Network Working Group
Request for Comments: 1515

D. McMaster
SynOptics Communications, Inc.
K. McCloghrie
Hughes LAN Systems, Inc.
S. Roberts
Farallon Computing, Inc.

September 1993

Definitions of Managed Objects for IEEE 802.3 Medium Attachment Units (MAUs)

Status of this Memo

Abstract

This document defines a portion of the Management Information Base (MIB) for use with network management protocols in TCP/IP-based internets. In particular, it defines objects for managing IEEE 802.3 Medium Attachment Units (MAUs).

Table of Contents

1. The Network Management Framework

The Internet-standard Network Management Framework consists of three components. They are:

STD 16, RFC 1155 [1] which defines the SMI, the mechanisms used for describing and naming objects for the purpose of management. STD 16, RFC 1212 [7] defines a more concise description mechanism, which is wholly consistent with the SMI.

STD 17, RFC 1213 [4] which defines MIB-II, the core set of managed objects for the Internet suite of protocols.

STD 15, RFC 1157 [3] which defines the SNMP, the protocol used for network access to managed objects.

The Framework permits new objects to be defined for the purpose of experimentation and evaluation.

2. Object Definitions

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. Objects in the MIB are defined using the subset of Abstract Syntax Notation One (ASN.1) defined in the SMI. In particular, each object object type is named by an OBJECT IDENTIFIER, an administratively assigned name. The object type together with an object instance serves to uniquely identify a specific instantiation of the object. For human convenience, we often use a textual string, termed the descriptor, to refer to the object type.

3. Overview

Instances of the object types defined in this document represent attributes of an IEEE 802.3 MAU. Several types of MAUs are defined in the IEEE 802.3/ISO 8802-3 CSMA/CD standard [9].

These MAUs may be connected to IEEE 802.3 repeaters or to 802.3 (Ethernet-like) interfaces. For convenience this document refers to these devices as "repeater MAUs" and "interface MAUs."

The definitions presented here are based on Draft 5 of Section 20 of IEEE P802.3p, "Layer Management for 10 Mb/s Medium Attachment Units

(MAUs), Section 20" [10] dated 11 July 1992.

3.1. Terminology

Refer to Section 3.1.2 of [13] for simple definitions of the terms "repeater," "port," and "MAU" as used in the context of this document. For a more complete and precise definition of these terms, refer to Section 9 of [9].

3.2. Structure of MIB

Objects in this MIB are arranged into MIB groups. Each MIB group is organized as a set of related objects.

3.2.1. The Repeater MAU Basic Group Definitions

This group contains all repeater MAU-related configuration, status, and control objects. Implementation of the dot3RpMauBasicGroup is mandatory for MAUs attached to repeaters.

3.2.2. The Interface MAU Basic Group Definitions

This group contains all interface MAU-related configuration, status, and control objects. Implementation of the dot3IfMauBasicGroup is mandatory for MAUs attached to interfaces.

3.2.3. The Broadband MAU Basic Group Definitions

This group contains all broadband-specific MAU-related configuration objects. Implementation of the dot3BroadMauBasicGroup is mandatory for 10BROAD36 MAUs, and is not appropriate for other types of MAUs.

3.3. Relationship to Other MIBs

It is assumed that an agent implementing this MIB will also implement (at least) the ’system’ group defined in MIB-II [4]. The following sections identify other MIBs that such an agent should implement.

3.3.1. Relationship to the ’system’ group

In MIB-II, the ’system’ group is defined as being mandatory for all systems such that each managed entity contains one instance of each object in the ’system’ group. Thus, those objects apply to the entity even if the entity’s sole functionality is management of a MAU.

3.3.2. Relationship to the ’interfaces’ group

The sections of this document that define interface MAU-related objects specify an extension to the ’interfaces’ group of MIB-II [4]. An agent implementing these interface-MAU related objects must also implement the ’interfaces’ group of MIB-II. The value of the same as the value of ’ifIndex’ used to instantiate the interface to which the given MAU is connected.

It is expected that an agent implementing the interface-MAU related objects in this MIB will also implement the Ethernet-like Interfaces MIB [11].

(Note that repeater ports are not represented as interfaces in the sense of MIB-II’s ’interfaces’ group. See section 3.4.2 of the repeater MIB [12] for more details.)

3.3.3. Relationship to the 802.3 Repeater MIB

The section of this document that defines repeater MAU-related objects specifies an extension to the 802.3 Repeater MIB defined in [13]. An agent implementing these repeater-MAU related objects must also implement the 802.3 Repeater MIB.

The values of ’rpMauGroupIndex’ and ’rpMauPortIndex’ used to instantiate a repeater MAU variable shall be the same as the values of ’rptrPortGroupIndex’ and ’rptrPortIndex’ used to instantiate the port to which the given MAU is connected.

3.4. Management of Internal MAUs

In some situations, a MAU can be "internal" -- i.e., its functionality is implemented entirely within a device. For example, a managed repeater may contain an internal repeater- MAU and/or an internal interface-MAU through which management communications originating on one of the repeater’s external ports pass in order to reach the management agent associated with the repeater. Such internal MAUs may or may not be managed. If they are managed, objects describing their attributes should appear in the appropriate MIB group -- dot3RpMauBasicGroup for internal repeater-MAUs and dot3IfMauBasicGroup for internal interface-MAUs.

4. Definitions

```
MAU-MIB DEFINITIONS ::= BEGIN


IMPORTS
    Counter                                FROM RFC1155-SMI
    OBJECT-TYPE                            FROM RFC-1212
    TRAP-TYPE                              FROM RFC-1215;


snmpDot3MauMgt OBJECT IDENTIFIER ::= { mib-2 26 }


--                      References
--
-- The following references are used throughout this MIB:
--
-- [RFC 1213]
--    refers to McCloghrie, K., and M. Rose, Editors,
--    Management Information Base for Network Management
--    of TCP/IP-based internets: MIB-II, STD 17, RFC 1213,
--    Hughes LAN Systems, Performance Systems International,
--    March 1991.
--
-- [RFC 1368]
--    refers to McMaster, D.,  and K. McCloghrie, Editors,
--    Definitions of Managed Objects for IEEE 802.3 Repeater
--    Devices, RFC 1368, SynOptics Communications, Hughes
--    LAN Systems, October 1992.
--
-- [IEEE 802.3 MAU Mgt]
--    refers to IEEE P802.3p, 'Layer Management for 10 Mb/s
--    Medium Access Unit (MAUs), Section 20,' Draft Supplement
--    to ANSI/IEEE 802.3, Draft 5, 11 July 1992.


--                      MIB Groups
--
--   The dot3RpMauBasicGroup is mandatory for MAUs attached to
--     repeaters.
--   The dot3IfMauBasicGroup is mandatory for MAUs attached to
--     DTEs (interfaces).
--   The dot3BroadMauBasicGroup is mandatory for broadband MAUs
--     attached to DTEs.


dot3RpMauBasicGroup
```

```
            OBJECT IDENTIFIER ::= { snmpDot3MauMgt 1 }
        dot3IfMauBasicGroup
            OBJECT IDENTIFIER ::= { snmpDot3MauMgt 2 }
        dot3BroadMauBasicGroup
            OBJECT IDENTIFIER ::= { snmpDot3MauMgt 3 }


        -- object identifiers for MAU types
        --  (see rpMauType and ifMauType for usage)
        dot3MauType
            OBJECT IDENTIFIER ::= { snmpDot3MauMgt 4 }
        dot3MauTypeAUI        -- no internal MAU, view from AUI
            OBJECT IDENTIFIER ::= { dot3MauType 1 }
        dot3MauType10Base5    -- thick coax MAU (per 802.3 section 8)
            OBJECT IDENTIFIER ::= { dot3MauType 2 }
        dot3MauTypeFoirl      -- FOIRL MAU (per 802.3 section 9.9)
            OBJECT IDENTIFIER ::= { dot3MauType 3 }
        dot3MauType10Base2    -- thin coax MAU (per 802.3 section 10)
            OBJECT IDENTIFIER ::= { dot3MauType 4 }
        dot3MauType10BaseT    -- UTP MAU (per 802.3 section 14)
            OBJECT IDENTIFIER ::= { dot3MauType 5 }
        dot3MauType10BaseFP   -- passive fiber MAU (per 802.3 section 16)
            OBJECT IDENTIFIER ::= { dot3MauType 6 }
        dot3MauType10BaseFB   -- sync fiber MAU (per 802.3 section 17)
            OBJECT IDENTIFIER ::= { dot3MauType 7 }
        dot3MauType10BaseFL   -- async fiber MAU (per 802.3 section 18)
            OBJECT IDENTIFIER ::= { dot3MauType 8 }
        dot3MauType10Broad36  -- broadband DTE MAU (per 802.3 section 11)
            -- note that 10BROAD36 MAUs can be attached to interfaces but
            -- not to repeaters
            OBJECT IDENTIFIER ::= { dot3MauType 9 }


        --
        --                    The Repeater MAU Basic Group
        --
        -- Implementation of the Repeater MAU Basic Group is mandatory
        -- for MAUs attached to repeaters.

        --
        -- The Basic Repeater MAU Table
        --

        rpMauTable OBJECT-TYPE
            SYNTAX    SEQUENCE OF RpMauEntry
            ACCESS    not-accessible
            STATUS    mandatory
            DESCRIPTION
```

```
                    "Table of descriptive and status information about
                    the MAU(s) attached to the ports of a repeater."
            ::= { dot3RpMauBasicGroup 1 }

        rpMauEntry OBJECT-TYPE
            SYNTAX    RpMauEntry
            ACCESS    not-accessible
            STATUS    mandatory
            DESCRIPTION
                    "An entry in the table, containing information
                    about a single MAU."
            INDEX    { rpMauGroupIndex, rpMauPortIndex, rpMauIndex }
            ::= { rpMauTable 1 }

        RpMauEntry ::=
            SEQUENCE {
                rpMauGroupIndex
                    INTEGER,
                rpMauPortIndex
                    INTEGER,
                rpMauIndex
                    INTEGER,
                rpMauType
                    OBJECT IDENTIFIER,
                rpMauStatus
                    INTEGER,
                rpMauMediaAvailable
                    INTEGER,
                rpMauMediaAvailableStateExits
                    Counter,
                rpMauJabberState
                    INTEGER,
                rpMauJabberingStateEnters
                    Counter
            }

        rpMauGroupIndex OBJECT-TYPE
            SYNTAX    INTEGER (1..1024)
            ACCESS    read-only
            STATUS    mandatory
            DESCRIPTION
                    "This variable uniquely identifies the repeater
                    group containing the port to which the MAU
                    described by this entry is connected."
            REFERENCE
                    "Reference RFC1368, rptrGroupIndex."
            ::= { rpMauEntry 1 }
        rpMauPortIndex OBJECT-TYPE
            SYNTAX    INTEGER (1..1024)
            ACCESS    read-only
            STATUS    mandatory
            DESCRIPTION
                    "This variable uniquely identifies the repeater
                    port within group rpMauGroupIndex to which the MAU
                    described by this entry is connected."
            REFERENCE
                    "Reference RFC 1368, rptrPortIndex."
            ::= { rpMauEntry 2 }

        rpMauIndex OBJECT-TYPE
            SYNTAX    INTEGER (1..9)
            ACCESS    read-only
            STATUS    mandatory
            DESCRIPTION
                    "This variable uniquely identifies the MAU
                    connected to port rpMauPortIndex within group
                    rpMauGroupIndex that is described by this entry."
            REFERENCE
                    "Reference IEEE 802.3 MAU Mgt, 20.2.3.2, aMAUID."
            ::= { rpMauEntry 3 }

        rpMauType OBJECT-TYPE
            SYNTAX    OBJECT IDENTIFIER
            ACCESS    read-only
            STATUS    mandatory
            DESCRIPTION
                    "This object identifies the 10 Mb/s baseband MAU
                    type.  An initial set of MAU types are defined
                    above.  The assignment of OBJECT IDENTIFIERs to
                    new types of MAUs is managed by the IANA.  If the
                    MAU type is unknown, the object identifier

                        unknownMauType OBJECT IDENTIFIER ::= { 0 0 }

                    is returned.  Note that unknownMauType is a
                    syntactically valid object identifier, and any
                    conformant implementation of ASN.1 and the BER
                    must be able to generate and recognize this
                    value."
            REFERENCE
                    "Reference IEEE 802.3 MAU Mgt, 20.2.3.2,
                    aMAUType."
            ::= { rpMauEntry 4 }

        rpMauStatus OBJECT-TYPE
```

```
            SYNTAX    INTEGER {
                          other(1),
                          unknown(2),
                          operational(3),
                          standby(4),
                          shutdown(5),
                          reset(6)
                      }
            ACCESS    read-write
            STATUS    mandatory
            DESCRIPTION
                    "The current state of the MAU.  This object may be
                    implemented as a read-only object by those agents
                    and MAUs that do not implement software control of
                    the MAU state.  Some agents may not support
                    setting the value of this object to some of the
                    enumerated values.

                    The value other(1) is returned if the MAU is in a
                    state other than one of the states 2 through 6.

                    The value unknown(2) is returned when the MAU's
                    true state is unknown; for example, when it is
                    being initialized.

                    A MAU in the operational(3) state is fully
                    functional, operates, and passes signals to its
                    attached DTE or repeater port in accordance to its
                    specification.

                    A MAU in standby(4) state forces DI and CI and the
                    media transmitter to idle.  Standby(4) mode only
                    applies to link type MAUs.  The state of
                    rpMauMediaAvailable is unaffected.

                    A MAU in shutdown(5) state assumes the same
                    condition on DI, CI, and the media transmitter as
                    though it were powered down.  The MAU may return
                    other(1) value for the mauJabber and
                    rpMauMediaAvailable objects when it is in this
                    state.  For an AUI, this state will remove power
                    from the AUI.

                    Setting this variable to the value reset(6) resets
                    the MAU in the same manner as a power-off, power-
                    on cycle of at least one-half second would.  The
                    agent is not required to return the value reset
                    (6).
                    Setting this variable to the value operational(3),
                    standby(4), or shutdown(5) causes the MAU to
                    assume the respective state except that setting a
                    mixing-type MAU or an AUI to standby(4) will cause
                    the MAU to enter the shutdown state."
                REFERENCE
                    "Reference IEEE 802.3 MAU Mgt, 20.2.3.2,
                    aMAUAdminState, and 20.2.3.3, acMAUAdminControl
                    and acResetMAUAction."
            ::= { rpMauEntry 5 }

        rpMauMediaAvailable OBJECT-TYPE
            SYNTAX    INTEGER {
                          other(1),
                          unknown(2),
                          available(3),
                          notAvailable(4),
                          remoteFault(5),
                          invalidSignal(6)
                      }
            ACCESS    read-only
            STATUS    mandatory
            DESCRIPTION
                    "If the MAU is a link or fiber type (FOIRL,
                    10BASE-T, 10BASE-F) then this is equivalent to the
                    link test fail state/low light function.  For an
                    AUI or a coax (including broadband) MAU this
                    indicates whether or not loopback is detected on
                    the DI circuit.  The value of this attribute
                    persists between packets for MAU types AUI,
                    10BASE5, 10BASE2, 10BROAD36, and 10BASE-FP.

                    The value other(1) is returned if the
                    mediaAvailable state is not one of 2 through 6.

                    The value unknown(2) is returned when the MAU's
                    true state is unknown; for example, when it is
                    being initialized.  At power-up or following a
                    reset, the value of this attribute will be unknown
                    for AUI, coax, and 10BASE-FP MAUs.  For these MAUs
                    loopback will be tested on each transmission
                    during which no collision is detected.  If DI is
                    receiving input when DO returns to IDL after a
                    transmission and there has been no collision
                    during the transmission then loopback will be
                    detected.  The value of this attribute will only
                    change during non-collided transmissions for AUI,
                    coax, and 10BASE-FP MAUs.
                    The value available(3) indicates that the link,
                    light, or loopback is normal.  The value
                    notAvailable(4) indicates link loss, low light, or
                    no loopback.

                    The value remoteFault(5) indicates that a fault
                    has been detected at the remote end of the link.
                    The value invalidSignal(6) indicates that an
                    invalid signal has been received from the other
                    end of the link.  Both remoteFault(5) and
                    invalidSignal(6) apply only to MAUs of type
                    10BASE-FB."
            REFERENCE
                    "Reference IEEE 802.3 MAU Mgt, 20.2.3.2,
                    aMediaAvailable."
            ::= { rpMauEntry 6 }

        rpMauMediaAvailableStateExits OBJECT-TYPE
            SYNTAX    Counter
            ACCESS    read-only
            STATUS    mandatory
            DESCRIPTION
                    "A count of the number of times that
                    rpMauMediaAvailable for this MAU instance leaves
                    the state available(3)."
            REFERENCE
                    "Reference IEEE 802.3 MAU Mgt, 20.2.3.2,
                    lostMediaCount."
            ::= { rpMauEntry 7 }

        rpMauJabberState OBJECT-TYPE
            SYNTAX    INTEGER {
                          other(1),
                          unknown(2),
                          noJabber(3),
                          jabbering(4)
                      }
            ACCESS    read-only
            STATUS    mandatory
            DESCRIPTION
                    "The value other(1) is returned if the jabber
                    state is not 2, 3, or 4.  The agent must always
                    return other(1) for MAU type dot3MauTypeAUI.

                    The value unknown(2) is returned when the MAU's
                    true state is unknown; for example, when it is
                    being initialized.
                    If the MAU is not jabbering the agent returns
                    noJabber(3).  This is the 'normal' state.

                    If the MAU is in jabber state the agent returns
                    the jabbering(4) value."
            REFERENCE
                    "Reference IEEE 802.3 MAU Mgt, 20.2.3.2,
                    aJabber.jabberFlag."
            ::= { rpMauEntry 8 }

        rpMauJabberingStateEnters OBJECT-TYPE
            SYNTAX    Counter
            ACCESS    read-only
            STATUS    mandatory
            DESCRIPTION
                    "A count of the number of times that
                    rpMauJabberState for this MAU instance enters the
                    state jabbering(4).  For a MAU of type
                    dot3MauTypeAUI, this counter will always indicate
                    zero."
            REFERENCE
                    "Reference IEEE 802.3 MAU Mgt, 20.2.3.2,
                    aJabber.jabberCounter."
            ::= { rpMauEntry 9 }


        --
        --                    The Interface MAU Basic Group
        --
        -- Implementation of the Interface MAU Basic Group is mandatory
        -- for MAUs attached to DTEs (interfaces).

        --
        -- The Basic Interface MAU Table
        --

        ifMauTable OBJECT-TYPE
            SYNTAX    SEQUENCE OF IfMauEntry
            ACCESS    not-accessible
            STATUS    mandatory
            DESCRIPTION
                    "Table of descriptive and status information about
                    the MAU(s) attached to an interface."
            ::= { dot3IfMauBasicGroup 1 }

        ifMauEntry OBJECT-TYPE
            SYNTAX    IfMauEntry
            ACCESS    not-accessible
            STATUS    mandatory
            DESCRIPTION
                    "An entry in the table, containing information
                    about a single MAU."
            INDEX    { ifMauIfIndex, ifMauIndex }
            ::= { ifMauTable 1 }

        IfMauEntry ::=
            SEQUENCE {
                ifMauIfIndex
                    INTEGER,
                ifMauIndex
                    INTEGER,
                ifMauType
                    OBJECT IDENTIFIER,
                ifMauStatus
                    INTEGER,
                ifMauMediaAvailable
                    INTEGER,
                ifMauMediaAvailableStateExits
                    Counter,
                ifMauJabberState
                    INTEGER,
                ifMauJabberingStateEnters
                    Counter
            }

        ifMauIfIndex OBJECT-TYPE
            SYNTAX    INTEGER
            ACCESS    read-only
            STATUS    mandatory
            DESCRIPTION
                    "This variable uniquely identifies the interface
                    to which the MAU described by this entry is
                    connected."
            REFERENCE
                    "Reference RFC 1213, ifIndex."
            ::= { ifMauEntry 1 }

        ifMauIndex OBJECT-TYPE
            SYNTAX    INTEGER (1..9)
            ACCESS    read-only
            STATUS    mandatory
            DESCRIPTION
                    "This variable uniquely identifies the MAU
                    connected to interface ifMauIfIndex that is
                    described by this entry."
            REFERENCE
```

```
                    "Reference IEEE 802.3 MAU Mgt, 20.2.3.2, aMAUID."
            ::= { ifMauEntry 2 }

        ifMauType OBJECT-TYPE
            SYNTAX    OBJECT IDENTIFIER
            ACCESS    read-only
            STATUS    mandatory
            DESCRIPTION
                    "This object identifies the 10 Mb/s baseband or
                    broadband MAU type.  An initial set of MAU types
                    are defined above.  The assignment of OBJECT
                    IDENTIFIERs to new types of MAUs is managed by the
                    IANA.  If the MAU type is unknown, the object
                    identifier

                        unknownMauType OBJECT IDENTIFIER ::= { 0 0 }

                    is returned.  Note that unknownMauType is a
                    syntactically valid object identifier, and any
                    conformant implementation of ASN.1 and the BER
                    must be able to generate and recognize this
                    value."
            REFERENCE
                    "Reference IEEE 802.3 MAU Mgt, 20.2.3.2,
                    aMAUType."
            ::= { ifMauEntry 3 }

        ifMauStatus OBJECT-TYPE
            SYNTAX    INTEGER {
                          other(1),
                          unknown(2),
                          operational(3),
                          standby(4),
                          shutdown(5),
                          reset(6)
                      }
            ACCESS    read-write
            STATUS    mandatory
            DESCRIPTION
                    "The current state of the MAU.  This object may be
                    implemented as a read-only object by those agents
                    and MAUs that do not implement software control of
                    the MAU state.  Some agents may not support
                    setting the value of this object to some of the
                    enumerated values.

                    The value other(1) is returned if the MAU is in a
                    state other than one of the states 2 through 6.
                    The value unknown(2) is returned when the MAU's
                    true state is unknown; for example, when it is
                    being initialized.

                    A MAU in the operational(3) state is fully
                    functional, operates, and passes signals to its
                    attached DTE or repeater port in accordance to its
                    specification.

                    A MAU in standby(4) state forces DI and CI and the
                    media transmitter to idle.  Standby(4) mode only
                    applies to link type MAUs.  The state of
                    ifMauMediaAvailable is unaffected.

                    A MAU in shutdown(5) state assumes the same
                    condition on DI, CI, and the media transmitter as
                    though it were powered down.  The MAU may return
                    other(1) value for the mauJabber and
                    ifMauMediaAvailable objects when it is in this
                    state.  For an AUI, this state will remove power
                    from the AUI.

                    Setting this variable to the value reset(6) resets
                    the MAU in the same manner as a power-off, power-
                    on cycle of at least one-half second would.  The
                    agent is not required to return the value reset
                    (6).

                    Setting this variable to the value operational(3),
                    standby(4), or shutdown(5) causes the MAU to
                    assume the respective state except that setting a
                    mixing-type MAU or an AUI to standby(4) will cause
                    the MAU to enter the shutdown state."
                REFERENCE
                    "Reference IEEE 802.3 MAU Mgt, 20.2.3.2,
                    aMAUAdminState, and 20.2.3.3, acMAUAdminControl
                    and acResetMAUAction."
            ::= { ifMauEntry 4 }

        ifMauMediaAvailable OBJECT-TYPE
            SYNTAX    INTEGER {
                          other(1),
                          unknown(2),
                          available(3),
                          notAvailable(4),
                          remoteFault(5),
                          invalidSignal(6)
                      }
            ACCESS    read-only
            STATUS    mandatory
            DESCRIPTION
                    "If the MAU is a link or fiber type (FOIRL,
                    10BASE-T, 10BASE-F) then this is equivalent to the
                    link test fail state/low light function.  For an
                    AUI or a coax (including broadband) MAU this
                    indicates whether or not loopback is detected on
                    the DI circuit.  The value of this attribute
                    persists between packets for MAU types AUI,
                    10BASE5, 10BASE2, 10BROAD36, and 10BASE-FP.

                    The value other(1) is returned if the
                    mediaAvailable state is not one of 2 through 6.

                    The value unknown(2) is returned when the MAU's
                    true state is unknown; for example, when it is
                    being initialized.  At power-up or following a
                    reset, the value of this attribute will be unknown
                    for AUI, coax, and 10BASE-FP MAUs.  For these MAUs
                    loopback will be tested on each transmission
                    during which no collision is detected.  If DI is
                    receiving input when DO returns to IDL after a
                    transmission and there has been no collision
                    during the transmission then loopback will be
                    detected.  The value of this attribute will only
                    change during non-collided transmissions for AUI,
                    coax, and 10BASE-FP MAUs.

                    The value available(3) indicates that the link,
                    light, or loopback is normal.  The value
                    notAvailable(4) indicates link loss, low light, or
                    no loopback.

                    The value remoteFault(5) indicates that a fault
                    has been detected at the remote end of the link.
                    The value invalidSignal(6) indicates that an
                    invalid signal has been received from the other
                    end of the link.  Both remoteFault(5) and
                    invalidSignal(6) apply only to MAUs of type
                    10BASE-FB."
            REFERENCE
                    "Reference IEEE 802.3 MAU Mgt, 20.2.3.2,
                    aMediaAvailable."
            ::= { ifMauEntry 5 }

        ifMauMediaAvailableStateExits OBJECT-TYPE
            SYNTAX    Counter
            ACCESS    read-only
            STATUS    mandatory
            DESCRIPTION
                    "A count of the number of times that
                    ifMauMediaAvailable for this MAU instance leaves
                    the state available(3)."
            REFERENCE
                    "Reference IEEE 802.3 MAU Mgt, 20.2.3.2,
                    lostMediaCount."
            ::= { ifMauEntry 6 }

        ifMauJabberState OBJECT-TYPE
            SYNTAX    INTEGER {
                          other(1),
                          unknown(2),
                          noJabber(3),
                          jabbering(4)
                      }
            ACCESS    read-only
            STATUS    mandatory
            DESCRIPTION
                    "The value other(1) is returned if the jabber
                    state is not 2, 3, or 4.  The agent must always
                    return other(1) for MAU type dot3MauTypeAUI.

                    The value unknown(2) is returned when the MAU’s
                    true state is unknown; for example, when it is
                    being initialized.

                    If the MAU is not jabbering the agent returns
                    noJabber(3).  This is the ’normal’ state.

                    If the MAU is in jabber state the agent returns
                    the jabbering(4) value."
            REFERENCE
                    "Reference IEEE 802.3 MAU Mgt, 20.2.3.2,
                    aJabber.jabberFlag."
            ::= { ifMauEntry 7 }

        ifMauJabberingStateEnters OBJECT-TYPE
            SYNTAX    Counter
            ACCESS    read-only
            STATUS    mandatory
            DESCRIPTION
                    "A count of the number of times that
                    ifMauJabberState for this MAU instance enters the
                    state jabbering(4).  For a MAU of type
                    dot3MauTypeAUI, this counter will always indicate
```

```
                zero."
            REFERENCE
                "Reference IEEE 802.3 MAU Mgt, 20.2.3.2,
                aJabber.jabberCounter."
            ::= { ifMauEntry 8 }


        --
        --                     The Broadband MAU Basic Group
        --
        -- Implementation of the Broadband MAU Basic Group is mandatory
        -- for broadband MAUs attached to DTEs.

        --
        -- The Basic Broadband MAU Table
        --

        broadMauBasicTable OBJECT-TYPE
            SYNTAX    SEQUENCE OF BroadMauBasicEntry
            ACCESS    not-accessible
            STATUS    mandatory
            DESCRIPTION
                "Table of descriptive and status information about
                the broadband MAUs connected to interfaces."
            ::= { dot3BroadMauBasicGroup 1 }

        broadMauBasicEntry OBJECT-TYPE
            SYNTAX    BroadMauBasicEntry
            ACCESS    not-accessible
            STATUS    mandatory
            DESCRIPTION
                "An entry in the table, containing information
                about a single broadband MAU."
            INDEX     { broadMauIfIndex, broadMauIndex }
            ::= { broadMauBasicTable 1 }

        BroadMauBasicEntry ::=
            SEQUENCE {
                broadMauIfIndex
                    INTEGER,
                broadMauIndex
                    INTEGER,
                broadMauXmtRcvSplitType
                    INTEGER,
                broadMauXmtCarrierFreq
                    INTEGER,
                broadMauTranslationFreq
                    INTEGER
            }

        broadMauIfIndex OBJECT-TYPE
            SYNTAX    INTEGER
            ACCESS    read-only
            STATUS    mandatory
            DESCRIPTION
                    "This variable uniquely identifies the interface
                    to which the MAU described by this entry is
                    connected."
            REFERENCE
                    "Reference RFC 1213, ifIndex."
            ::= { broadMauBasicEntry 1 }

        broadMauIndex OBJECT-TYPE
            SYNTAX    INTEGER (1..9)
            ACCESS    read-only
            STATUS    mandatory
            DESCRIPTION
                    "This variable uniquely identifies the MAU
                    connected to interface broadMauIfIndex that is
                    described by this entry."
            REFERENCE
                    "Reference IEEE 802.3 MAU Mgt, 20.2.3.2, aMAUID."
            ::= { broadMauBasicEntry 2 }

        broadMauXmtRcvSplitType OBJECT-TYPE
            SYNTAX    INTEGER {
                          other(1),
                          single(2),
                          dual(3)
                      }
            ACCESS    read-only
            STATUS    mandatory
            DESCRIPTION
                    "This object indicates the type of frequency
                    multiplexing/cabling system used to separate the
                    transmit and receive paths for the 10BROAD36 MAU.

                    The value other(1) is returned if the split type
                    is not either single or dual.

                    The value single(2) indicates a single cable
                    system.  The value dual(3) indicates a dual cable
                    system, offset normally zero."
            REFERENCE
                    "Reference IEEE 802.3 MAU Mgt, 20.2.3.2,
                    aBbMAUXmitRcvSplitType."
            ::= { broadMauBasicEntry 3 }

        broadMauXmtCarrierFreq OBJECT-TYPE
            SYNTAX    INTEGER
            ACCESS    read-only
            STATUS    mandatory
            DESCRIPTION
                    "This variable indicates the transmit carrier
                    frequency of the 10BROAD36 MAU in MHz/4; that is,
                    in units of 250 kHz."
            REFERENCE
                    "Reference IEEE 802.3 MAU Mgt, 20.2.3.2,
                    aBroadbandFrequencies.xmitCarrierFrequency."
            ::= { broadMauBasicEntry 4 }

        broadMauTranslationFreq OBJECT-TYPE
            SYNTAX    INTEGER
            ACCESS    read-only
            STATUS    mandatory
            DESCRIPTION
                    "This variable indicates the translation offset
                    frequency of the 10BROAD36 MAU in MHz/4; that is,
                    in units of 250 kHz."
            REFERENCE
                    "Reference IEEE 802.3 MAU Mgt, 20.2.3.2,
                    aBroadbandFrequencies.translationFrequency."
            ::= { broadMauBasicEntry 5 }


        -- Traps for use by 802.3 MAUs

        -- Traps are defined using the conventions in RFC 1215 [8].

        rpMauJabberTrap TRAP-TYPE
            ENTERPRISE  snmpDot3MauMgt
            VARIABLES   { rpMauJabberState }
            DESCRIPTION
                    "This trap is sent whenever a managed repeater MAU
                    enters the jabber state.

                     The agent must throttle the generation of
                    consecutive rpMauJabberTraps so that there is at
                    least a five-second gap between them."
            REFERENCE
                    "Reference IEEE 802.3 MAU Mgt, 20.2.3.4,
                    nJabberNotification."
            ::= 1
        ifMauJabberTrap TRAP-TYPE
            ENTERPRISE  snmpDot3MauMgt
            VARIABLES   { ifMauJabberState }
            DESCRIPTION
                    "This trap is sent whenever a managed interface
                    MAU enters the jabber state.

                    The agent must throttle the generation of
                    consecutive ifMauJabberTraps so that there is at
                    least a five-second gap between them."
            REFERENCE
                    "Reference IEEE 802.3 MAU Mgt, 20.2.3.4,
                    nJabberNotification."
            ::= 2

        END
```

5. Acknowledgments

This document is the work of the IETF Hub MIB Working Group. It is based on a proposal written by Geoff Thompson and modified by the IEEE 802.3 Repeater Management Task Force. Paul Woodruff provided valuable corrections and suggestions for improvement.

Members of the IETF Hub MIB Working Group included:

Karl Auerbach	karl@eng.sun.com
Jim Barnes	barnes@xylogics.com
Steve Bostock	steveb@novell.com
David Bridgham	dab@asylum.sf.ca.us
Jack Brown	jbrown@huahuca-emh8.army.mil
Howard Brown	brown@ctron.com
Lida Canin	lida@apple.com
Jeffrey Case	case@cs.utk.edu
Carson Cheung	carson@bnr.com.ca
James Codespote	jpcodes@tycho.ncsc.mil
John Cook	cook@chipcom.com
Dave Cullerot	cullerot@ctron.com
James Davin	jrd@ptt.lcs.mit.edu
Gary Ellis	garye@hpspd.spd.hp.com
David Engel	david@cds.com
Mike Erlinger	mike@mti.com
Jeff Erwin	
Bill Fardy	fardy@ctron.com
Jeff Fried	jmf@relay.proteon.com
Bob Friesenhahn	pdrusa!bob@uunet.uu.net
Shawn Gallagher	gallagher@quiver.enet.dec.com

Mike Grieves	mgrieves@chipcom.com
Walter Guilarte	70026.1715@compuserve.com
Phillip Hasse	phasse@honchuca-emh8.army.mil
Mark Hoerth	mark_hoerth@hp0400.desk.hp.com
Greg Hollingsworth	gregh@mailer.jhuapl.edu
Ron Jacoby	rj@sgi.com
Mike Janson	mjanson@mot.com
Ken Jones	konkord!ksj@uunet.uu.net
Satish Joshi	sjoshi@synoptics.com
Frank Kastenholz	kasten@europa.clearpoint.com
Manu Kaycee	kaycee@trlian.enet.dec.com
Mark Kepke	mak@cnd.hp.com
Mark Kerestes	att!alux2!hawk@uunet.uu.net
Kenneth Key	key@cs.utk.edu
Yoav Kluger	ykluger@fibhaifa.com
Cheryl Krupczak	cheryl@cc.gatech.edu
Ron Lau	rlau@synoptics.com
Chao-Yu Liang	cliang@synoptics.com
Dave Lindemulder	da@mtung.att.com
Richie McBride	rm@bix.co.uk
Keith McCloghrie	kzm@hls.com
Evan McGinnis	bem@3com.com
Donna McMaster	mcmaster@synoptics.com
David Minnich	dwm@fibercom.com
Lynn Monsanto	monsanto@sun.com
Miriam Nihart	miriam@decwet.zso.dec.com
Niels Ole Brunsgaard	nob@dowtyns.dk
Edison Paw	esp@3com.com
David Perkins	dperkins@synoptics.com
Jason Perreault	perreaul@interlan.interlan.com
John Pickens	jrp@3com.com
Jim Reinstedler	jimr@sceng.ub.com
Anil Rijsinghani	anil@levers.enet.dec.com
Sam Roberts	sroberts@farallon.com
Dan Romascanu	dan@lannet.com
Marshall Rose	mrose@dbc.mtview.ca.us
Rick Royston	rick@lsumus.sncc.lsu.edu
Michael Sabo	sabo@dockmaster.ncsc.mil
Jonathan Saperia	saperia@tcpjon.enet.dec.com
Mark Schaefer	schaefer@davidsys.com
Anil Singhal	nsinghal@hawk.ulowell.edu
Timon Sloane	peernet!timon@uunet.uu.net
Bob Stewart	rlstewart@eng.xyplex.com
Emil Sturniolo	emil@dss.com
Bruce Taber	taber@interlan.com
Iris Tal	437-3580@mcimail.com
Mark Therieau	markt@python.eng.microcom.com
Geoff Thompson	thompson@synoptics.com

Dean Throop	throop@dg-rtp.dg.com
Steven Waldbusser	waldbusser@andrew.cmu.edu
Timothy Walden	tmwalden@saturn.sys.acc.com
Philip Wang	watadn!phil@uunet.uu.net
Drew Wansley	dwansley@secola.columbia.ncr.com
David Ward	dward@chipcom.com
Steve Wong	wong@took.enet.dec.com
Paul Woodruff	paul-woodruff@3com.com
Brian Wyld	brianw@spider.co.uk
June-Kang Yang	natadm!yang@uunet.uu.net
Henry Yip	natadm!henry@uunet.uu.net
John Ziegler	ziegler@artel.com
Joseph Zur	zur@fibhaifa.com

6. References

[1] Rose, M., and K. McCloghrie, "Structure and Identification of Management Information for TCP/IP-based internets", STD 16, RFC 1155, Performance Systems International, Hughes LAN Systems, May 1990.

[2] McCloghrie, K., and M. Rose, "Management Information Base for Network Management of TCP/IP-based internets", RFC 1156, Hughes LAN Systems, Performance Systems International, May 1990.

[3] Case, J., Fedor M., Schoffstall, M., and J. Davin, "Simple Network Management Protocol", STD 15, RFC 1157, SNMP Research, Performance Systems International, Performance Systems International, MIT Laboratory for Computer Science, May 1990.

[4] McCloghrie, K., and M. Rose, Editors, "Management Information Base for Network Management of TCP/IP-based internets: MIB-II", STD 17, RFC 1213, Hughes LAN Systems, Performance Systems International, March 1991.

[5] Information processing systems - Open Systems Interconnection - Specification of Abstract Syntax Notation One (ASN.1), International Organization for Standardization, International Standard 8824, December 1987.

[6] Information processing systems - Open Systems Interconnection - Specification of Basic Encoding Rules for Abstract Notation One (ASN.1), International Organization for Standardization, International Standard 8825, December 1987.

[7] Rose, M., and K. McCloghrie, Editors, "Concise MIB Definitions", STD 16, RFC 1212, Performance Systems International, Hughes LAN Systems, March 1991.

[8] Rose, M., Editor, "A Convention for Defining Traps for use with the SNMP", RFC 1215, Performance Systems International, March 1991.

[9] IEEE 802.3/ISO 8802-3 Information processing systems - Local area networks - Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications, 2nd edition, September 21, 1990.

[10] IEEE P802.3p, "Layer Management for 10 Mb/s Medium Access Unit (MAUs), Section 20", Draft Supplement to ANSI/IEEE 802.3, Draft 5, July 11, 1992.

[11] Kastenholz, F., "Definitions of Managed Objects for the Ethernet-like Interface Types", RFC 1398, FTP Software, Inc., January 1993.

[12] McMaster, D., and K. McCloghrie, Editors, "Definitions of Managed Objects for IEEE 802.3 Repeater Devices", RFC 1368, SynOptics Communications, Hughes LAN Systems, October 1992.

7. Security Considerations

Security issues are not discussed in this memo.

8. Authors' Addresses

Donna McMaster
SynOptics Communications, Inc.
4401 Great America Parkway
P.O. Box 58185
Santa Clara, CA 95052-8185

Phone: (408) 764-1206
EMail: mcmaster@synoptics.com

Keith McCloghrie
Hughes LAN Systems, Inc.
1225 Charleston Road
Mountain View, CA 94043

Phone: (415) 966-7934
EMail: kzm@hls.com

Sam Roberts
Farallon Computing, Inc.
2470 Mariner Square Loop
Alameda, CA 94501-1010

Phone: (510) 814-5215
EMail: sroberts@farallon.com