Hub MIB Working Group J. Flick
INTERNET DRAFT Hewlett-Packard Company
J. Johnson
RedBack Networks
F. Kastenholz
FTP Software, Inc.
November 1997

Definitions of Managed Objects for the Ethernet-like Interface Types

<draft-ietf-hubmib-etherif-mib-02.txt>

Status of this Memo

Copyright Notice

Abstract

This memo is an extension to the SNMP MIB. It specifies an IAB standards track protocol for the Internet community, and requests discussion and suggestions for improvements. The origin of this memo is from RFC 1650 ''Definitions of Managed Objects for the Ethernet-like Interface Types using SMIv2.'' This memo extends that specification by including management information useful for the management of 100BaseT ethernet interfaces.

Distribution of this memo is unlimited. Please forward comments to hubmib@hprnd.rose.hp.com.

1. Introduction

This memo defines a portion of the Management Information Base (MIB) for use with network management protocols in the Internet community. In particular, it defines objects for managing ethernet-like interfaces.

This memo also includes a MIB module. This MIB module extends the list of managed objects specified in the earlier version of this MIB: RFC1650 [11].

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 [13].

2. The SNMP Network Management Framework

The SNMP Network Management Framework consists of several components. For the purpose of this specification, the applicable components of the Framework are the SMI and related documents [2, 3, 4], which define the mechanisms used for describing and naming objects for the purpose of management.

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

2.1. 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) [1] defined in the SMI [2]. 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. Change Log

This section enumerates changes made to RFC 1650 to produce this document.

(1) The MODULE-IDENTITY has been updated to reflect the changes in the MIB.

(2) A new object, dot3StatsSymbolErrors, has been added.

(3) The definition of the object dot3StatsIndex has been converted to use the SMIv2 OBJECT-TYPE macro.

(4) A new conformance group, etherStats100MbsGroup, has been added.

(5) A new compliance statement, ether100MbsCompliance, has been added.

(6) The Acknowledgements were extended to provide a more complete history of the origin of this document.

(7) The discussion of ifType has been expanded.

(8) A section on mapping of Interfaces MIB objects has been added.

(9) A section defining the relationship of this MIB to the MAU MIB has been added.

(10) A section on the mapping of IEEE 802.3 managed objects to this MIB and the Interfaces MIB has been added.

(11) Comverted the dot3Tests, dot3Errors, and dot3ChipSets OIDs to use the OBJECT-IDENTITY macro.

(12) An intellectual property notice and copyright notice were added, as required by RFC 2026.

4. Overview

Instances of these object types represent attributes of an interface to an ethernet-like communications medium. At present, ethernet-like media are identified by the following values of the ifType object in the Interfaces MIB [12]:

```
ethernetCsmacd(6)
iso88023Csmacd(7)
starLan(11)
```

The definitions presented here are based on the IEEE 802.3 Layer Management Specification [5], as originally interpreted by Frank Kastenholz then of Interlan in [7]. Implementors of these MIB objects should note that the IEEE document explicitly describes (in the form of Pascal pseudocode) when, where, and how various MAC

attributes are measured. The IEEE document also describes the effects of MAC actions that may be invoked by manipulating instances of the MIB objects defined here.

To the extent that some of the attributes defined in [5] are represented by previously defined objects in the Internet-standard MIB or in the Interfaces Group Evolution MIB [12], such attributes are not redundantly represented by objects defined in this memo. Among the attributes represented by objects defined in other memos are the number of octets transmitted or received on a particular interface, the number of frames transmitted or received on a particular interface, the promiscuous status of an interface, the MAC address of an interface, and multicast information associated with an interface.

4.1. Relation to MIB-2

This section applies only when this MIB is used in conjunction with the "old" (RFC 1213) interface group.

The relationship between an ethernet-like interface and an interface in the context of the Internet-standard MIB is one-to-one. As such, the value of an ifIndex object instance can be directly used to identify corresponding instances of the objects defined herein.

For agents which implement the (now deprecated) ifSpecific object, an instance of that object that is associated with an ethernet-like interface has the OBJECT IDENTIFIER value:

```
      dot3    OBJECT IDENTIFER ::= { transmission 7 }
```

4.2. Relation to the Interfaces MIB

The Interface MIB [12] requires that any MIB which is an adjunct of the Interface MIB clarify specific areas within the Interface MIB. These areas were intentionally left vague in the Interface MIB to avoid over constraining the MIB, thereby precluding management of certain media-types.

Section 3.3 of [12] enumerates several areas which a media-specific MIB must clarify. Each of these areas is addressed in a following subsection. The implementor is referred to [12] in order to understand the general intent of these areas.

4.2.1. Layering Model

This MIB does not provide for layering. There are no sublayers.

EDITOR'S NOTE:

One could foresee the development of an 802.2 and enet-transceiver MIB. They could be higher and lower sublayers, respectively. All that THIS document should do is allude to the possibilities and urge the implementor to be aware of the possibility and that they may have requirements which supersede the requirements in this document.

4.2.2. Virtual Circuits

This medium does not support virtual circuits and this area is not applicable to this MIB.

4.2.3. ifTestTable

This MIB defines two tests for media which are instrumented with this MIB; TDR and Loopback. Implementation of these tests is not required. Many common interface chips do not support one or both of these tests.

These two tests are provided as a convenience, allowing a common method to invoke the test.

Standard MIBs do not include objects in which to return the results of the TDR test. Any needed objects MUST be provided in the vendor specific MIB.

Note that the ifTestTable is now deprecated. Work is underway to define a replacement MIB for system and interface testing. It is expected that the tests defined in this document will be usable in this replacement MIB.

4.2.4. ifRcvAddressTable

This table contains all IEEE 802.3 addresses, unicast, multicast, and broadcast, for which this interface will receive packets and forward them up to a higher layer entity for local consumption. The format of the address, contained in ifRcvAddressAddress, is the same as for ifPhysAddress.

In the event that the interface is part of a MAC bridge, this table

does not include unicast addresses which are accepted for possible forwarding out some other port. This table is explicitly not intended to provide a bridge address filtering mechanism.

4.2.5. ifPhysAddress

This object contains the IEEE 802.3 address which is placed in the source-address field of any Ethernet, Starlan, or IEEE 802.3 frames that originate at this interface. Usually this will be kept in ROM on the interface hardware. Some systems may set this address via software.

In a system where there are several such addresses the designer has a tougher choice. The address chosen should be the one most likely to be of use to network management (e.g. the address placed in ARP responses for systems which are primarily IP systems).

If the designer truly can not chose, use of the factory- provided ROM address is suggested.

If the address can not be determined, an octet string of zero length should be returned.

The address is stored in binary in this object. The address is stored in "canonical" bit order, that is, the Group Bit is positioned as the low-order bit of the first octet. Thus, the first byte of a multicast address would have the bit 0x01 set.

4.2.6. ifType

This MIB applies to interfaces which have any of the following ifType values:

```
ethernetCsmacd(6)
iso88023Csmacd(7)
starLan(11)
```

It is RECOMMENDED that all Ethernet-like interfaces use an ifType of ethernetCsmacd(6) regardless of the speed that the interface is running or the link-layer encapsulation in use. iso88023Csmacd(7) and starLan(11) are supported for backwards compatability.

There are two other interface types defined in the IANAifType-MIB for 100 Mbit Ethernet. They are fastEther(62), and fastEtherFX(69). This document takes the position that an Ethernet is an Ethernet, and Ethernet interfaces SHOULD always have the same value of ifType.

Information on the particular flavor of Ethernet that an interface is running is available from ifSpeed in the Interfaces MIB, and ifMauType in the 802.3 MAU MIB. An Ethernet-like interface SHOULD NOT use the fastEther(62) or fastEtherFX(69) ifTypes.

Interfaces with any of the supported ifType values map to the EtherLike-MIB in the same manner. Which compliance statement an interface should implement is dependent on the maximum speed supported on the interface. The EtherLike-MIB etherCompliance compliance statement applies to all Ethernet-like interfaces whose maximum supported speed is 10 Mbit/sec or less. There are no implementation differences. Similarly, the EtherLike-MIB ether100MbsCompliance compliance statement applies to all Ethernet-like interfaces whose maximum supported speed of 100Mbit/sec.

An interface that is capable of operating at 100Mbit/sec MUST implement the ether100MbsCompliance compliance statement, even if it is currently operating at a lower speed. Counters in the ether100MbsCompliance compliance statement that only apply to 100 Mbit interfaces would simply not increment when the interface is operating at a lower speed.

4.2.7. Specific Interface MIB Objects

The following table provides specific implementation guidelines for applying the interface group objects to ethernet-like media.

Object	
ifIndex	Each ethernet-like interface is represented by an ifEntry. The dot3StatsTable in this MIB module is indexed by dot3StatsIndex. The interface identified by a particular value of dot3StatsIndex is the same interface as identified by the same value of ifIndex.
ifDescr	Refer to [12].
ifType	Refer to section 4.2.6.
ifMtu	1500 octets.
ifSpeed	The current operational speed of the interface in bits per second. For current ethernet-like interfaces, this will be equal to 1,000,000 (1 million),

10,000,000 (10 million), or 100,000,000 (100 million). If the interface implements auto-negotiation, auto-negotiation is enabled for this interface, and the interface has not yet negotiated to an operational speed, this object SHOULD reflect the maximum speed supported by the interface. Note that this object MUST NOT indicate a doubled value when operating in full-duplex mode. It MUST indicate the correct line speed regardless of the current duplex mode. The correct object to use to determine the duplex mode of the interface is the ifMauType object in the 802.3 MAU MIB.

ifPhysAddress
Refer to section 4.2.5.

ifAdminStatus
Write access is not required. Support for 'testing' is not required.

ifOperStatus
The operational state of the interface. Support for 'testing' is not required. The value 'dormant' has no meaning for an ethernet-like interface.

ifLastChange
Refer to [12].

ifInOctets
The number of octets in valid MAC frames received on this interface, including the MAC header and FCS.

ifInUcastPkts
Refer to [12].

ifInDiscards
Refer to [12].

ifInErrors
The sum for this interface of dot3StatsAlignmentErrors, dot3StatsFCSErrors, dot3StatsFrameTooLongs, dot3StatsInternalMacReceiveErrors and dot3StatsSymbolErrors.

ifInUnknownProtos
Refer to [12].

ifOutOctets
The number of octets transmitted in valid MAC frames on this interface,

including the MAC header and FCS.

ifOutUcastPkts
Refer to [12].

ifOutDiscards
Refer to [12].

ifOutErrors
The sum for this interface of: dot3StatsSQETestErrors, dot3StatsLateCollisions, dot3StatsExcessiveCollisions, dot3StatsInternalMacTransmitErrors and dot3StatsCarrierSenseErrors.

ifName
Locally-significant textual name for the interface (e.g. lan0).

ifInMulticastPkts
Refer to [12].

ifInBroadcastPkts
Refer to [12].

ifOutMulticastPkts
Refer to [12].

ifOutBroadcastPkts
Refer to [12].

ifHCInOctets
ifHCOutOctets
64-bit versions of counters. Required for ethernet-like interfaces that are capable of operating at 20Mbit/sec or faster, even if the interface is currently operating at less than 20Mbit/sec.

ifHCInUcastPkts
ifHCInMulticastPkts
ifHCInBroadcastPkts
ifHCOutUcastPkts
ifHCOutMulticastPkts
ifHCOutBroadcastPkts
64-bit versions of packet counters. Support for these counters is not required for the interface types supported by this MIB. They are only required for interfaces capable of operating at 640Mbit/sec or faster. Note that a future revision of this document may support faster interfaces, and therefore may require support for these counters.

ifLinkUpDownTrapEnable
Refer to [12]. Default is 'enabled'

ifHighSpeed
The current operational speed of the interface in millions of bits per second. For current ethernet-like interfaces, this will be equal to 1, 10,

	or 100. If the interface implements auto-negotiation, auto-negotiation is enabled for this interface, and the interface has not yet negotiated to an operational speed, this object SHOULD reflect the maximum speed supported by the interface. Note that this object MUST NOT indicate a doubled value when operating in full-duplex mode. It MUST indicate the correct line speed regardless of the current duplex mode. The correct object to use to determine the duplex mode of the interface is the ifMauType object in the 802.3 MAU MIB.
ifPromiscuousMode	Refer to [12].
ifConnectorPresent	This will normally be 'true'.
ifAlias	Refer to [12].
ifCounterDiscontinuityTime	Refer to [12].
ifStackHigherLayer ifStackLowerLayer ifStackStatus	Refer to section 4.2.1.
ifRcvAddressAddress ifRcvAddressStatus ifRcvAddressType	Refer to section 4.2.4.

4.3. Relation to the 802.3 MAU MIB

Support for the mauModIfCompl compliance statement of the MAU-MIB [14] is REQUIRED for Ethernet-like interfaces. This MIB is needed in order to allow applications to determine the current MAU type in use by the interface. The MAU type indicates not only the media type in use, but also indicates whether the interface is operating in half-duplex or full-duplex mode. Implementing this MIB module without implementing the MAU-MIB would leave applications with no standard way to determine the duplex mode of the interface.

4.4. Mapping of IEEE 802.3 Managed Objects

IEEE 802.3 Managed Object	Corresponding SNMP Object

oMacEntity	
.aMACID	dot3StatsIndex or IF-MIB - ifIndex
.aFramesTransmittedOK	IF-MIB - ifOutUCastPkts + ifOutMulticastPkts + ifOutBroadcastPkts
.aSingleCollisionFrames	dot3StatsSingleCollisionFrames
.aMultipleCollisionFrames	dot3StatsMultipleCollisionFrames
.aFramesReceivedOK	IF-MIB - ifInUcastPkts + ifInMulticastPkts + ifInBroadcastPkts
.aFrameCheckSequenceErrors	dot3StatsFCSErrors
.aAlignmentErrors	dot3StatsAlignmentErrors
.aOctetsTransmittedOK	IF-MIB - ifOutOctets
.aFramesWithDeferredXmissions	dot3StatsDeferredTransmissions
.aLateCollisions	dot3StatsLateCollisions
.aFramesAbortedDueToXSColls	dot3StatsExcessiveCollisions
.aFramesLostDueToIntMACXmitError	dot3StatsInternalMacTransmitErrors
.aCarrierSenseErrors	dot3StatsCarrierSenseErrors
.aOctetsReceivedOK	IF-MIB - ifInOctets
.aFramesLostDueToIntMACRcvError	dot3StatsInternalMacReceiveErrors
.aPromiscuousStatus	IF-MIB - ifPromiscuousMode
.aReadMulticastAddressList	IF-MIB - ifRcvAddressTable
.aMulticastFramesXmittedOK	IF-MIB - ifOutMulticastPkts
.aBroadcastFramesXmittedOK	IF-MIB - ifOutBroadcastPkts
.aMulticastFramesReceivedOK	IF-MIB - ifInMulticastPkts
.aBroadcastFramesReceivedOK	IF-MIB - ifInBroadcastPkts
.aFrameTooLongErrors	dot3StatsFrameTooLongs
.aReadWriteMACAddress	IF-MIB - ifPhysAddress
.aCollisionFrames	dot3CollFrequencies
.acAddGroupAddress	IF-MIB - ifRcvAddressTable
.acDeleteGroupAddress	IF-MIB - ifRcvAddressTable
.acExecuteSelfTest	dot3TestLoopBack
oPHYEntity	
.aSQETestErrors	dot3StatsSQETestErrors
.aSymbolErrorDuringCarrier	dot3StatsSymbolErrors

The following IEEE 802.3 managed objects have been removed from this MIB module as a result of implementation feedback:

oMacEntity
  .aFramesWithExcessiveDeferral
  .aInRangeLengthErrors
  .aOutOfRangeLengthField
  .aMACEnableStatus
  .aTransmitEnableStatus

```
     .aMulticastReceiveStatus
     .acInitializeMAC
```

Please see [15] for the detailed reasoning on why these objects were removed.

5. Definitions

```
   EtherLike-MIB DEFINITIONS ::= BEGIN

       IMPORTS
           MODULE-IDENTITY, OBJECT-TYPE, OBJECT-IDENTITY,
           Counter32, mib-2, transmission
               FROM SNMPv2-SMI
           MODULE-COMPLIANCE, OBJECT-GROUP
               FROM SNMPv2-CONF
           ifIndex, InterfaceIndex
               FROM IF-MIB;

       etherMIB MODULE-IDENTITY
           LAST-UPDATED "9711102157Z"  -- November 10, 1997
           ORGANIZATION "IETF 802.3 Hub MIB Working Group"
           CONTACT-INFO
               "WG E-mail: hubmib@hprnd.rose.hp.com

                    Chair: Dan Romascanu
                   Postal: Madge Networks
                           Atidum Technology Park, Bldg. 3
                           Tel Aviv 61131
                           Israel
                      Tel: +972 3 645 8414
                   E-mail: dromasca@madge.com

                  Editor: John Flick
                  Postal: Hewlett-Packard Company
                          8000 Foothills Blvd. M/S 5556
                          Roseville, CA 95747-5556
                          USA
                     Tel: +1 916 785 4018
                     Fax: +1 916 785 3583
                  E-mail: johnf@hprnd.rose.hp.com

                  Editor: Jeffrey Johnson
                  Postal: RedBack Networks
                          2570 North First Street, Suite 410
                          San Jose, CA, 95131
                          USA
                     Tel: +1 408 571 2699
                     Fax: +1 408 571 2698
                  E-Mail: jeff@redbacknetworks.com"

           DESCRIPTION "The MIB module to describe generic objects for
                       Ethernet-like network interfaces. This MIB is an
                       updated version of the Ethernet-like MIB in RFC
```

```
                    1650."

        REVISION    "9711102157Z"
        DESCRIPTION "Updated to include support for 100 Mb/sec
                    interfaces."

        REVISION    "9402030400Z"
        DESCRIPTION "Version published as RFC 1650."
        ::= { mib-2 35 }

    etherMIBObjects OBJECT IDENTIFIER ::= { etherMIB 1 }

    dot3    OBJECT IDENTIFIER ::= { transmission 7 }

    -- the Ethernet-like Statistics group

    dot3StatsTable OBJECT-TYPE
        SYNTAX     SEQUENCE OF Dot3StatsEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION "Statistics for a collection of ethernet-like
                    interfaces attached to a particular system."
        ::= { dot3 2 }

    dot3StatsEntry OBJECT-TYPE
        SYNTAX     Dot3StatsEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION "Statistics for a particular interface to an
                    ethernet-like medium."
        INDEX       { dot3StatsIndex }
        ::= { dot3StatsTable 1 }

    Dot3StatsEntry ::=
        SEQUENCE {
            dot3StatsIndex                      InterfaceIndex,
            dot3StatsAlignmentErrors            Counter32,
            dot3StatsFCSErrors                  Counter32,
            dot3StatsSingleCollisionFrames      Counter32,
            dot3StatsMultipleCollisionFrames    Counter32,
            dot3StatsSQETestErrors              Counter32,
            dot3StatsDeferredTransmissions      Counter32,
            dot3StatsLateCollisions             Counter32,
            dot3StatsExcessiveCollisions        Counter32,
            dot3StatsInternalMacTransmitErrors  Counter32,
            dot3StatsCarrierSenseErrors         Counter32,
            dot3StatsFrameTooLongs              Counter32,
            dot3StatsInternalMacReceiveErrors   Counter32,
```

```
               dot3StatsEtherChipSet              OBJECT IDENTIFIER,
               dot3StatsSymbolErrors              Counter32
           }

       dot3StatsIndex OBJECT-TYPE
           SYNTAX      InterfaceIndex
           MAX-ACCESS  read-only
           STATUS      current
           DESCRIPTION "An index value that uniquely identifies an
                       interface to an ethernet-like medium.  The
                       interface identified by a particular value of
                       this index is the same interface as identified
                       by the same value of ifIndex."
           ::= { dot3StatsEntry 1 }

       dot3StatsAlignmentErrors OBJECT-TYPE
           SYNTAX      Counter32
           MAX-ACCESS  read-only
           STATUS      current
           DESCRIPTION "A count of frames received on a particular
                       interface that are not an integral number of
                       octets in length and do not pass the FCS check.

                       The count represented by an instance of this
                       object is incremented when the alignmentError
                       status is returned by the MAC service to the
                       LLC (or other MAC user). Received frames for
                       which multiple error conditions obtain are,
                       according to the conventions of IEEE 802.3
                       Layer Management, counted exclusively according
                       to the error status presented to the LLC."
           REFERENCE   "IEEE 802.3 Layer Management"
           ::= { dot3StatsEntry 2 }

       dot3StatsFCSErrors OBJECT-TYPE
           SYNTAX      Counter32
           MAX-ACCESS  read-only
           STATUS      current
           DESCRIPTION "A count of frames received on a particular
                       interface that are an integral number of octets
                       in length but do not pass the FCS check.

                       The count represented by an instance of this
                       object is incremented when the frameCheckError
                       status is returned by the MAC service to the
                       LLC (or other MAC user). Received frames for
                       which multiple error conditions obtain are,
                       according to the conventions of IEEE 802.3
```

```
                       Layer Management, counted exclusively according
                       to the error status presented to the LLC."
           REFERENCE   "IEEE 802.3 Layer Management"
           ::= { dot3StatsEntry 3 }

       dot3StatsSingleCollisionFrames OBJECT-TYPE
           SYNTAX      Counter32
           MAX-ACCESS  read-only
           STATUS      current
           DESCRIPTION "A count of successfully transmitted frames on
                       a particular interface for which transmission
                       is inhibited by exactly one collision.

                       A frame that is counted by an instance of this
                       object is also counted by the corresponding
                       instance of either the ifOutUcastPkts,
                       ifOutMulticastPkts, or ifOutBroadcastPkts,
                       and is not counted by the corresponding
                       instance of the dot3StatsMultipleCollisionFrames
                       object."
           REFERENCE   "IEEE 802.3 Layer Management"
           ::= { dot3StatsEntry 4 }


       dot3StatsMultipleCollisionFrames OBJECT-TYPE
           SYNTAX      Counter32
           MAX-ACCESS  read-only
           STATUS      current
           DESCRIPTION "A count of successfully transmitted frames on
                       a particular interface for which transmission
                       is inhibited by more than one collision.

                       A frame that is counted by an instance of this
                       object is also counted by the corresponding
                       instance of either the ifOutUcastPkts,
                       ifOutMulticastPkts, or ifOutBroadcastPkts,
                       and is not counted by the corresponding
                       instance of the dot3StatsSingleCollisionFrames
                       object."
           REFERENCE   "IEEE 802.3 Layer Management"
           ::= { dot3StatsEntry 5 }


       dot3StatsSQETestErrors OBJECT-TYPE
           SYNTAX      Counter32
           MAX-ACCESS  read-only
           STATUS      current
           DESCRIPTION "A count of times that the SQE TEST ERROR
```

```
                       message is generated by the PLS sublayer for a
                       particular interface. The SQE TEST ERROR
                       message is defined in section 7.2.2.2.4 of
                       ANSI/IEEE 802.3-1985 and its generation is
                       described in section 7.2.4.6 of the same
                       document."
           REFERENCE   "ANSI/IEEE Std 802.3-1985 Carrier Sense
                       Multiple Access with Collision Detection Access
                       Method and Physical Layer Specifications"
           ::= { dot3StatsEntry 6 }

       dot3StatsDeferredTransmissions OBJECT-TYPE
           SYNTAX      Counter32
           MAX-ACCESS  read-only
           STATUS      current
           DESCRIPTION "A count of frames for which the first
                       transmission attempt on a particular interface
                       is delayed because the medium is busy.

                       The count represented by an instance of this
                       object does not include frames involved in
                       collisions."
           REFERENCE   "IEEE 802.3 Layer Management"
           ::= { dot3StatsEntry 7 }

       dot3StatsLateCollisions OBJECT-TYPE
           SYNTAX      Counter32
           MAX-ACCESS  read-only
           STATUS      current
           DESCRIPTION "The number of times that a collision is
                       detected on a particular interface later than
                       512 bit-times into the transmission of a
                       packet.

                       Five hundred and twelve bit-times corresponds
                       to 51.2 microseconds on a 10 Mbit/s system. A
                       (late) collision included in a count
                       represented by an instance of this object is
                       also considered as a (generic) collision for
                       purposes of other collision-related
                       statistics."
           REFERENCE   "IEEE 802.3 Layer Management"
           ::= { dot3StatsEntry 8 }

       dot3StatsExcessiveCollisions OBJECT-TYPE
           SYNTAX      Counter32
           MAX-ACCESS  read-only
           STATUS      current
```

```
           DESCRIPTION "A count of frames for which transmission on a
                       particular interface fails due to excessive
                       collisions."
           REFERENCE   "IEEE 802.3 Layer Management"
           ::= { dot3StatsEntry 9 }

       dot3StatsInternalMacTransmitErrors OBJECT-TYPE
           SYNTAX      Counter32
           MAX-ACCESS  read-only
           STATUS      current
           DESCRIPTION "A count of frames for which transmission on a
                       particular interface fails due to an internal
                       MAC sublayer transmit error. A frame is only
                       counted by an instance of this object if it is
                       not counted by the corresponding instance of
                       either the dot3StatsLateCollisions object, the
                       dot3StatsExcessiveCollisions object, or the
                       dot3StatsCarrierSenseErrors object.

                       The precise meaning of the count represented by
                       an instance of this object is implementation-
                       specific.  In particular, an instance of this
                       object may represent a count of transmission
                       errors on a particular interface that are not
                       otherwise counted."
           REFERENCE   "IEEE 802.3 Layer Management"
           ::= { dot3StatsEntry 10 }

       dot3StatsCarrierSenseErrors OBJECT-TYPE
           SYNTAX      Counter32
           MAX-ACCESS  read-only
           STATUS      current
           DESCRIPTION "The number of times that the carrier sense
                       condition was lost or never asserted when
                       attempting to transmit a frame on a particular
                       interface.

                       The count represented by an instance of this
                       object is incremented at most once per
                       transmission attempt, even if the carrier sense
                       condition fluctuates during a transmission
                       attempt."
           REFERENCE   "IEEE 802.3 Layer Management"
           ::= { dot3StatsEntry 11 }

       -- { dot3StatsEntry 12 } is not assigned

       dot3StatsFrameTooLongs OBJECT-TYPE
```

```
            SYNTAX      Counter32
            MAX-ACCESS  read-only
            STATUS      current
            DESCRIPTION "A count of frames received on a particular
                        interface that exceed the maximum permitted
                        frame size.

                        The count represented by an instance of this
                        object is incremented when the frameTooLong
                        status is returned by the MAC service to the
                        LLC (or other MAC user). Received frames for
                        which multiple error conditions obtain are,
                        according to the conventions of IEEE 802.3
                        Layer Management, counted exclusively according
                        to the error status presented to the LLC."
            REFERENCE   "IEEE 802.3 Layer Management"
            ::= { dot3StatsEntry 13 }

        -- { dot3StatsEntry 14 } is not assigned

        -- { dot3StatsEntry 15 } is not assigned

        dot3StatsInternalMacReceiveErrors OBJECT-TYPE
            SYNTAX      Counter32
            MAX-ACCESS  read-only
            STATUS      current
            DESCRIPTION "A count of frames for which reception on a
                        particular interface fails due to an internal
                        MAC sublayer receive error. A frame is only
                        counted by an instance of this object if it is
                        not counted by the corresponding instance of
                        either the dot3StatsFrameTooLongs object, the
                        dot3StatsAlignmentErrors object, or the
                        dot3StatsFCSErrors object.
                        The precise meaning of the count represented by
                        an instance of this object is implementation-
                        specific.  In particular, an instance of this
                        object may represent a count of receive errors
                        on a particular interface that are not
                        otherwise counted."
            REFERENCE   "IEEE 802.3 Layer Management"
            ::= { dot3StatsEntry 16 }

        dot3StatsEtherChipSet OBJECT-TYPE
            SYNTAX      OBJECT IDENTIFIER
            MAX-ACCESS  read-only
            STATUS      current
            DESCRIPTION "This object contains an OBJECT IDENTIFIER
```

```
                       which identifies the chipset used to
                       realize the interface. Ethernet-like
                       interfaces are typically built out of
                       several different chips. The MIB implementor
                       is presented with a decision of which chip
                       to identify via this object. The implementor
                       should identify the chip which is usually
                       called the Medium Access Control chip.
                       If no such chip is easily identifiable,
                       the implementor should identify the chip
                       which actually gathers the transmit
                       and receive statistics and error
                       indications. This would allow a
                       manager station to correlate the
                       statistics and the chip generating
                       them, giving it the ability to take
                       into account any known anomalies
                       in the chip."
           ::= { dot3StatsEntry 17 }

       dot3StatsSymbolErrors OBJECT-TYPE
           SYNTAX      Counter32
           MAX-ACCESS  read-only
           STATUS      current
           DESCRIPTION "The number of times there was an invalid data
                       symbol when a valid carrier was present on a
                       particular interface.

                       The count represented by an instance of this
                       object is incremented at most once per carrier
                       event, even if multiple symbol errors occur
                       during the carrier event."
           REFERENCE   "IEEE 802.3u-1995 10 & 100 Mb/s Management"
           ::= { dot3StatsEntry 18 }

       -- the Ethernet-like Collision Statistics group

       -- Implementation of this group is optional; it is appropriate
       -- for all systems which have the necessary metering

       dot3CollTable OBJECT-TYPE
           SYNTAX      SEQUENCE OF Dot3CollEntry
           MAX-ACCESS  not-accessible
           STATUS      current
           DESCRIPTION "A collection of collision histograms for a
                       particular set of interfaces."
           ::= { dot3 5 }
```

```
       dot3CollEntry OBJECT-TYPE
           SYNTAX      Dot3CollEntry
           MAX-ACCESS  not-accessible
           STATUS      current
           DESCRIPTION "A cell in the histogram of per-frame
                       collisions for a particular interface.  An
                       instance of this object represents the
                       frequency of individual MAC frames for which
                       the transmission (successful or otherwise) on a
                       particular interface is accompanied by a
                       particular number of media collisions."
           INDEX       { ifIndex, dot3CollCount }
           ::= { dot3CollTable 1 }

       Dot3CollEntry ::=
           SEQUENCE {
               dot3CollCount        INTEGER,
               dot3CollFrequencies  Counter32
           }

       -- { dot3CollEntry 1 } is no longer in use

       dot3CollCount OBJECT-TYPE
           SYNTAX      INTEGER (1..16)
           MAX-ACCESS  not-accessible
           STATUS      current
           DESCRIPTION "The number of per-frame media collisions for
                       which a particular collision histogram cell
                       represents the frequency on a particular
                       interface."
           ::= { dot3CollEntry 2 }


       dot3CollFrequencies OBJECT-TYPE
           SYNTAX      Counter32
           MAX-ACCESS  read-only
           STATUS      current
           DESCRIPTION "A count of individual MAC frames for which the
                       transmission (successful or otherwise) on a
                       particular interface occurs after the
                       frame has experienced exactly the number
                       of collisions in the associated
                       dot3CollCount object.

                       For example, a frame which is transmitted
                       on interface 77 after experiencing
                       exactly 4 collisions would be indicated
                       by incrementing only dot3CollFrequencies.77.4.
```

```
                       No other instance of dot3CollFrequencies would
                       be incremented in this example."
           ::= { dot3CollEntry 3 }

       --  802.3 Tests

       dot3Tests   OBJECT IDENTIFIER ::= { dot3 6 }

       dot3Errors  OBJECT IDENTIFIER ::= { dot3 7 }


       --  TDR Test

       dot3TestTdr OBJECT-IDENTITY
           STATUS      current
           DESCRIPTION "The Time-Domain Reflectometry (TDR) test is
                       specific to ethernet-like interfaces of type
                       10Base5 and 10Base2.  The TDR value may be
                       useful in determining the approximate distance
                       to a cable fault.  It is advisable to repeat
                       this test to check for a consistent resulting
                       TDR value, to verify that there is a fault.

                       A TDR test returns as its result the time
                       interval, measured in 10 MHz ticks or 100 nsec
                       units, between the start of TDR test
                       transmission and the subsequent detection of a
                       collision or deassertion of carrier.  On
                       successful completion of a TDR test, the result
                       is stored as the value of an appropriate
                       instance of an appropriate vendor specific MIB
                       object, and the OBJECT IDENTIFIER of that
                       instance is stored in the appropriate instance
                       of the appropriate test result code object
                       (thereby indicating where the result has been
                       stored).
           ::= { dot3Tests 1 }

       -- Loopback Test

       dot3TestLoopBack OBJECT-IDENTITY
           STATUS      current
           DESCRIPTION "This test configures the MAC chip and executes
                       an internal loopback test of memory, data paths,
                       and the MAC chip logic.  This loopback test can
                       only be executed if the interface is offline.
                       Once the test has completed, the MAC chip should
                       be reinitialized for network operation, but it
```

```
                       should remain offline.

                       If an error occurs during a test, the
                       appropriate test result object will be set
                       to indicate a failure.  The two OBJECT
                       IDENTIFIER values dot3ErrorInitError and
                       dot3ErrorLoopbackError may be used to provided
                       more information as values for an appropriate
                       test result code object."
           ::= { dot3Tests 2 }

       dot3ErrorInitError OBJECT-IDENTITY
           STATUS      current
           DESCRIPTION "Couldn't initialize MAC chip for test."
           ::= { dot3Errors 1 }

       dot3ErrorLoopbackError OBJECT-IDENTITY
           STATUS      current
           DESCRIPTION "Expected data not received (or not received
                       correctly) in loopback test."
           ::= { dot3Errors 2 }


       --  802.3 Hardware Chipsets

       -- The object dot3StatsEtherChipSet is provided to
       -- identify the MAC hardware used to communicate on an
       -- interface.  The following hardware chipsets are
       -- provided:

       dot3ChipSets           OBJECT IDENTIFIER ::= { dot3 8 }

       dot3ChipSetAMD         OBJECT IDENTIFIER ::= { dot3ChipSets 1 }

       dot3ChipSetAMD7990 OBJECT-IDENTITY
           STATUS      current
           DESCRIPTION "The authoritative identifier for the Advanced
                       Micro Devices Am7990 Local Area Network
                       Controller for Ethernet (LANCE)."
           ::= { dot3ChipSetAMD 1 }

       dot3ChipSetAMD79900 OBJECT-IDENTITY
           STATUS      current
           DESCRIPTION "The authoritative identifier for the Advanced
                       Micro Devices Am79900 chip."
           ::= { dot3ChipSetAMD 2 }

       dot3ChipSetAMD79C940 OBJECT-IDENTITY
```

```
           STATUS      current
           DESCRIPTION "The authoritative identifier for the Advanced
                       Micro Devices am79C940 Media Access Controller
                       for Ethernet (MACE)."
           ::= { dot3ChipSetAMD 3 }

       dot3ChipSetIntel      OBJECT IDENTIFIER ::= { dot3ChipSets 2 }

       dot3ChipSetIntel82586 OBJECT-IDENTITY
           STATUS      current
           DESCRIPTION "The authoritative identifier for the Intel
                       82586 IEEE 802.3 Ethernet LAN Coprocessor."
           ::= { dot3ChipSetIntel 1 }

       dot3ChipSetIntel82596 OBJECT-IDENTITY
           STATUS      current
           DESCRIPTION "The authoritative identifier for the Intel
                       82596 High-Performance 32-Bit Local Area Network
                       Coprocessor."
           ::= { dot3ChipSetIntel 2 }

       dot3ChipSetSeeq       OBJECT IDENTIFIER ::= { dot3ChipSets 3 }

       dot3ChipSetSeeq8003 OBJECT-IDENTITY
           STATUS      current
           DESCRIPTION "The authoritative identifier for the SEEQ
                       8003 chip set."
           ::= { dot3ChipSetSeeq 1 }

       dot3ChipSetNational   OBJECT IDENTIFIER ::= { dot3ChipSets 4 }

       dot3ChipSetNational8390 OBJECT-IDENTITY
           STATUS      current
           DESCRIPTION "The authoritative identifier for the National
                       Semiconductor DP8390 Network Interface
                       Controller."
           ::= { dot3ChipSetNational 1 }

       dot3ChipSetNationalSonic OBJECT-IDENTITY
           STATUS      current
           DESCRIPTION "The authoritative identifier for the National
                       Semiconductor DP83932 Systems-Oriented Network
                       Interface Controller (SONIC)."
           ::= { dot3ChipSetNational 2 }

       dot3ChipSetFujitsu    OBJECT IDENTIFIER ::= { dot3ChipSets 5 }

       dot3ChipSetFujitsu86950 OBJECT-IDENTITY
```

```
            STATUS      current
            DESCRIPTION "The authoritative identifier for the Fujitsu
                        86950 chip."
            ::= { dot3ChipSetFujitsu 1 }

        dot3ChipSetDigital    OBJECT IDENTIFIER ::= { dot3ChipSets 6 }

        dot3ChipSetDigitalDC21040 OBJECT-IDENTITY
            STATUS      current
            DESCRIPTION "The authoritative identifier for the Digital
                        DC21040 chip."
            ::= { dot3ChipSetDigital 1 }

        -- For those chipsets not represented above, OBJECT IDENTIFIER
        -- assignment is required in other documentation, e.g.,
        -- assignment within that part of the registration tree
        -- deletaged to individual enterprises (see RFC1155).

        -- conformance information

        etherConformance OBJECT IDENTIFIER ::= { etherMIB 2 }

        etherGroups      OBJECT IDENTIFIER ::= { etherConformance 1 }
        etherCompliances OBJECT IDENTIFIER ::= { etherConformance 2 }


        -- compliance statements

        etherCompliance MODULE-COMPLIANCE
            STATUS      current
            DESCRIPTION "The compliance statement for managed network
                        entities which have ethernet-like network
                        interfaces."

            MODULE  -- this module
                MANDATORY-GROUPS { etherStatsGroup }

                GROUP       etherCollisionTableGroup
                DESCRIPTION "This group is optional. It is appropriate
                            for all systems which have the necessary
                            metering Implementation in such systems is
                            highly recommended."
            ::= { etherCompliances 1 }

        ether100MbsCompliance MODULE-COMPLIANCE
            STATUS      current
            DESCRIPTION "The compliance statement for managed network
                        entities which have 100 Mb/sec ethernet-like
```

```
                     network interfaces."

         MODULE  -- this module
             MANDATORY-GROUPS { etherStats100MbsGroup }

             GROUP       etherCollisionTableGroup
             DESCRIPTION "This group is optional. It is appropriate
                         for all systems which have the necessary
                         metering Implementation in such systems is
                         highly recommended."
         ::= { etherCompliances 2 }

     -- units of conformance

     etherStatsGroup OBJECT-GROUP
         OBJECTS     { dot3StatsIndex,
                       dot3StatsAlignmentErrors,
                       dot3StatsFCSErrors,
                       dot3StatsSingleCollisionFrames,
                       dot3StatsMultipleCollisionFrames,
                       dot3StatsSQETestErrors,
                       dot3StatsDeferredTransmissions,
                       dot3StatsLateCollisions,
                       dot3StatsExcessiveCollisions,
                       dot3StatsInternalMacTransmitErrors,
                       dot3StatsCarrierSenseErrors,
                       dot3StatsFrameTooLongs,
                       dot3StatsInternalMacReceiveErrors,
                       dot3StatsEtherChipSet
                     }
         STATUS      current
         DESCRIPTION "A collection of objects providing information
                     applicable to all ethernet-like network
                     interfaces."
         ::= { etherGroups 1 }


     etherCollisionTableGroup OBJECT-GROUP
         OBJECTS     { dot3CollFrequencies
                     }
         STATUS      current
         DESCRIPTION "A collection of objects providing a histogram
                     of packets successfully transmitted after
                     experiencing exactly N collisions."
         ::= { etherGroups 2 }


     etherStats100MbsGroup OBJECT-GROUP
```

```
           OBJECTS     { dot3StatsIndex,
                         dot3StatsAlignmentErrors,
                         dot3StatsFCSErrors,
                         dot3StatsSingleCollisionFrames,
                         dot3StatsMultipleCollisionFrames,
                         dot3StatsDeferredTransmissions,
                         dot3StatsLateCollisions,
                         dot3StatsExcessiveCollisions,
                         dot3StatsInternalMacTransmitErrors,
                         dot3StatsCarrierSenseErrors,
                         dot3StatsFrameTooLongs,
                         dot3StatsInternalMacReceiveErrors,
                         dot3StatsEtherChipSet,
                         dot3StatsSymbolErrors
                       }
           STATUS      current
           DESCRIPTION "A collection of objects providing information
                       applicable to 100 Mb/sec ethernet-like network
                       interfaces."
           ::= { etherGroups 3 }


   END
```

6. 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.

7. Acknowledgements

This document was produced by the 802.3 Hub MIB Working Group.

This document is almost completely based on both the Standard Ethernet MIB, RFC 1623 [10], and the Proposed Standard Ethernet MIB using the SNMPv2 SMI, RFC 1650 [11], both of which were edited by Frank Kastenholz of FTP Software and produced by the Ethernet MIB Working Group. This document extends those documents by providing support for 100 Mb/sec ethernet interfaces as outlined in [6].

RFC 1623 and RFC 1650, in turn, are based on the Draft Standard Ethernet MIB, RFC 1398 [9], also edited by Frank Kastenholz and produced by the Ethernet MIB Working Group.

RFC 1398, in turn, is based on the Proposed Standard Ethernet MIB, RFC 1284 [8], which was edited by John Cook of Chipcom and produced by the Transmission MIB Working Group. The Ethernet MIB Working Group gathered implementation experience of the variables specified in RFC 1284 and used that information to develop this revised MIB.

RFC 1284, in turn, is based on a document written by Frank Kastenholz, then of Interlan, entitled IEEE 802.3 Layer Management Draft M compatible MIB for TCP/IP Networks [7]. This document has been modestly reworked, initially by the SNMP Working Group, and then

by the Transmission Working Group, to reflect the current conventions for defining objects for MIB interfaces. James Davin, of the MIT Laboratory for Computer Science, and Keith McCloghrie of Hughes LAN Systems, contributed to later drafts of this memo. Marshall Rose of Performance Systems International, Inc. converted the document into its current concise format. Anil Rijsinghani of DEC contributed text that more adequately describes the TDR test. Thanks to Frank Kastenholz of Interlan and Louis Steinberg of IBM for their experimentation.

8. References

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

[2] SNMPv2 Working Group, Case, J., McCloghrie, K., Rose, M., and S. Waldbusser, "Structure of Management Information for Version 2 of the Simple Network Management Protocol (SNMPv2)", RFC 1902, January 1996.

[3] SNMPv2 Working Group, Case, J., McCloghrie, K., Rose, M., and S. Waldbusser, "Textual Conventions for Version 2 of the Simple Network Management Protocol (SNMPv2)", RFC 1903, January 1996.

[4] SNMPv2 Working Group, Case, J., McCloghrie, K., Rose, M., and S. Waldbusser, "Conformance Statements for Version 2 of the Simple Network Management Protocol (SNMPv2)", RFC 1904, January 1996.

[5] IEEE, IEEE 802.3 Layer Management, November 1988.

[6] IEEE, IEEE 802.3u-1995, "10 & 100 Mb/s Management," Section 30, Supplement to IEEE Std 802.3, October 26, 1995.

[7] Kastenholz, F., "IEEE 802.3 Layer Management Draft compatible MIB for TCP/IP Networks", electronic mail message to mib-wg@nnsc.nsf.net, 9 June 1989.

[8] Cook, J., "Definitions of Managed Objects for Ethernet-Like Interface Types", RFC 1284, Chipcom Corporation, December 1991.

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

[10] Kastenholz, F., "Definitions of Managed Objects for the Ethernet-like Interface Types", RFC 1623, FTP Software, Inc., May 1994.

[11] Kastenholz, F., "Definitions of Managed Objects for the Ethernet-like Interface Types using SMIv2", RFC 1650, FTP Software, Inc., August 1994.

[12] McCloghrie, K., and F. Kastenholz, "Evolution of the Interfaces Group of MIB-II", RFC 1573, Hughes LAN Systems, FTP Software, January 1994.

[13] Bradner, S., "Key words for use in RFCs to Indicate Requirements Levels", BCP 14, RFC 2119, March 1997.

[14] deGraaf, K., Romascanu, D., McMaster, D., McCloghrie, K., and S. Roberts, "Definitions of Managed Objects for IEEE 802.3 Medium Attachment Units (MAUs) using SMIv2", RFCXXXX, 3Com Corporation, Madge Networks (Israel) Ltd., Cisco Systems Inc., Cisco Systems Inc., Farallon Computing Inc., October 1997.

[15] Kastenholz, F., "Implementation Notes and Experience for The Internet Ethernet MIB", RFC 1369, FTP Software, October 1992.

9. Security Considerations

Certain management information defined in this MIB may be considered sensitive in some network environments. Therefore, authentication of received SNMP requests and controlled access to management information should be employed in such environments. The method for this authentication is a function of the SNMP Administrative Framework, and has not been expanded by this MIB.

10. Author's Addresses

John Flick
Hewlett-Packard Company
8000 Foothills Blvd. M/S 5556
Roseville, CA 95747-5556

Phone: +1 916 785 4018
Email: johnf@hprnd.rose.hp.com

Jeffrey Johnson
RedBack Networks

2570 North First Street, Suite 410
San Jose, CA, 95131, USA

Phone: +1 408 571 2699
EMail: jeff@redbacknetworks.com

Frank Kastenholz
FTP Software, Inc.
2 High Street
North Andover, Mass, USA 01845

Phone: +1 508 685 4000
EMail: kasten@ftp.com

11. Full Copyright Statement

Table of Contents