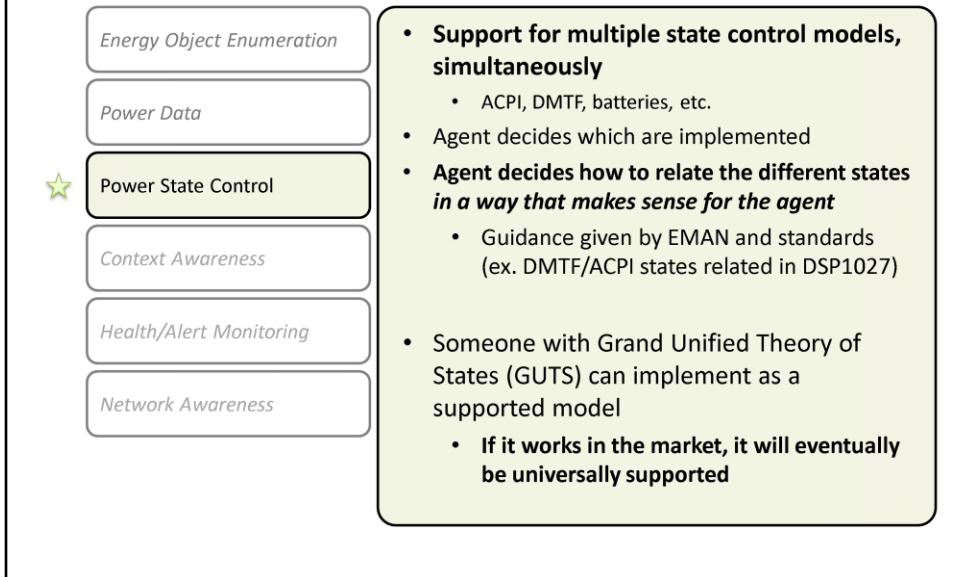
# Elements



The “Power Data” sparse table focuses on providing instantaneous power measurements for current, voltage, frequency, active power, etc. Historical power data (energy, or kilowatt-hours) is to be handled by a separate sparse table.

As mentioned before, Three Phase objects are de-aggregated into separate Single Phase objects. Since Single Phase offers a “base” data model that can be used by all power types known today (DC, Single Phase AC, and Three Phase AC), creating a data model around this concept allows for maximum flexibility with maximum simplicity.

# Elements

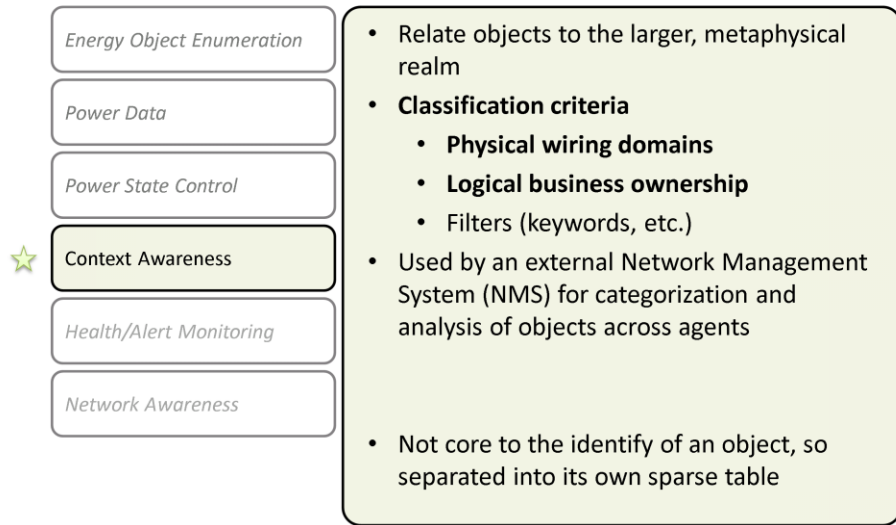


The issue of control models is a contentious one on the mailing list. Rather than trying to develop a “Grand Unified Theory” of power state control, I propose that we acknowledge the existing models and allow end users the maximum flexibility to use whatever model they want.

Each of the control models can be represented simultaneously, and it is up to the implement agent to determine what the best way is for the agent’s implementation to link ACPI to DMTF to Battery Control, etc. We should not presume to be able to predict all of the use cases at this time. This proposal creates separate SNMP OIDs for each control model: ACPI gets its own OID, DMTF gets its own, Battery Control gets its own, etc. If an agent has an ACPI state machine, it can implement the ACPI OID. Likewise, if it has a DMTF state machine, it can implement DMTF. This allows an agent to clearly enumerate what capabilities it has, with no ambiguity.

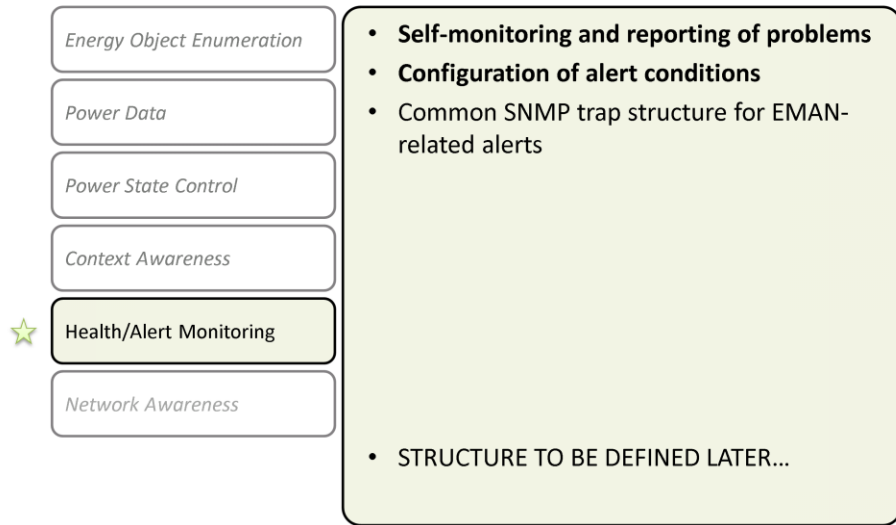
The benefit of this model is that if someone DOES create a “Grand Unified Theory of Power States,” it can be added as a separate control model. If this new model proves to be beneficial, it will be universally adopted across the industry and the others will be deprecated. However, forcing implementers to use only one model right now might limit or slow down the adoption of the EMAN MIB in the industry.

# Elements



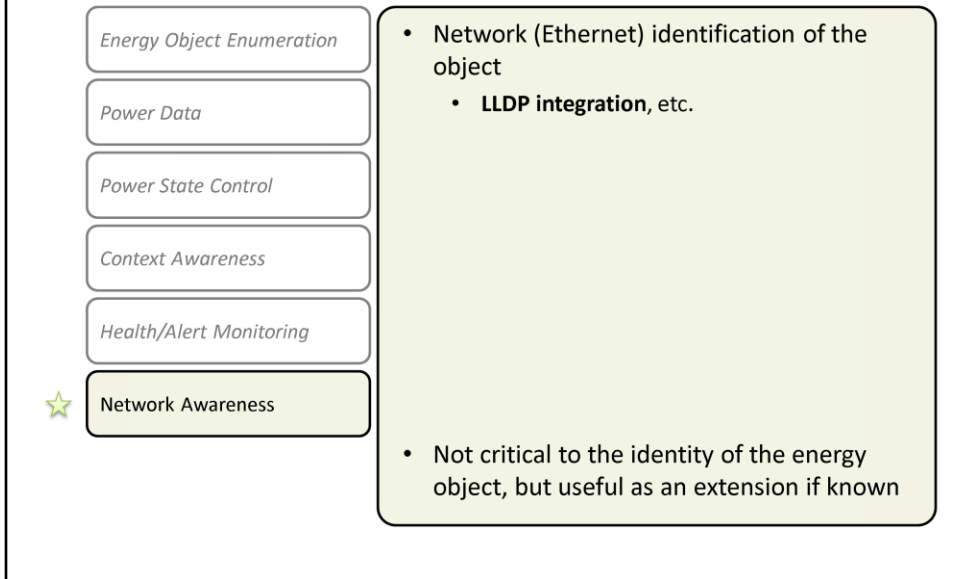
Context awareness is the metaphysical information that relates an object to the larger picture. It is similar in scope to the proposed “Domain” and “Keywords,” but creates a single interface where physical wiring and logical categorization are explicitly called out.

# Elements



Little has been mentioned in EMAN of the notion for monitoring an energy object's health. Several objects already monitor health, and so supporting a common interface to report this data and configure alert thresholds is something that is an obvious extension. I do not propose to tackle it now, but mention it as an example of a use case that has not yet been considered by EMAN.

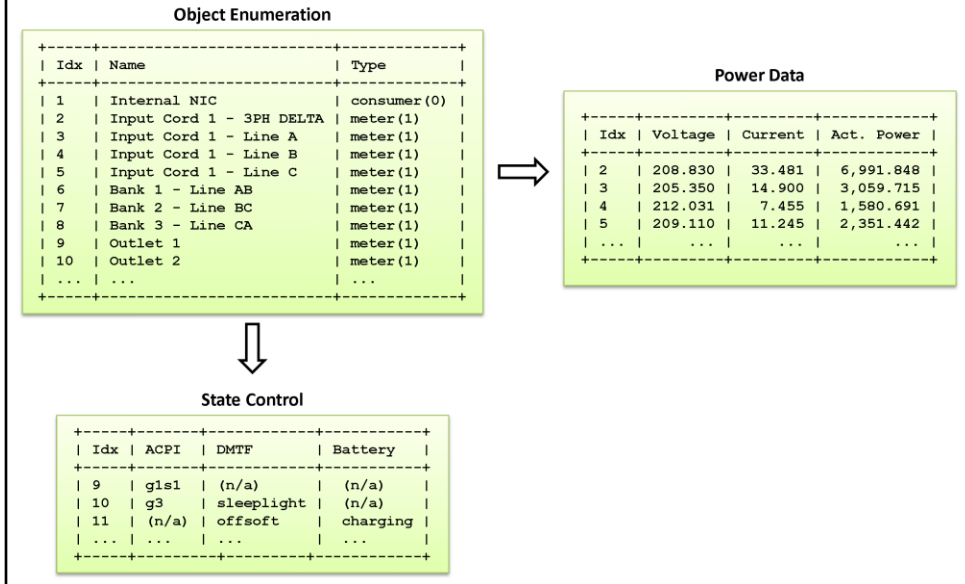
# Elements



Finally, network awareness allows an entity to relate who it is to some object identifier on the network, such as LLDP. While useful if that information is known, it is not always known nor is critical to the identity of what an object is. As a result, such an item would be implemented using a role-specific sparse table to accomplish the goal.



# Example

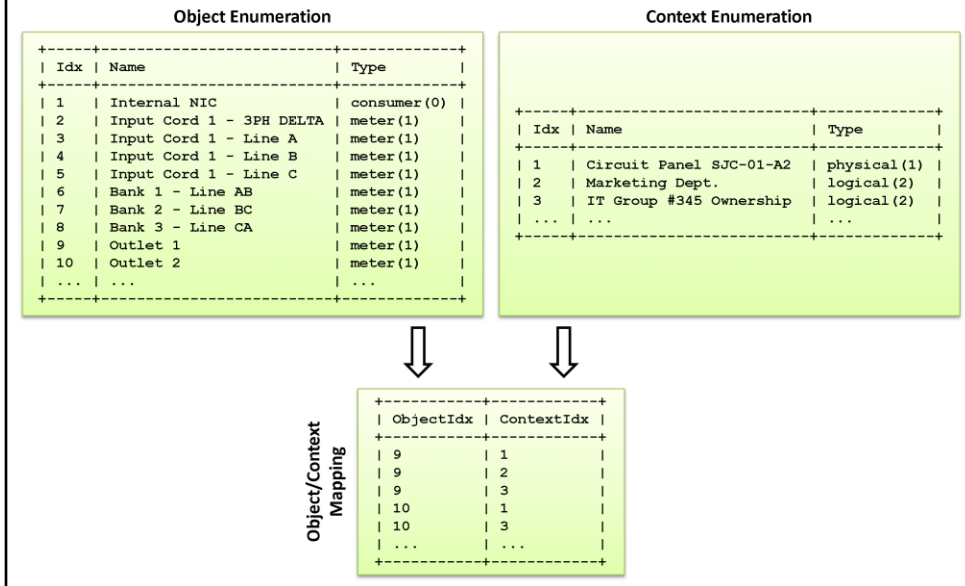


This is an example of how the various tables relate to one other. Here, we see a Power Distribution Unit described. The unit has a Three Phase input cord, which has child objects listed. (The relationship between parents and children is not shown due to space constraints.) Each object is given a unique index value, which is then used by sparse tables that extend the functionality in role-specific ways.

The Power Data sparse table is shown for the Three Phase input cord (Index 2) and the multiple Single Phase child meters that comprise it. The detailed data exists if an Network Management Systems requires it, but can be summarized into a single phase equivalency if not needed.

Also shown is the State Control sparse table. Separate objects exist for the various control models: ACPI, DMTF, Battery, etc. As future and/or better control models are developed, additional OIDs can be added to this structure and existing ones can be deprecated as needed.

# Example



This is an example of how an entity can be mapped to a context. It is a simple table mapping, where membership in a context by an object is simple to derive for an NMS.

## Relationship to ENTITY-MIB

- Modeling virtual objects
  - “Ganged outlets”
    - Physical entities shared across logical contexts
  - Virtual machines managed by the host OS
  - Software processes
- No common index for both physical entities and logical contexts
- As a result, RFC4133 may not make sense for “core”
  - Good to reference through a “decoration”

(See <http://www.ietf.org/mail-archive/web/eman/current/msg00275.html> for more)

There has been a detailed discussion on the mailing list related to the use of ENTITY-MIB as the Energy Object Enumeration block in the model proposed here. The following is a summary of that discussion.

The main goal from the “core” object list is to provide a common index value that EMAN can use for sparse table extension. Here is an example of how this might work using ENTITY-MIB for a smart PDU. If a PDU is defined as a series of physical entities (inputs, banks, and outlets) and two logical contexts are created (the PDU itself and one outlet “gang”), how should EMAN extend off this setup? Using entPhysicalIndex is a natural desire, but ignores the outlet “gang” object that is only defined as a logical context. We might be able to use entLogicalIndex, but then we lose the granularity of information available from the physical entity breakdown. We could use some odd combination of both, but that removes any structural constraints to help relate the data together. In the end, mixing both physical entities and logical contexts requires “hacks” to make work, either conceptually or structurally.

As a result, the conclusion discussed on the mailing list is to refer to ENTITY-MIB as a “decorator” of an object, but to allow EMAN to establish its own index identifier. If desired, an implementing agent can use the ENTITY-MIB index as the EMAN index, but the MIB will not enforce this.

## Relationship to Existing Drafts

**draft-ietf-eman-energy-aware-mib**

**draft-claise-energy-monitoring-mib**

- Core structure and attribute decoration tightly linked
  - Coupling/cohesion problem
  - Limits options for extension in future working groups
- Complex data structure
  - Separate tables for 1PH, 3PH Wye, and 3PH Delta
- Power states are unilateral
- Propose merging with draft-verges to fix coupling/cohesion issues

The flexible structure described here is a result of analyzing the monolithic structure described in the -energy-aware and -energy-monitoring drafts. The structure in these drafts tends to be difficult to extend and change, making maintenance and customer use difficult.

All of the functionality described in these drafts has been included (or can be included) in the structure described in draft-verges, so I am proposing that we merge these documents going forward, using the object-oriented structure described in draft-verges as a basis.

# Proposals

1. Merge draft-verges with other drafts
  - draft-ietf-eman-energy-aware-mib
  - draft-claise-energy-monitoring-mib
2. Maintain object-oriented structure
3. Consider that our understanding of the problem has improved since the charter
  - Are items 3-5 still the best breakdown?

Special thanks to Benoit for presenting in my absence! (Chris Verges)

In conclusion, there are three proposals on the table:

First, merge draft-verges with the other two drafts.

Second, use the object-oriented structure that is described in draft-verges as the basis for the changes going forward.

Third, consider that our understanding of the EMAN problem has significantly improved since the charter was written. The charter defines three sets of MIB documents, but does this breakdown continue to make the best sense?

Thank you.