

Entity MIB Working Group  
Internet Draft  
Document: [draft-ietf-entmib-state-04.txt](#)  
Category: Standards Track  
Expiration Date: January 2005

S. Chisholm  
Nortel Networks  
D. Perkins  
SNMPinfo  
July 2004

## Entity State MIB

### Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of [Section 10 of RFC2026](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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

The list of current Internet-Drafts can be accessed at

<http://www.ietf.org/ietf/1id-abstracts.txt>

The list of Internet-Draft Shadow Directories can be accessed at

<http://www.ietf.org/shadow.html>.

### 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 extensions to the Entity MIB to provide information about the state of physical entities.

### Table of Contents

1. The Internet-Standard Management Framework
2. Entity State
  - 2.1. Hierarchical State Management
  - 2.2. Entity Redundancy
  - 2.3. Physical Entity Users
  - 2.4. Physical Class Behaviour
3. Relationship to Other MIBs

- 3.1. Relation to Interfaces MIB
- 3.2. Relation to Alarm MIB
- 3.3. Relation to Bridge MIB

3.4. Relation to Host Resource MIB

4. Textual Conventions

5. Definitions

6. Security Considerations

7. Intellectual Property

8. Authors' Addresses

9. Acknowledgements

[10.](#) **References**

[11.](#) **Full Copyright Statement**



## 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 a MIB module 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. Entity State

The goal in adding state objects to the Entity MIB [[RFC2737](#)] is to define a useful subset of the possible state attributes that could be tracked for a given entity that both fit into the state models such as those used in the Interfaces MIB [[RFC2863](#)] as well as leverage existing well-deployed models. The entStateTable contains state objects that are a subset of the popular ISO/OSI states that are also defined in ITU's X.731 specification [[X.731](#)]. Objects are defined to capture administrative, operational and usage states. In addition there are further state objects defined to provide additional information for these three basic states.

Administrative state indicates permission to use or prohibition against using the entity and is imposed through the management services.

Operational state indicates whether or not the entity is physically installed and working. Note that unlike the ifOperStatus [[RFC2863](#)], this operational state is independent of the administrative state.

Usage state indicates whether or not the entity is in use at a specific instance, and if so, whether or not it currently has spare capacity to serve additional users. In the context of this MIB, the user is equivalent to an entity, so this term is substituted. This state refers to the ability of the entity to service other entities within its containment hierarchy.

Alarm state indicates whether or not there are any alarms active against the entity. In addition to those alarm status defined in X.731 [[X.731](#)], warning and indeterminate status are also defined to provide a more complete mapping to the Alarm MIB [Alarm-MIB].

Standby state indicates whether the entity is currently running as hot standby, cold standby or is currently providing service.

The terms state and status are used interchangeably in this memo.

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](#)].

## **[2.1](#) Hierarchical State Management**

Physical entities exist within a containment hierarchy. Physical containment is defined by the `entPhysicalContainedIn` object[RFC2737]. This raises some interesting issues not addressed in existing work on state management [[X.731](#)].

There are two types of state for an entity:

- 1) The state of the entity independent of the states of its parents and children in its containment hierarchy. This is often referred to as raw state.
- 2) The state of the entity, as it may be influenced by the state of its parents and children. This is often referred to as computed state.

All state objects in this memo are raw state.

## **[2.2](#) Entity Redundancy**

While this memo is not attempting to address the entire problem space around redundancy, the `entStateStandby` object provides an important piece of state information for entities, which helps identify which pieces of redundant equipment are currently providing service, and which are waiting in either hot or cold standby mode.

## **[2.3](#) Physical Entity Users**

There are three ways to define the 'user' of a physical entity

1. Direct Containment in physical hierarchy
2. Anywhere in physical hierarchy
3. As defined by a means outside the scope of this MIB. This could include logical interfaces that could run on a port, software that could run on a module, etc.

Administrative, operational, alarm and standby state use all three definitions of 'user'. Usage state only supports the concept of

direct containment to simplify implementations of this object.



## **2.4 Physical Class Behaviour**

This MIB makes no effort to standardize on the behaviours and characteristics of the various physical classes [[RFC2737](#)], but rather how this information is reported. In looking at real-world products, items within the same physical class vary substantially. The MIB has therefore provided guidance on how to support objects where a particular instance of a physical class can not support part or all of a particular state.

## **3 Relation to other MIBs**

### **3.1 Relationship to the Interfaces MIB**

The Interfaces MIB [[RFC2863](#)] defines the `ifAdminStatus` object, which has states of up, down and testing and the `ifOperStatus` object, which has states of up, down, testing, unknown, dormant, notPresent and lowerLayerDown.

An `ifAdminStatus` of 'up' is equivalent to setting the `entStateAdmin` object to 'unlocked'. An `ifAdminStatus` of 'down' is equivalent to setting the `entStateAdmin` object to either 'locked' or 'shuttingDown', depending on a systems interpretation of 'down'.

An `ifOperStatus` of 'up' is equivalent to an `entStateOper` value of 'enabled'. An `ifOperStatus` of 'down' due to operational failure is equivalent to an `entStateOper` value of 'disabled'. An `ifOperStatus` of 'down' due to being administratively disabled is equivalent to an `entStateAdmin` value of 'locked' and an `entStateOper` value of either 'enabled' or 'disabled' depending on whether there are any known issues that would prevent the entity from becoming operational when its `entStateAdmin` is set to 'unlocked'. An `ifOperStatus` of 'unknown' is equivalent to an `entStateOper` value of 'unavailable'. The `ifOperStatus` values of 'testing' and 'dormant' are not explicitly supported by this MIB, but the state objects will be able to reflect other aspects of the entities administrative and operational state. The `ifOperStatus` values of 'notPresent' and 'lowerLayerDown' are in some ways computed states and so are therefore not supported in this MIB. They can though be computed by examining the states of entities within this objects containment hierarchy and other available related states.

### **3.2 Relation to Alarm MIB**

The `entStateAlarm` object indicates whether or not there are any active alarms against this entity. If there are active alarms, then

the alarmActiveTable in the Alarm MIB [Alarm MIB] should be searched for alarmActiveResourceId that match this entPhysicalIndex.

Alternatively, if the alarmActiveTable is queried first and an active alarm with a value of alarmActiveResourceId that matches this entPhysicalIndex is found, then entStateAlarm can be used to quickly determine if there are additional active alarms against this physical entity.

### **3.3 Relation to Bridge MIB**

For entities of physical type of 'port' that support the dot1dStpPortEnable object in the Bridge MIB [[RFC1493](#)], a value of 'enabled' is equivalent to setting the entStateAdmin object to 'unlocked'. Setting dot1dStpPortEnable to 'disabled' is equivalent to setting the entStateAdmin object to 'locked'.

### **3.4 Relation to the Host Resources MIB**

The hrDeviceStatus object in the Host Resources MIB [[RFC2790](#)] provides an operational state for devices. For entities that logically correspond to the concept of a device, a value of 'unknown' for hrDeviceStatus corresponds to an entStateOper value of 'unavailable'. A value of 'running' corresponds to an entStateOper value of 'enabled'. A value of 'warning' also corresponds to an entStateOper value of 'enabled', but with appropriate bits set in the entStateAlarm object to indicate the alarms corresponding to the unusual error condition detected. A value of 'testing' or 'down' is equivalent to an entStateOper value of 'disabled'.

## **4. Textual Conventions**

```
ENTITY-STATE-TC DEFINITIONS ::= BEGIN
```

```
IMPORTS
```

```
    MODULE-IDENTITY, mib-2          FROM SNMPv2-SMI
    TEXTUAL-CONVENTION              FROM SNMPv2-TC;
```

```
entityStateTc MODULE-IDENTITY
```

```
    LAST-UPDATED "200407190000Z"
```

```
    ORGANIZATION "IETF Entity MIB Working Group"
```

```
    CONTACT-INFO
```

```
        "General Discussion: entmib@ietf.org
```

```
        To Subscribe:
```

```
            http://www.ietf.org/mailman/listinfo/entmib
```

```
            http://www.ietf.org/html.charters/entmib-charter.html
```

```
        Sharon Chisholm
```

```
        Nortel Networks
```

```
        PO Box 3511 Station C
```

```
        Ottawa, Ont. K1Y 4H7
```

Canada  
schishol@nortelnetworks.com

David T. Perkins  
548 Qualbrook Ct  
San Jose, CA 95110  
USA  
Phone: 408 394-8702  
dperkins@snmpinfo.com"

## DESCRIPTION

"This MIB defines a state textual conventions.

Copyright (C) The Internet Society 2004. This version  
of this MIB module is part of RFC yyyy; see the RFC  
itself for full legal notices."

-- RFC Ed.: replace yyyy with actual RFC number & remove  
-- this note

REVISION "200407190000Z"

## DESCRIPTION

"Initial version, published as RFC yyyy."

-- RFC-Editor assigns yyyy

::= { mib-2 XX } -- to be assigned by IANA

AdminState ::= TEXTUAL-CONVENTION

STATUS current

## DESCRIPTION

" Represents the various possible administrative states.

A value of 'locked' means the resource is administratively  
prohibited from use. A value of 'shuttingDown' means that  
usage is administratively limited to current instances of  
use. A value of 'unlocked' means the resource is not  
administratively prohibited from use. A value of  
'unavailable' means that this resource is unable to  
report administrative state."

## REFERENCE

"ITU Recommendation X.731, 'Information Technology - Open  
Systems Interconnection - System Management: State  
Management Function', 1992"

SYNTAX INTEGER

{  
unavailable(1),  
locked(2),  
shuttingDown(3),  
unlocked(4)  
}

OperState ::= TEXTUAL-CONVENTION

STATUS current

## DESCRIPTION

" Represents the possible values of operational states.

A value of 'disabled' means the resource is totally inoperable. A value of 'enabled' means the resource is partially or fully operable. A value of 'testing'

means the resource is currently being tested and cannot therefore report whether it is operational or not. A value of 'unavailable' means that this resource is unable to report operational state. "

## REFERENCE

"ITU Recommendation X.731, 'Information Technology - Open Systems Interconnection - System Management: State Management Function', 1992"

## SYNTAX INTEGER

```
{
  unavailable (1),
  disabled(2),
  enabled(3),
  testing (4)
}
```

UsageState ::= TEXTUAL-CONVENTION

STATUS current

## DESCRIPTION

" Represents the possible values of usage states. A value of 'idle' means the resource is servicing no users. A value of 'active' means the resource is currently in use and it has sufficient spare capacity to provide for additional users. A value of 'busy' means the resource is currently in use, but it currently has no spare capacity to provide for additional users. A value of 'unavailable' means that this resource is unable to report usage state."

## REFERENCE

"ITU Recommendation X.731, 'Information Technology - Open Systems Interconnection - System Management: State Management Function', 1992"

## SYNTAX INTEGER

```
{
  unavailable (1),
  idle(2),
  active(3),
  busy(4)
}
```

AlarmStatus ::= TEXTUAL-CONVENTION

STATUS current

## DESCRIPTION

"Represents the possible values of alarm status. An Alarm [[ALARM-MIB](#)] is a persistent indication of an error or warning condition."

When no bits of this attribute are set, then none of the value of under repair is set, the resource is currently being repaired, which depending on the implementation, may make the other values in this bit



string unreliable.

When the value of 'critical' is set, one or more critical alarms are active against the resource. When the value of 'major' is set, one or more major alarms are active against the resource. When the value of 'minor' is set, one or more minor alarms are active against the resource. When the value of 'warning' is set, one or more warning alarms are active against the resource. When the value of 'indeterminate' is set, one or more alarms whose of perceived severity cannot be determined are active against this resource.

A value of 'unavailable' means that this resource is unable to report alarm state."

#### REFERENCE

"ITU Recommendation X.731, 'Information Technology - Open Systems Interconnection - System Management: State Management Function', 1992"

#### SYNTAX

#### BITS

```
{
  unavailable (0),
  underRepair(1),
  critical(2),
  major(3),
  minor(4),
  -- The following are not defined in X.733
  warning (5),
  indeterminate (6)
}
```

StandbyStatus ::= TEXTUAL-CONVENTION

STATUS current

#### DESCRIPTION

" Represents the possible values of standby status.

A value of 'hotStandby' means the resource is not providing service, but it will be immediately able to take over the role of the resource to be backed-up, without the need for initialization activity, and will contain the same information as the resource to be backed up. A value of 'coldStandby' means that the resource is to back-up another resource, but will not be immediately able to take over the role of a resource to be backed up, and will require some initialization activity. A value of 'providingService' means the resource is providing service. A value of

'unavailable' means that this resource is unable to  
report standby state."

REFERENCE

"ITU Recommendation X.731, 'Information Technology - Open

```
        Systems Interconnection - System Management: State
        Management Function', 1992"
SYNTAX          INTEGER
{
    unavailable (1),
    hotStandby(2),
    coldStandby(3),
    providingService(4)
}
```

END

## 5. Definitions

ENTITY-STATE-MIB DEFINITIONS ::= BEGIN

IMPORTS

```
    MODULE-IDENTITY, OBJECT-TYPE, NOTIFICATION-TYPE, mib-2
        FROM SNMPv2-SMI
    TEXTUAL-CONVENTION, DateAndTime
        FROM SNMPv2-TC
    MODULE-COMPLIANCE, OBJECT-GROUP, NOTIFICATION-GROUP
        FROM SNMPv2-CONF
    entPhysicalIndex
        FROM ENTITY-MIB
    AdminState, OperState, UsageState, AlarmStatus, StandbyStatus
        FROM ENTITY-STATE-TC;
```

entityStateMIB MODULE-IDENTITY

LAST-UPDATED "200407190000Z"

ORGANIZATION "IETF Entity MIB Working Group"

CONTACT-INFO

" General Discussion: [entmib@ietf.org](mailto:entmib@ietf.org)

To Subscribe:

<http://www.ietf.org/mailman/listinfo/entmib>

<http://www.ietf.org/html.charters/entmib-charter.html>

Sharon Chisholm  
Nortel Networks  
PO Box 3511 Station C  
Ottawa, Ont. K1Y 4H7  
Canada  
[schishol@nortelnetworks.com](mailto:schishol@nortelnetworks.com)

David T. Perkins  
548 Qualbrook Ct  
San Jose, CA 95110  
USA

Phone: 408 394-8702  
dperkins@snmpinfo.com  
"

## DESCRIPTION

"This MIB defines a state extension to the Entity MIB.

Copyright (C) The Internet Society 2004. This version of this MIB module is part of RFC yyyy; see the RFC itself for full legal notices."

-- RFC Ed.: replace yyyy with actual RFC number & remove  
-- this note

REVISION "200407190000Z"

## DESCRIPTION

"Initial version, published as RFC YYYY."

-- RFC-Editor assigns yyyy

::= { mib-2 XX } -- to be assigned by IANA

-- Entity State Objects

entStateObjects OBJECT IDENTIFIER ::= { entityStateMIB 1 }

entStateTable OBJECT-TYPE

SYNTAX SEQUENCE OF EntStateEntry

MAX-ACCESS not-accessible

STATUS current

## DESCRIPTION

"A table of information about state/status of entities.

This is a sparse augment of the entPhysicalTable. Entries appear in this table for values of entPhysicalClass [[RFC2737](#)] that in this implementation are able to report any of the state or status stored in this table.

"

::= { entStateObjects 1 }

entStateEntry OBJECT-TYPE

SYNTAX EntStateEntry

MAX-ACCESS not-accessible

STATUS current

## DESCRIPTION

"State information about this physical entity."

INDEX { entPhysicalIndex }

::= { entStateTable 1 }

EntStateEntry ::= SEQUENCE {

entStateLastChanged DateAndTime,

entStateAdmin AdminState,

entStateOper OperState,

entStateUsage UsageState,

```
entStateAlarm      AlarmStatus,  
entStateStandby    StandbyStatus  
}
```

## entStateLastChanged OBJECT-TYPE

SYNTAX DateAndTime

MAX-ACCESS read-only

STATUS current

## DESCRIPTION

"The value of this object is the date and time when the value of any of entStateAdmin, entStateOper, entStateUsage, entStateAlarm, or entStateStandby changed for this entity.

If there has been no change since the last re-initialization of the local system, this object contains the date and time of local system initialization. If there has been no change since the entity was added to the local system, this object contains the date and time of the insertion"

::= { entStateEntry 1 }

## entStateAdmin OBJECT-TYPE

SYNTAX AdminState

MAX-ACCESS read-write

STATUS current

## DESCRIPTION

"This object refers to an entities administrative permission to service both other entities within its containment hierarchy as well other users of its services defined by means outside the scope of this MIB.

The administrative state for this entity. Setting this object to 'notSupported' will result in an 'inconsistentValue' error. For entities that do not support administrative state, all set operations will result in an 'inconsistentValue' error

Some physical entities exhibit only a subset of the remaining administrative state values. Some entities cannot be locked, and hence this object exhibits only the 'unlocked' state. Other entities can not be shutdown gracefully, and hence this object does not exhibit the 'shuttingDown' state. A value of 'inconsistentValue' will be returned if attempts are made to set this object to values not supported by its administrative model."

::= { entStateEntry 2 }

entStateOper OBJECT-TYPE  
SYNTAX        OperState  
MAX-ACCESS   read-only



STATUS current

DESCRIPTION

"The operational state for this entity.

Note that unlike the state model used within the Interfaces MIB [[RFC2863](#)], this object does not follow the administrative state. An administrative state of down does not predict an operational state of disabled.

A value of 'disabled' means that an entity is totally inoperable and unable to provide service both to entities within its containment hierarchy, or to other receivers of its service as defined in ways outside the scope of this MIB.

A value of 'enabled' means that an entity is fully or partially operable and able to provide service both to entities within its containment hierarchy, or to other receivers of its service as defined in ways outside the scope of this MIB.

Note that some implementations may not be able to accurately report entStateOper while the entStateAdmin object has a value other than 'unlocked'. In these cases, this object MUST have a value of 'unavailable'."

::= { entStateEntry 3 }

entStateUsage OBJECT-TYPE

SYNTAX UsageState

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The usage state for this entity.

This object refers to an entity's ability to service more physical entities in a containment hierarchy. A value of 'idle' means this entity is able to contain other entities but that no other entity is currently contained within this entity.

A value of 'active' means that at least one entity is contained within this entity, but that it could handle more. A value of 'busy' means that the entity is unable to handle any additional entities being contained in it.

Some entities will exhibit only a subset of the usage state values. Entities that are unable to ever

service any entities within a containment hierarchy will always have a usage state of 'busy'. Some entities will only ever be able to support one entity within its

containment hierarchy and will therefore only exhibit values of 'idle' and 'busy'."

```
::= { entStateEntry 4 }
```

entStateAlarm OBJECT-TYPE

SYNTAX AlarmStatus

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The alarm status for this entity. It does not include the alarms raised on child components within its containment hierarchy.

Note that this differs from 'indeterminate' which means that that alarm state is supported and there are alarms against this entity, but the severity of some of the alarms is not known.

If no bits are set, then this entity supports reporting of alarms, but there are currently no active alarms against this entity.  
"

```
::= { entStateEntry 5 }
```

entStateStandby OBJECT-TYPE

SYNTAX StandbyStatus

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The standby status for this entity.

Some entities will exhibit only a subset of the remaining standby state values. If this entity cannot operate in a standby role, the value of this object will always be 'providingService'."

```
::= { entStateEntry 6 }
```

-- Notifications

```
entStateNotifications OBJECT IDENTIFIER ::= { entityStateMIB 0 }
```

entStateOperEnabled NOTIFICATION-TYPE

OBJECTS { entStateAdmin,  
          entStateAlarm  
}

STATUS current

DESCRIPTION

"An entStateOperEnabled Notification signifies that the SNMP entity, acting in an agent role, has detected that

the entStateOper object for one of its entities has left the 'disabled' state and transitioned into the 'enabled' state.

The entity this notification refers can be identified by extracting the entPhysicalIndex from one of the variable bindings. The entStateAdmin and entStateAlarm varbinds may be examined to find out additional information on the administrative state at the time of the operation state change as well to find out whether there were any known alarms against the entity at that time that may explain why the physical entity has become operationally disabled."

```
::= { entStateNotifications 1 }
```

entStateOperDisabled NOTIFICATION-TYPE

OBJECTS { entStateAdmin,  
          entStateAlarm }

STATUS current

DESCRIPTION

"An entStateOperDisabled Notification signifies that the SNMP entity, acting in an agent role, has detected that the entStateOper object for one of its entities has left the 'enabled' state and transitioned into the 'disabled' state.

The entity this notification refers can be identified by extracting the entPhysicalIndex from one of the variable bindings. The entStateAdmin and entStateAlarm varbinds may be examined to find out additional information on the administrative state at the time of the operation state change as well to find out whether there were any known alarms against the entity at that time that may have affect on the physical entity's ability to stay operationally enabled."

```
::= { entStateNotifications 2 }
```

-- Conformance and Compliance

entStateConformance OBJECT IDENTIFIER ::= { entityStateMIB 3 }

entStateCompliances OBJECT IDENTIFIER  
                      ::= { entStateConformance 1 }

entStateCompliance MODULE-COMPLIANCE

STATUS current

DESCRIPTION

"The compliance statement for systems supporting the Entity State MIB."

MODULE -- this module

MANDATORY-GROUPS {  
  entStateGroup

```
}  
GROUP          entStateNotificationsGroup  
DESCRIPTION  
    "This group is optional."
```

```
OBJECT entStateAdmin
  MIN-ACCESS read-only
  DESCRIPTION
    "Write access is not required."
  ::= { entStateCompliances 1 }

entStateGroups OBJECT IDENTIFIER ::= { entStateConformance 2 }

entStateGroup OBJECT-GROUP
  OBJECTS {
    entStateLastChanged,
    entStateAdmin,
    entStateOper,
    entStateUsage,
    entStateAlarm,
    entStateStandby
  }
  STATUS current
  DESCRIPTION
    "Standard Entity State group."
  ::= { entStateGroups 1}

entStateNotificationsGroup NOTIFICATION-GROUP
  NOTIFICATIONS {
    entStateOperEnabled,
    entStateOperDisabled
  }
  STATUS current
  DESCRIPTION
    "Standard Entity State Notification group."
  ::= { entStateGroups 2}

END
```

## 6. Security Considerations

There is one management object defined in this MIB that has a MAX-ACCESS clause of read-write. The object 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 object is defined with a MAX-ACCESS clause of read-write: entStateAdmin.

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 (entities) that have legitimate rights to indeed GET or SET (change/create/delete) them.

Note that setting the entStateAdmin to 'locked' or 'shuttingDown' can cause disruption of services ranging from those running on a port to those on an entire device, depending on the type of entity. Access to this object should be properly protected.

Access to the objects defined in this MIB allows one to figure out what the active and standby resources in a network are. This information can be used to optimize attacks on networks so even read-only access to this MIB should be properly protected.

## **[7. Intellectual Property](#)**

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

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

## **[8. Authors' Addresses](#)**

Sharon Chisholm

Nortel Networks  
PO Box 3511, Station C  
Ottawa, Ontario, K1Y 4H7

Canada  
Email: schishol@nortelnetworks.com

David T. Perkins  
548 Qualbrook Ct  
San Jose, CA 95110  
USA  
Phone: 408 394-8702  
Email: dperkins@snmpinfo.com

## **9. Acknowledgments**

This document is a product of the Entity MIB Working Group.

## **10. References**

### **10.1 Normative**

- [ALARM-MIB] Chisholm, S., Romascanu, D., "Alarm MIB",  
[draft-ietf-disman-alarm-mib-18.txt](#), February 2004
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate  
Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC2578] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J.,  
Rose, M. and S. Waldbusser, "Structure of Management  
Information Version 2 (SMIV2)", STD 58, [RFC 2578](#), April  
1999.
- [RFC2579] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J.,  
Rose, M. and S. Waldbusser, "Textual Conventions for  
SMIV2", STD 58, [RFC 2579](#), April 1999.
- [RFC2580] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J.,  
Rose, M. and S. Waldbusser, "Conformance Statements for  
SMIV2", STD 58, [RFC 2580](#), April 1999.
- [RFC2737] McCloghrie, K., Bierman, A., "Entity MIB (Version 2)",  
December 1999.
- [X.731] ITU Recommendation X.731, "Information Technology - Open  
Systems Interconnection - System Management: State  
Management Function", 1992

### **10.2 Informative References**

- [RFC1493] Decker, E., Langille, P., Rijsinghani, A., McCloghrie, K.,

"Definitions of Managed Objects for Bridges", [RFC 1493](#),  
July 1993

- [RFC2790] Waldbusser, S., Grillo, P., "Host Resources MIB",  
[RFC 2790](#), March 2000
- [RFC2863] McCloghrie, K., Kastenholz, F., "The Interfaces Group  
MIB using SMIV2", [RFC2863](#), June 2000
- [RFC3410] Case, J., Mundy, R., Partain, D. and B. Stewart,  
"Introduction and Applicability Statements for Internet-  
Standard Management Framework", [RFC 3410](#), December 2002.

## **11. Full Copyright Statement**

Copyright (C) The Internet Society (2004). All Rights Reserved.

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

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

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

