Definitions of Managed Objects for Bridges

Status of this Memo

Copyright Notice

Abstract

   This memo 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 MAC bridges based on
the IEEE 802.1D-1990 standard between Local Area Network (LAN)
segments. Provisions are made for support of transparent bridging.
Provisions are also made so that these objects apply to bridges
connected by subnetworks other than LAN segments.

The MIB presented in this memo is a direct translation of the BRIDGE
MIB defined in [RFC1493], to the SMIv2 syntax required for current
IETF MIB standards. This memo obsoletes RFC 1493.

1. The SNMP Management Framework

The SNMP Management Framework presently consists of five major
components:

o   An overall architecture, described in RFC 2571 [RFC2571].

o   Mechanisms for describing and naming objects and events for the
    purpose of management. The first version of this Structure of
    Management Information (SMI) is called SMIv1 and described in
    STD 16, RFC 1155 [RFC1155], STD 16, RFC 1212 [RFC1212] and RFC
    1215 [RFC1215]. The second version, called SMIv2, is described
    in STD 58, RFC 2578 [RFC2578], STD 58, RFC 2579 [RFC2579] and
    STD 58, RFC 2580 [RFC2580].

o   Message protocols for transferring management information. The
    first version of the SNMP message protocol is called SNMPv1 and
    described in STD 15, RFC 1157 [RFC1157]. A second version of
    the SNMP message protocol, which is not an Internet standards
    track protocol, is called SNMPv2c and described in RFC 1901
    [RFC1901] and RFC 1906 [RFC1906]. The third version of the
    message protocol is called SNMPv3 and described in RFC 1906
    [RFC1906], RFC 2572 [RFC2572] and RFC 2574 [RFC2574].

o   Protocol operations for accessing management information. The
    first set of protocol operations and associated PDU formats is
    described in STD 15, RFC 1157 [RFC1157]. A second set of
    protocol operations and associated PDU formats is described in
    RFC 1905 [RFC1905].

o   A set of fundamental applications described in RFC 2573
    [RFC2573] and the view-based access control mechanism described
    in RFC 2575 [RFC2575].

A more detailed introduction to the current SNMP Management Framework
can be found in RFC 2570 [RFC2570].

Managed objects are accessed via a virtual information store, termed

the Management Information Base or MIB. Objects in the MIB are
defined using the mechanisms defined in the SMI.

This memo specifies a MIB module that is compliant to the SMIv2. A

   MIB conforming to the SMIv1 can be produced through the appropriate
   translations. The resulting translated MIB must be semantically
   equivalent, except where objects or events are omitted because no
   translation is possible (use of Counter64). Some machine readable
   information in SMIv2 will be converted into textual descriptions in
   SMIv1 during the translation process. However, this loss of machine
   readable information is not considered to change the semantics of the
   MIB.

2. Overview

   A common device present in many networks is the Bridge. This device
   is used to connect Local Area Network segments below the network
   layer.

   There are two major modes defined for this bridging; transparent and
   source route. The transparent method of bridging is defined in the
   IEEE 802.1D specification [IEEE8021D]. This memo defines those
   objects needed for the management of a bridging entity operating in
   the transparent mode, as well as some objects applicable to all types
   of bridges.

   To be consistent with IAB directives and good engineering practice,
   an explicit attempt was made to keep this MIB as simple as possible.
   This was accomplished by applying the following criteria to objects
   proposed for inclusion:

   (1) Start with a small set of essential objects and add only as
        further objects are needed.

   (2) Require objects be essential for either fault or configuration
        management.

   (3) Consider evidence of current use and/or utility.

   (4) Limit the total of objects.

   (5) Exclude objects which are simply derivable from others in this
        or other MIBs.

   (6) Avoid causing critical sections to be heavily instrumented. The
        guideline that was followed is one counter per critical section
        per layer.

2.1. Structure of MIB

   Objects in this MIB are arranged into groups. Each group is
   organized as a set of related objects. The overall structure and
   assignment of objects to their groups is shown below. Where
   appropriate the corresponding IEEE 802.1D [IEEE8021D] management

object name is also included.

        Bridge MIB Name   IEEE 802.1D Name

```
        dot1dBridge
          dot1dBase
            BridgeAddress              Bridge.BridgeAddress
            NumPorts                   Bridge.NumberOfPorts
            Type
            PortTable
              Port                     BridgePort.PortNumber
              IfIndex
              Circuit
              DelayExceededDiscards      .DiscardTransitDelay
              MtuExceededDiscards        .DiscardOnError
          dot1dStp
            ProtocolSpecification
            Priority                   SpanningTreeProtocol
                                         .BridgePriority
            TimeSinceTopologyChange      .TimeSinceTopologyChange
            TopChanges                   .TopologyChangeCount
            DesignatedRoot               .DesignatedRoot
            RootCost                     .RootCost
            RootPort                     .RootPort
            MaxAge                       .MaxAge
            HelloTime                    .HelloTime
            HoldTime                     .HoldTime
            ForwardDelay                 .ForwardDelay
            BridgeMaxAge                 .BridgeMaxAge
            BridgeHelloTime              .BridgeHelloTime
            BridgeForwardDelay           .BridgeForwardDelay
            PortTable
              Port                     SpanningTreeProtocolPort
                                         .PortNumber
              Priority                   .PortPriority
              State                      .SpanningTreeState
              Enable
              PathCost                   .PortPathCost
              DesignatedRoot             .DesignatedRoot
              DesignatedCost             .DesignatedCost
              DesignatedBridge           .DesignatedBridge
              DesignatedPort             .DesignatedPort
              ForwardTransitions
          dot1dTp
            LearnedEntryDiscards       BridgeFilter.DatabaseSize
                                         .NumDynamic,NumStatic
            AgingTime                  BridgeFilter.AgingTime
            FdbTable
              Address
              Port
              Status
            PortTable
              Port
```

MaxInfo
           InFrames   BridgePort.FramesReceived
           OutFrames   .ForwardOutbound
           InDiscards   .DiscardInbound
       dot1dStatic

```
        StaticTable
          Address
          ReceivePort
          AllowedToGoTo
          Status
```

The following IEEE 802.1D management objects have not been included in the Bridge MIB for the indicated reasons.

```
    IEEE 802.1D Object              Disposition

    Bridge.BridgeName               Same as sysDescr (MIB II)
    Bridge.BridgeUpTime             Same as sysUpTime (MIB II)
    Bridge.PortAddresses            Same as ifPhysAddress (MIB II)
    BridgePort.PortName             Same as ifDescr (MIB II)
    BridgePort.PortType             Same as ifType (MIB II)
    BridgePort.RoutingType          Derivable from the implemented
                                    groups

    SpanningTreeProtocol
        .BridgeIdentifier           Combination of dot1dStpPriority
                                    and dot1dBaseBridgeAddress
        .TopologyChange             Since this is transitory, it
                                    is not considered useful.
    SpanningTreeProtocolPort
        .Uptime                     Same as ifLastChange (MIB II)
        .PortIdentifier             Combination of dot1dStpPort
                                    and dot1dStpPortPriority
        .TopologyChangeAcknowledged Since this is transitory, it
                                    is not considered useful.
        .DiscardLackOfBuffers       Redundant

    Transmission Priority           These objects are not required
                                    as per the Pics Proforma and
                                    not considered useful.
        .TransmissionPriorityName
        .OutboundUserPriority
        .OutboundAccessPriority
```

2.1.1. The dot1dBase Group

This mandatory group contains the objects which are applicable to all types of bridges.

2.1.2. The dot1dStp Group

This group contains the objects that denote the bridge's state with respect to the Spanning Tree Protocol. If a node does not

implemented the Spanning Tree Protocol, this group will not be
implemented.

2.1.3. The dot1dSr Group

This group contains the objects that describe the entity's state with respect to source route bridging. If source routing is not supported this group will not be implemented. This group is applicable to source route only, and SRT bridges. This group will be described in a separate document applicable only to source route bridging.

2.1.4. The dot1dTp Group

This group contains objects that describe the entity's state with respect to transparent bridging. If transparent bridging is not supported this group will not be implemented. This group is applicable to transparent only and SRT bridges.

2.1.5. The dot1dStatic Group

This group contains objects that describe the entity's state with respect to destination-address filtering. If destination-address filtering is not supported this group will not be implemented. This group is applicable to any type of bridge which performs destination-address filtering.

2.2. Relationship to Other MIBs

As described above, some IEEE 802.1D management objects have not been included in this MIB because they overlap with objects in other MIBs applicable to a bridge implementing this MIB. In particular, it is assumed that a bridge implementing this MIB will also implement (at least) the 'system' group and the 'interfaces' group defined in MIB-II [RFC1213].

2.2.1. Relationship to the 'system' group

In MIB-II [RFC1907], 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 as a whole irrespective of whether the entity's sole functionality is bridging, or whether bridging is only a subset of the entity's functionality.

2.2.2. Relationship to the 'interfaces' group

In the Interfaces Group MIB [RFC2863], the 'interfaces' group is defined as being mandatory for all systems and contains information on an entity's interfaces, where each interface is thought of as being attached to a `subnetwork'.  (Note that this term is not to be confused with `subnet' which refers to an addressing partitioning scheme used in the Internet suite of protocols.) The term 'segment' is used in this memo to refer to such a subnetwork, whether it be an Ethernet segment, a 'ring', a WAN link, or even an X.25 virtual

circuit.

   Implicit in this Bridge MIB is the notion of ports on a bridge. Each
   of these ports is associated with one interface of the 'interfaces'

   group, and in most situations, each port is associated with a
   different interface. However, there are situations in which multiple
   ports are associated with the same interface. An example of such a
   situation would be several ports each corresponding one-to-one with
   several X.25 virtual circuits but all on the same interface.

   Each port is uniquely identified by a port number. A port number has
   no mandatory relationship to an interface number, but in the simple
   case a port number will have the same value as the corresponding
   interface's interface number. Port numbers are in the range
   (1..dot1dBaseNumPorts).

   Some entities perform other functionality as well as bridging through
   the sending and receiving of data on their interfaces. In such
   situations, only a subset of the data sent/received on an interface
   is within the domain of the entity's bridging functionality. This
   subset is considered to be delineated according to a set of
   protocols, with some protocols being bridged, and other protocols not
   being bridged. For example, in an entity which exclusively performed
   bridging, all protocols would be considered as being bridged, whereas
   in an entity which performed IP routing on IP datagrams and only
   bridged other protocols, only the non-IP data would be considered as
   being bridged.

   Thus, this Bridge MIB (and in particular, its counters) are
   applicable only to that subset of the data on an entity's interfaces
   which is sent/received for a protocol being bridged. All such data
   is sent/received via the ports of the bridge.

2.3. Textual Conventions

   The datatypes, MacAddress, BridgeId and Timeout, are used as textual
   conventions in this document. Objects defined using these
   conventions are always encoded by means of the rules that define
   their primitive type. Hence, no changes to the SMI or the SNMP are
   necessary to accommodate these textual conventions which are adopted
   merely for the convenience of readers.

3. Definitions

     BRIDGE-MIB DEFINITIONS ::= BEGIN

     -- --- --
     -- MIB for IEEE 802.1D devices
     -- --- --

     IMPORTS
         MODULE-IDENTITY, OBJECT-TYPE, NOTIFICATION-TYPE,
         Counter32, Integer32, TimeTicks, mib-2
             FROM SNMPv2-SMI

```
        TEXTUAL-CONVENTION, MacAddress
            FROM SNMPv2-TC
        MODULE-COMPLIANCE, OBJECT-GROUP, NOTIFICATION-GROUP
            FROM SNMPv2-CONF
```

```
        ;

     bridgeMIB MODULE-IDENTITY
         LAST-UPDATED "200111180000Z"
         ORGANIZATION "IETF Bridge MIB Working Group"
         CONTACT-INFO
             "Email: Bridge-mib@ietf.org"
         DESCRIPTION
             "The Bridge MIB module for managing devices that support
             IEEE 802.1D."
         REVISION      "200111190000Z"
         DESCRIPTION
              "Draft 1: 01 version of draft."
         REVISION      "200106260000Z"
         DESCRIPTION
              "Draft 1: initial translation of RFC 1493 to SMIv2."
         REVISION      "199307310000Z"
         DESCRIPTION
              "RFC 1493: SMIv1 version."
         ::= { dot1dBridge 8 }

dot1dNotification  OBJECT IDENTIFIER ::= { dot1dBridge 0 }

     -- ----------------------------------------------------------- --
     -- Textual Conventions
     -- ----------------------------------------------------------- --

     -- All representations of MAC addresses in this MIB Module use,
     -- as a textual convention (i.e. this convention does not affect
     -- their encoding), the data type MacAddress, defined in
     -- SNMPv2-TC.

     -- Similarly, all representations of Bridge-Id in this MIB
     -- Module use, as a textual convention (i.e. this convention
     -- does not affect their encoding), the data type:

     BridgeId ::= TEXTUAL-CONVENTION
         STATUS      current
         DESCRIPTION
             "The Bridge-Identifier as used in the Spanning Tree
             Protocol to uniquely identify a bridge.  Its first two
             octets (in network byte order) contain a priority value
             and its last 6 octets contain the MAC address used to
             refer to a bridge in a unique fashion (typically, the
             numerically smallest MAC address of all ports on the
             bridge)."
         SYNTAX      OCTET STRING (SIZE (8))

     -- Several objects in this MIB module represent values of timers
```

```
          -- used by the Spanning Tree Protocol.  In this MIB, these
          -- timers have values in units of hundreths of a second (i.e.
          -- 1/100 secs).
          -- These timers, when stored in a Spanning Tree Protocol's BPDU,
          -- are in units of 1/256 seconds.  Note, however, that
```

```
     -- 802.1D-1990 specifies a settable granularity of no more
     -- than 1 second for these timers.  To avoid ambiguity, a data
     -- type is defined here as a textual convention and all
     -- representation of these timers in this MIB module are defined
     -- using this data type.  An algorithm is also defined for
     -- converting between the different units, to ensure a timer's
     -- value is not distorted by multiple conversions.
     -- The data type is:

     Timeout ::= TEXTUAL-CONVENTION
         STATUS      current
         DESCRIPTION
             "A STP timer in units of 1/100 seconds.
             To convert a Timeout value into a value in units of
             1/256 seconds, the following algorithm should be used:

                 b = floor( (n * 256) / 100)

             where:
                 floor   =  quotient [ignore remainder]
                 n is the value in 1/100 second units
                 b is the value in 1/256 second units

             To convert the value from 1/256 second units back to
             1/100 seconds, the following algorithm should be used:

                 n = ceiling( (b * 100) / 256)

             where:
                 ceiling = quotient [if remainder is 0], or
                           quotient + 1 [if remainder is non-zero]
                 n is the value in 1/100 second units
                 b is the value in 1/256 second units


             Note: it is important that the arithmetic operations are
             done in the order specified (i.e., multiply first,
             divide second)."
         SYNTAX      Integer32


     -- ------------------------------------------------------------- --

     dot1dBridge    OBJECT IDENTIFIER ::= { mib-2 17 }


     -- ------------------------------------------------------------- --
     -- groups in the Bridge MIB
     -- ------------------------------------------------------------- --

     dot1dBase       OBJECT IDENTIFIER ::= { dot1dBridge 1 }
     dot1dStp        OBJECT IDENTIFIER ::= { dot1dBridge 2 }
```

```
dot1dSr          OBJECT IDENTIFIER ::= { dot1dBridge 3 }
-- separately documented
```

```
     dot1dTp          OBJECT IDENTIFIER ::= { dot1dBridge 4 }
     dot1dStatic      OBJECT IDENTIFIER ::= { dot1dBridge 5 }

     -- Groups defined in the Bridge Mib Extensions:
     --       pBridgeMIB MODULE-IDENTITY ::= { dot1dBridge 6 }
     --       qBridgeMIB MODULE-IDENTITY ::= { dot1dBridge 7 }

     -- The MODULE-IDENTITY for this MIB has been defined above as:
     --       bridgeMIB MODULE-IDENTITY ::= { dot1dBridge 8 }

     -- The MODULE-IDENTITY for the Source Routing MIB has been
     -- defined in that MIB as:
     --       srMIB MODULE-IDENTITY ::= { dot1dBridge 9 }

     -- Groups defined in the Source Routing MIB:
     --       dot1dPortPair OBJECT IDENTIFIER ::= { dot1dBridge 10 }

     dot1dConformance         OBJECT IDENTIFIER ::= { bridgeMIB 1 }

     -- ------------------------------------------------------------ --
     -- the dot1dBase group
     -- ------------------------------------------------------------ --
     -- Implementation of the dot1dBase group is mandatory for all
     -- bridges.
     -- ------------------------------------------------------------ --

     dot1dBaseBridgeAddress OBJECT-TYPE
         SYNTAX       MacAddress
         MAX-ACCESS   read-only
         STATUS       current
         DESCRIPTION
             "The MAC address used by this bridge when it must be
             referred to in a unique fashion.   It is recommended
             that this be the numerically smallest MAC address of all
             ports that belong to this bridge.  However it is only
             required to be unique.  When concatenated with
             dot1dStpPriority a unique BridgeIdentifier is formed
             which is used in the Spanning Tree Protocol."
         REFERENCE
             "IEEE 802.1D-1990: Sections 6.4.1.1.3 and 3.12.5"
         ::= { dot1dBase 1 }

     dot1dBaseNumPorts OBJECT-TYPE
         SYNTAX       Integer32
         MAX-ACCESS   read-only
         STATUS       current
         DESCRIPTION
             "The number of ports controlled by this bridging
             entity."
```

```
        REFERENCE
            "IEEE 802.1D-1990: Section 6.4.1.1.3"
        ::= { dot1dBase 2 }

    dot1dBaseType OBJECT-TYPE
```

```
     SYNTAX       INTEGER {
                     unknown(1),
                     transparentOnly(2),
                     sourcerouteOnly(3),
                     srt(4)
                 }
     MAX-ACCESS  read-only
     STATUS      current
     DESCRIPTION
         "Indicates what type of bridging this bridge can
         perform.  If a bridge is actually performing a
         certain type of bridging this will be indicated by
         entries in the port table for the given type."
     ::= { dot1dBase 3 }


  -- ------------------------------------------------------------ --
  -- The Generic Bridge Port Table
  -- ------------------------------------------------------------ --

  dot1dBasePortTable OBJECT-TYPE
     SYNTAX       SEQUENCE OF Dot1dBasePortEntry
     MAX-ACCESS  not-accessible
     STATUS      current
     DESCRIPTION
         "A table that contains generic information about every
         port that is associated with this bridge.  Transparent,
         source-route, and srt ports are included."

     ::= { dot1dBase 4 }

  dot1dBasePortEntry OBJECT-TYPE
     SYNTAX       Dot1dBasePortEntry
     MAX-ACCESS  not-accessible
     STATUS      current
     DESCRIPTION
         "A list of information for each port of the bridge."
     REFERENCE
         "IEEE 802.1D-1990: Section 6.4.2, 6.6.1"
     INDEX  { dot1dBasePort }
     ::= { dot1dBasePortTable 1 }

  Dot1dBasePortEntry ::=
     SEQUENCE {
         dot1dBasePort
             Integer32,
         dot1dBasePortIfIndex
             Integer32,
         dot1dBasePortCircuit
```

```
            OBJECT IDENTIFIER,
       dot1dBasePortDelayExceededDiscards
            Counter32,
       dot1dBasePortMtuExceededDiscards
            Counter32
```

```
           }

      dot1dBasePort OBJECT-TYPE
          SYNTAX      Integer32 (1..65535)
          MAX-ACCESS  not-accessible
          STATUS      current
          DESCRIPTION
              "The port number of the port for which this entry
              contains bridge management information."
          ::= { dot1dBasePortEntry 1 }

      dot1dBasePortIfIndex OBJECT-TYPE
          SYNTAX      InterfaceIndex
          MAX-ACCESS  read-only
          STATUS      current
          DESCRIPTION
              "The value of the instance of the ifIndex object,
              defined in MIB-II, for the interface corresponding
              to this port."
          ::= { dot1dBasePortEntry 2 }

      dot1dBasePortCircuit OBJECT-TYPE
          SYNTAX      OBJECT IDENTIFIER
          MAX-ACCESS  read-only
          STATUS      current

          DESCRIPTION
              "For a port which (potentially) has the same value of
              dot1dBasePortIfIndex as another port on the same bridge,
              this object contains the name of an object instance
              unique to this port.  For example, in the case where
              multiple ports correspond one- to-one with multiple X.25
              virtual circuits, this value might identify an (e.g.,
              the first) object instance associated with the X.25
              virtual circuit corresponding to this port.

              For a port which has a unique value of
              dot1dBasePortIfIndex, this object can have the value
              { 0 0 }."
          ::= { dot1dBasePortEntry 3 }

      dot1dBasePortDelayExceededDiscards OBJECT-TYPE
          SYNTAX      Counter32
          MAX-ACCESS  read-only
          STATUS      current
          DESCRIPTION
              "The number of frames discarded by this port due
              to excessive transit delay through the bridge.  It
              is incremented by both transparent and source
```

```
            route bridges."
        REFERENCE
            "IEEE 802.1D-1990: Section 6.6.1.1.3"
        ::= { dot1dBasePortEntry 4 }
```

```
     dot1dBasePortMtuExceededDiscards OBJECT-TYPE
         SYNTAX        Counter32
         MAX-ACCESS  read-only
         STATUS        current
         DESCRIPTION
             "The number of frames discarded by this port due
             to an excessive size.  It is incremented by both
             transparent and source route bridges."
         REFERENCE
             "IEEE 802.1D-1990: Section 6.6.1.1.3"
         ::= { dot1dBasePortEntry 5 }


     -- ------------------------------------------------------------ --
     -- the dot1dStp group
     -- ------------------------------------------------------------ --
     -- Implementation of the dot1dStp group is optional.  It is
     -- implemented by those bridges that support the Spanning Tree
     -- Protocol.
     -- ------------------------------------------------------------ --

     dot1dStpProtocolSpecification OBJECT-TYPE
         SYNTAX        INTEGER {
                         unknown(1),
                         decLb100(2),
                         ieee8021d(3)
                     }
         MAX-ACCESS  read-only
         STATUS        current
         DESCRIPTION
             "An indication of what version of the Spanning Tree
             Protocol is being run.  The value 'decLb100(2)'
             indicates the DEC LANbridge 100 Spanning Tree protocol.
             IEEE 802.1D implementations will return 'ieee8021d(3)'.
             If future versions of the IEEE Spanning Tree Protocol
             are released that are incompatible with the current
             version a new value will be defined."
         ::= { dot1dStp 1 }

     dot1dStpPriority OBJECT-TYPE
         SYNTAX        Integer32 (0..65535)
         MAX-ACCESS  read-write
         STATUS        current
         DESCRIPTION
             "The value of the write-able portion of the Bridge ID,
             i.e., the first two octets of the (8 octet long) Bridge
             ID.  The other (last) 6 octets of the Bridge ID are
             given by the value of dot1dBaseBridgeAddress.
             Permissible values are 0-61440, in steps of 4096."
         REFERENCE
```

```
                " IEEE 802.1t clause 8.10.2, Table 8-4, clause 14.3.""
            ::= { dot1dStp 2 }

    dot1dStpTimeSinceTopologyChange OBJECT-TYPE
        SYNTAX      TimeTicks
```

```
     MAX-ACCESS  read-only
     STATUS      current
     DESCRIPTION
         "The time (in hundredths of a second) since the
         last time a topology change was detected by the
         bridge entity.
         For RSTP, this reports the time since the tcWhile timer for
         any port on this Bridge was non-zero."
     REFERENCE
         " IEEE 802.1w clause 14.8.1.1."
     ::= { dot1dStp 3 }

 dot1dStpTopChanges OBJECT-TYPE
     SYNTAX      Counter32
     MAX-ACCESS  read-only
     STATUS      current
     DESCRIPTION
         "The total number of topology changes detected by
         this bridge since the management entity was last
         reset or initialized.
         For RSTP, this reports the count of times that there have
         been at least one non-zero tcWhile timer on this Bridge."
     REFERENCE
         " IEEE 802.1w clause 14.8.1.1."
     ::= { dot1dStp 4 }

 dot1dStpDesignatedRoot OBJECT-TYPE
     SYNTAX      BridgeId
     MAX-ACCESS  read-only
     STATUS      current
     DESCRIPTION
         "The bridge identifier of the root of the spanning
         tree as determined by the Spanning Tree Protocol
         as executed by this node.  This value is used as
         the Root Identifier parameter in all Configuration
         Bridge PDUs originated by this node."
     REFERENCE
         "IEEE 802.1D-1990: Section 4.5.3.1"
     ::= { dot1dStp 5 }

 dot1dStpRootCost OBJECT-TYPE
     SYNTAX      Integer32
     MAX-ACCESS  read-only
     STATUS      current
     DESCRIPTION
         "The cost of the path to the root as seen from
         this bridge."
     REFERENCE
         "IEEE 802.1D-1990: Section 4.5.3.2"
```

```
        ::= { dot1dStp 6 }

    dot1dStpRootPort OBJECT-TYPE
        SYNTAX      Integer32
        MAX-ACCESS  read-only
```

```
        STATUS       current
        DESCRIPTION
            "The port number of the port which offers the lowest
            cost path from this bridge to the root bridge."
        REFERENCE
            "IEEE 802.1D-1990: Section 4.5.3.3"
        ::= { dot1dStp 7 }

    dot1dStpMaxAge OBJECT-TYPE
        SYNTAX       Timeout
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
            "The maximum age of Spanning Tree Protocol information
            learned from the network on any port before it is
            discarded, in units of hundredths of a second.  This is
            the actual value that this bridge is currently using."
        REFERENCE
            "IEEE 802.1D-1990: Section 4.5.3.4"
        ::= { dot1dStp 8 }

    dot1dStpHelloTime OBJECT-TYPE
        SYNTAX       Timeout
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
            "The amount of time between the transmission of
            Configuration bridge PDUs by this node on any port when
            it is the root of the spanning tree or trying to become
            so, in units of hundredths of a second.  This is the
            actual value that this bridge is currently using."
        REFERENCE
            "IEEE 802.1D-1990: Section 4.5.3.5"
        ::= { dot1dStp 9 }

    dot1dStpHoldTime OBJECT-TYPE
        SYNTAX       Integer32
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
            "This time value determines the interval length
            during which no more than two Configuration bridge
            PDUs shall be transmitted by this node, in units
            of hundredths of a second."
        REFERENCE
            "IEEE 802.1D-1990: Section 4.5.3.14"
        ::= { dot1dStp 10 }

    dot1dStpForwardDelay OBJECT-TYPE
```

```
SYNTAX      Timeout
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "This time value, measured in units of hundredths of a
```

             second, controls how fast a port changes its spanning
             state when moving towards the Forwarding state. The
             value determines how long the port stays in each of the
             Listening and Learning states, which precede the
             Forwarding state. This value is also used, when a
             topology change has been detected and is underway, to
             age all dynamic entries in the Forwarding Database.
             [Note that this value is the one that this bridge is
             currently using, in contrast to
             dot1dStpBridgeForwardDelay which is the value that this
             bridge and all others would start using if/when this
             bridge were to become the root.]"
         REFERENCE
             "IEEE 802.1D-1990: Section 4.5.3.6"
         ::= { dot1dStp 11 }

     dot1dStpBridgeMaxAge OBJECT-TYPE
         SYNTAX   Timeout (600..4000)
         MAX-ACCESS read-write
         STATUS   current
         DESCRIPTION
             "The value that all bridges use for MaxAge when this
             bridge is acting as the root. Note that 802.1D-1990
             specifies that the range for this parameter is related
             to the value of dot1dStpBridgeHelloTime. The
             granularity of this timer is specified by 802.1D-1990 to
             be 1 second. An agent may return a badValue error if a
             set is attempted to a value which is not a whole number
             of seconds."
         REFERENCE
             "IEEE 802.1D-1990: Section 4.5.3.8"
         ::= { dot1dStp 12 }

     dot1dStpBridgeHelloTime OBJECT-TYPE
         SYNTAX   Timeout (100..1000)
         MAX-ACCESS read-write
         STATUS   current
         DESCRIPTION
             "The value that all bridges use for HelloTime when this
             bridge is acting as the root. The granularity of this
             timer is specified by 802.1D- 1990 to be 1 second. An
             agent may return a badValue error if a set is attempted
             to a value which is not a whole number of seconds."
         REFERENCE
             "IEEE 802.1D-1990: Section 4.5.3.9"
         ::= { dot1dStp 13 }

     dot1dStpBridgeForwardDelay OBJECT-TYPE
         SYNTAX   Timeout (400..3000)

```
     MAX-ACCESS  read-write
     STATUS      current
     DESCRIPTION
         "The value that all bridges use for ForwardDelay when
         this bridge is acting as the root.  Note that
```

```
             802.1D-1990 specifies that the range for this parameter
             is related to the value of dot1dStpBridgeMaxAge.  The
             granularity of this timer is specified by 802.1D-1990 to
             be 1 second.  An agent may return a badValue error if a
             set is attempted to a value which is not a whole number
             of seconds."
         REFERENCE
             "IEEE 802.1D-1990: Section 4.5.3.10"
         ::= { dot1dStp 14 }


    -- ------------------------------------------------------------ --
    -- The Spanning Tree Port Table
    -- ------------------------------------------------------------ --

    dot1dStpPortTable OBJECT-TYPE
        SYNTAX       SEQUENCE OF Dot1dStpPortEntry
        MAX-ACCESS   not-accessible
        STATUS       current
        DESCRIPTION
            "A table that contains port-specific information
            for the Spanning Tree Protocol."
        ::= { dot1dStp 15 }

    dot1dStpPortEntry OBJECT-TYPE
        SYNTAX       Dot1dStpPortEntry
        MAX-ACCESS   not-accessible
        STATUS       current
        DESCRIPTION
            "A list of information maintained by every port about
            the Spanning Tree Protocol state for that port."
        INDEX   { dot1dStpPort }
        ::= { dot1dStpPortTable 1 }

    Dot1dStpPortEntry ::=
        SEQUENCE {
            dot1dStpPort
                Integer32,
            dot1dStpPortPriority
                Integer32,
            dot1dStpPortState
                INTEGER,
            dot1dStpPortEnable
                INTEGER,
            dot1dStpPortPathCost
                Integer32,
            dot1dStpPortDesignatedRoot
                BridgeId,
            dot1dStpPortDesignatedCost
                Integer32,
```

```
            dot1dStpPortDesignatedBridge
                BridgeId,
            dot1dStpPortDesignatedPort
                OCTET STRING,
            dot1dStpPortForwardTransitions
```

```
                   Counter32
              }

      dot1dStpPort OBJECT-TYPE
          SYNTAX      Integer32 (1..65535)
          MAX-ACCESS  not-accessible
          STATUS      current
          DESCRIPTION
              "The port number of the port for which this entry
              contains Spanning Tree Protocol management information."
          REFERENCE
              "IEEE 802.1D-1990: Section 6.8.2.1.2"
          ::= { dot1dStpPortEntry 1 }

      dot1dStpPortPriority OBJECT-TYPE
          SYNTAX      Integer32 (0..255)
          MAX-ACCESS  read-write
          STATUS      current
          DESCRIPTION
              "The value of the priority field which is contained in
              the first (in network byte order) octet of the (2 octet
              long) Port ID.  The other octet of the Port ID is given
              by the value of dot1dStpPort.
              Permissible values are 0-240, in steps of 16."
          REFERENCE
              " IEEE 802.1t clause 8.10.2, Table 8-4, clause 14.3."
          ::= { dot1dStpPortEntry 2 }

      dot1dStpPortState OBJECT-TYPE
          SYNTAX        INTEGER {
                          disabled(1),
                          blocking(2),
                          listening(3),
                          learning(4),
                          forwarding(5),
                          broken(6)
                      }
          MAX-ACCESS  read-only
          STATUS      current
          DESCRIPTION
              "The port's current state as defined by application of
              the Spanning Tree Protocol.  This state controls what
              action a port takes on reception of a frame.  If the
              bridge has detected a port that is malfunctioning it
              will place that port into the broken(6) state.  For
              ports which are disabled (see dot1dStpPortEnable), this
              object will have a value of disabled(1)."
          REFERENCE
              "IEEE 802.1D-1990: Section 4.5.5.2"
```

```
        ::= { dot1dStpPortEntry 3 }

   dot1dStpPortEnable OBJECT-TYPE
       SYNTAX      INTEGER {
                     enabled(1),
```

```
                        disabled(2)
                  }
      MAX-ACCESS  read-write
      STATUS      current
      DESCRIPTION
          "The enabled/disabled status of the port."
      REFERENCE
          "IEEE 802.1D-1990: Section 4.5.5.2"
      ::= { dot1dStpPortEntry 4 }

  dot1dStpPortPathCost OBJECT-TYPE
      SYNTAX      Integer32 (1..65535)
      MAX-ACCESS  read-write
      STATUS      current
      DESCRIPTION
          "The contribution of this port to the path cost of
          paths towards the spanning tree root which include
          this port.  802.1D-1990 recommends that the
          default value of this parameter be in inverse
          proportion to the speed of the attached LAN.
          The permissible values must be extended to 1-200,000,000."
      REFERENCE
          " IEEE 802.1t clause 8.10.2, Table 8-5."
      ::= { dot1dStpPortEntry 5 }

  dot1dStpPortDesignatedRoot OBJECT-TYPE
      SYNTAX      BridgeId
      MAX-ACCESS  read-only
      STATUS      current
      DESCRIPTION
          "The unique Bridge Identifier of the Bridge
          recorded as the Root in the Configuration BPDUs
          transmitted by the Designated Bridge for the
          segment to which the port is attached."
      REFERENCE
          "IEEE 802.1D-1990: Section 4.5.5.4"
      ::= { dot1dStpPortEntry 6 }

  dot1dStpPortDesignatedCost OBJECT-TYPE
      SYNTAX      Integer32
      MAX-ACCESS  read-only
      STATUS      current
      DESCRIPTION
          "The path cost of the Designated Port of the segment
          connected to this port.  This value is compared to the
          Root Path Cost field in received bridge PDUs."
      REFERENCE
          "IEEE 802.1D-1990: Section 4.5.5.5"
      ::= { dot1dStpPortEntry 7 }
```

```
dot1dStpPortDesignatedBridge OBJECT-TYPE
    SYNTAX      BridgeId
    MAX-ACCESS  read-only
    STATUS      current
```

            DESCRIPTION
                "The Bridge Identifier of the bridge which this
                port considers to be the Designated Bridge for
                this port's segment."
            REFERENCE
                "IEEE 802.1D-1990: Section 4.5.5.6"
            ::= { dot1dStpPortEntry 8 }

        dot1dStpPortDesignatedPort OBJECT-TYPE
            SYNTAX   OCTET STRING (SIZE (2))
            MAX-ACCESS read-only
            STATUS   current
            DESCRIPTION
                "The Port Identifier of the port on the Designated
                Bridge for this port's segment."
            REFERENCE
                "IEEE 802.1D-1990: Section 4.5.5.7"
            ::= { dot1dStpPortEntry 9 }

        dot1dStpPortForwardTransitions OBJECT-TYPE
            SYNTAX   Counter32
            MAX-ACCESS read-only
            STATUS   current
            DESCRIPTION
                "The number of times this port has transitioned
                from the Learning state to the Forwarding state."
            ::= { dot1dStpPortEntry 10 }

        -- -- --
        -- the dot1dTp group
        -- -- --
        -- Implementation of the dot1dTp group is optional. It is
        -- implemented by those bridges that support the transparent
        -- bridging mode. A transparent or SRT bridge will implement
        -- this group.
        -- -- --

        dot1dTpLearnedEntryDiscards OBJECT-TYPE
            SYNTAX   Counter32
            MAX-ACCESS read-only
            STATUS   current
            DESCRIPTION
                "The total number of Forwarding Database entries, which
                have been or would have been learnt, but have been
                discarded due to a lack of space to store them in the
                Forwarding Database. If this counter is increasing, it
                indicates that the Forwarding Database is regularly
                becoming full (a condition which has unpleasant
                performance effects on the subnetwork). If this counter

has a significant value but is not presently increasing,
                it indicates that the problem has been occurring but is
                not persistent."
            REFERENCE
                "IEEE 802.1D-1990: Section 6.7.1.1.3"

```
     ::= { dot1dTp 1 }

dot1dTpAgingTime OBJECT-TYPE
    SYNTAX       Integer32 (10..1000000)
    MAX-ACCESS   read-write
    STATUS       current
    DESCRIPTION
        "The timeout period in seconds for aging out
        dynamically learned forwarding information.
        802.1D-1990 recommends a default of 300 seconds."
    REFERENCE
        "IEEE 802.1D-1990: Section 6.7.1.1.3"
    ::= { dot1dTp 2 }

-- ------------------------------------------------------------ --
--   The Forwarding Database for Transparent Bridges
-- ------------------------------------------------------------ --

dot1dTpFdbTable OBJECT-TYPE
    SYNTAX       SEQUENCE OF Dot1dTpFdbEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
        "A table that contains information about unicast
        entries for which the bridge has forwarding and/or
        filtering information.  This information is used
        by the transparent bridging function in
        determining how to propagate a received frame."
    ::= { dot1dTp 3 }

dot1dTpFdbEntry OBJECT-TYPE
    SYNTAX       Dot1dTpFdbEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
        "Information about a specific unicast MAC address
        for which the bridge has some forwarding and/or
        filtering information."
    INDEX   { dot1dTpFdbAddress }
    ::= { dot1dTpFdbTable 1 }

Dot1dTpFdbEntry ::=
    SEQUENCE {
        dot1dTpFdbAddress
            MacAddress,
        dot1dTpFdbPort
            Integer32,
        dot1dTpFdbStatus
            INTEGER
```

```
        }

    dot1dTpFdbAddress OBJECT-TYPE
        SYNTAX      MacAddress
        MAX-ACCESS  not-accessible
```

```
        STATUS      current
        DESCRIPTION
            "A unicast MAC address for which the bridge has
            forwarding and/or filtering information."
        REFERENCE
            "IEEE 802.1D-1990: Section 3.9.1, 3.9.2"
        ::= { dot1dTpFdbEntry 1 }

    dot1dTpFdbPort OBJECT-TYPE
        SYNTAX      Integer32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "Either the value '0', or the port number of the port on
            which a frame having a source address equal to the value
            of the corresponding instance of dot1dTpFdbAddress has
            been seen.  A value of '0' indicates that the port
            number has not been learned but that the bridge does
            have some forwarding/filtering information about this
            address (e.g. in the dot1dStaticTable).  Implementors
            are encouraged to assign the port value to this object
            whenever it is learned even for addresses for which the
            corresponding value of dot1dTpFdbStatus is not
            learned(3)."
        ::= { dot1dTpFdbEntry 2 }

    dot1dTpFdbStatus OBJECT-TYPE
        SYNTAX      INTEGER {
                        other(1),
                        invalid(2),
                        learned(3),
                        self(4),
                        mgmt(5)
                    }
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The status of this entry.  The meanings of the
            values are:
                other(1) - none of the following.  This would
                    include the case where some other MIB object
                    (not the corresponding instance of
                    dot1dTpFdbPort, nor an entry in the
                    dot1dStaticTable) is being used to determine if
                    and how frames addressed to the value of the
                    corresponding instance of dot1dTpFdbAddress are
                    being forwarded.
                invalid(2) - this entry is not longer valid (e.g.,
                    it was learned but has since aged-out), but has
```

not yet been flushed from the table.
learned(3) - the value of the corresponding instance
    of dot1dTpFdbPort was learned, and is being
    used.
self(4) - the value of the corresponding instance of

                      dot1dTpFdbAddress represents one of the bridge's
                      addresses. The corresponding instance of
                      dot1dTpFdbPort indicates which of the bridge's
                      ports has this address.
                 mgmt(5) - the value of the corresponding instance of
                      dot1dTpFdbAddress is also the value of an
                      existing instance of dot1dStaticAddress."
          ::= { dot1dTpFdbEntry 3 }

     -- -- --
     -- Port Table for Transparent Bridges
     -- -- --

     dot1dTpPortTable OBJECT-TYPE
         SYNTAX   SEQUENCE OF Dot1dTpPortEntry
         MAX-ACCESS   not-accessible
         STATUS   current
         DESCRIPTION
             "A table that contains information about every port that
             is associated with this transparent bridge."
         ::= { dot1dTp 4 }

     dot1dTpPortEntry OBJECT-TYPE
         SYNTAX   Dot1dTpPortEntry
         MAX-ACCESS   not-accessible
         STATUS   current
         DESCRIPTION
             "A list of information for each port of a transparent
             bridge."
         INDEX   { dot1dTpPort }
         ::= { dot1dTpPortTable 1 }

     Dot1dTpPortEntry ::=
         SEQUENCE {
             dot1dTpPort
                 Integer32,
             dot1dTpPortMaxInfo
                 Integer32,
             dot1dTpPortInFrames
                 Counter32,
             dot1dTpPortOutFrames
                 Counter32,
             dot1dTpPortInDiscards
                 Counter32
         }

     dot1dTpPort OBJECT-TYPE
         SYNTAX   Integer32 (1..65535)
         MAX-ACCESS   not-accessible

```
    STATUS      current
    DESCRIPTION
        "The port number of the port for which this entry
        contains Transparent bridging management information."
    ::= { dot1dTpPortEntry 1 }
```

```
    -- It would be nice if we could use ifMtu as the size of the
    -- largest INFO field, but we can't because ifMtu is defined
    -- to be the size that the (inter-)network layer can use which
    -- can differ from the MAC layer (especially if several layers
    -- of encapsulation are used).

    dot1dTpPortMaxInfo OBJECT-TYPE
        SYNTAX        Integer32
        MAX-ACCESS   read-only
        STATUS        current
        DESCRIPTION
            "The maximum size of the INFO (non-MAC) field that
            this port will receive or transmit."
        ::= { dot1dTpPortEntry 2 }

    dot1dTpPortInFrames OBJECT-TYPE
        SYNTAX         Counter32
        MAX-ACCESS    read-only
        STATUS         current
        DESCRIPTION
            "The number of frames that have been received by this
            port from its segment. Note that a frame received on the
            interface corresponding to this port is only counted by
            this object if and only if it is for a protocol being
            processed by the local bridging function, including
            bridge management frames."
        REFERENCE
            "IEEE 802.1D-1990: Section 6.6.1.1.3"
        ::= { dot1dTpPortEntry 3 }

    dot1dTpPortOutFrames OBJECT-TYPE
        SYNTAX         Counter32
        MAX-ACCESS    read-only
        STATUS         current
        DESCRIPTION
            "The number of frames that have been transmitted by this
            port to its segment.  Note that a frame transmitted on
            the interface corresponding to this port is only counted
            by this object if and only if it is for a protocol being
            processed by the local bridging function, including
            bridge management frames."
        REFERENCE
            "IEEE 802.1D-1990: Section 6.6.1.1.3"
        ::= { dot1dTpPortEntry 4 }

    dot1dTpPortInDiscards OBJECT-TYPE
        SYNTAX         Counter32
        MAX-ACCESS    read-only
```

```
STATUS          current
DESCRIPTION
    "Count of valid frames received which were discarded
    (i.e., filtered) by the Forwarding Process."
REFERENCE
```

```
        "IEEE 802.1D-1990: Section 6.6.1.1.3"
    ::= { dot1dTpPortEntry 5 }


-- ----------------------------------------------------------- --
-- The Static (Destination-Address Filtering) Database
-- ----------------------------------------------------------- --
-- Implementation of this group is optional.
-- ----------------------------------------------------------- --

dot1dStaticTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF Dot1dStaticEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A table containing filtering information configured
        into the bridge by (local or network) management
        specifying the set of ports to which frames received
        from specific ports and containing specific destination
        addresses are allowed to be forwarded.  The value of
        zero in this table as the port number from which frames
        with a specific destination address are received, is
        used to specify all ports for which there is no specific
        entry in this table for that particular destination
        address.  Entries are valid for unicast and for
        group/broadcast addresses."
    REFERENCE
        "IEEE 802.1D-1990: Section 6.7.2"
    ::= { dot1dStatic 1 }

dot1dStaticEntry OBJECT-TYPE
    SYNTAX      Dot1dStaticEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION

        "Filtering information configured into the bridge by
        (local or network) management specifying the set of
        ports to which frames received from a specific port and
        containing a specific destination address are allowed to
        be forwarded."
    REFERENCE
        "IEEE 802.1D-1990: Section 6.7.2"
    INDEX   { dot1dStaticAddress, dot1dStaticReceivePort }
    ::= { dot1dStaticTable 1 }

Dot1dStaticEntry ::=
    SEQUENCE {
        dot1dStaticAddress
            MacAddress,
```

```
            dot1dStaticReceivePort
                Integer32,
            dot1dStaticAllowedToGoTo
                OCTET STRING,
            dot1dStaticStatus
```

```
              INTEGER
        }


    dot1dStaticAddress OBJECT-TYPE
        SYNTAX        MacAddress
        MAX-ACCESS  not-accessible
        STATUS        current
        DESCRIPTION
            "The destination MAC address in a frame to which this
            entry's filtering information applies.  This object can
            take the value of a unicast address, a group address or
            the broadcast address."
        REFERENCE
            "IEEE 802.1D-1990: Section 3.9.1, 3.9.2"
        ::= { dot1dStaticEntry 1 }


    dot1dStaticReceivePort OBJECT-TYPE
        SYNTAX        Integer32 (1..65535)
        MAX-ACCESS  not-accessible
        STATUS        current
        DESCRIPTION
            "Either the value '0', or the port number of the port
            from which a frame must be received in order for this
            entry's filtering information to apply.  A value of zero
            indicates that this entry applies on all ports of the
            bridge for which there is no other applicable entry."
        ::= { dot1dStaticEntry 2 }


    dot1dStaticAllowedToGoTo OBJECT-TYPE

        SYNTAX        OCTET STRING
        MAX-ACCESS  read-write
        STATUS        current
        DESCRIPTION
            "The set of ports to which frames received from a
            specific port and destined for a specific MAC address,
            are allowed to be forwarded.  Each octet within the
            value of this object specifies a set of eight ports,
            with the first octet specifying ports 1 through 8, the
            second octet specifying ports 9 through 16, etc.  Within
            each octet, the most significant bit represents the
            lowest numbered port, and the least significant bit
            represents the highest numbered port.  Thus, each port
            of the bridge is represented by a single bit within the
            value of this object.  If that bit has a value of '1'
            then that port is included in the set of ports; the port
            is not included if its bit has a value of '0'.  (Note
            that the setting of the bit corresponding to the port
            from which a frame is received is irrelevant.)  The
```

default value of this object is a string of ones of
            appropriate length."
        ::= { dot1dStaticEntry 3 }

    dot1dStaticStatus OBJECT-TYPE

```
      SYNTAX        INTEGER {
                        other(1),
                        invalid(2),
                        permanent(3),
                        deleteOnReset(4),
                        deleteOnTimeout(5)
                    }
      MAX-ACCESS  read-write
      STATUS      current
      DESCRIPTION
          "This object indicates the status of this entry.
          The default value is permanent(3).
              other(1) - this entry is currently in use but the
                  conditions under which it will remain so are
                  different from each of the following values.
              invalid(2) - writing this value to the object
                  removes the corresponding entry.
              permanent(3) - this entry is currently in use and
                  will remain so after the next reset of the
                  bridge.
              deleteOnReset(4) - this entry is currently in use
                  and will remain so until the next reset of the
                  bridge.
              deleteOnTimeout(5) - this entry is currently in use
                  and will remain so until it is aged out."
      ::= { dot1dStaticEntry 4 }

-- ------------------------------------------------------------- --
-- Notifications for use by Bridges
-- ------------------------------------------------------------- --
-- Notifications for the Spanning Tree Protocol
-- ------------------------------------------------------------- --

newRoot NOTIFICATION-TYPE
    -- OBJECTS     { }
    STATUS      current
    DESCRIPTION
        "The newRoot trap indicates that the sending agent has
        become the new root of the Spanning Tree; the trap is
        sent by a bridge soon after its election as the new
        root, e.g., upon expiration of the Topology Change Timer
        immediately subsequent to its election.  Implementation
        of this trap is optional."
::= { dot1dNotification 1 }

topologyChange NOTIFICATION-TYPE
    -- OBJECTS     { }
    STATUS      current
    DESCRIPTION
```

"A topologyChange trap is sent by a bridge when any of
its configured ports transitions from the Learning state
to the Forwarding state, or from the Forwarding state to
the Blocking state. The trap is not sent if a newRoot
trap is sent for the same transition. Implementation of

```
             this trap is optional."
      ::= { dot1dNotification 2 }


      -- ---------------------------------------------------------- --
      -- IEEE 802.1D MIB - Conformance Information
      -- ---------------------------------------------------------- --

      dot1dGroups         OBJECT IDENTIFIER ::= { dot1dConformance 1 }
      dot1dCompliances    OBJECT IDENTIFIER ::= { dot1dConformance 2 }


      -- ---------------------------------------------------------- --
      -- units of conformance
      -- ---------------------------------------------------------- --


      -- ---------------------------------------------------------- --
      -- the dot1dBase group
      -- ---------------------------------------------------------- --

      dot1dBaseBridgeGroup OBJECT-GROUP
          OBJECTS {
              dot1dBaseBridgeAddress,
              dot1dBaseNumPorts,
              dot1dBaseType
          }
          STATUS      current
          DESCRIPTION
              "Bridge level information for this device."
          ::= { dot1dGroups 1 }

      dot1dBasePortGroup OBJECT-GROUP
          OBJECTS {
              dot1dBasePortIfIndex,
              dot1dBasePortCircuit,
              dot1dBasePortDelayExceededDiscards,
              dot1dBasePortMtuExceededDiscards
          }
          STATUS      current
          DESCRIPTION
              "Information for each port on this device."
          ::= { dot1dGroups 2 }

      -- ---------------------------------------------------------- --
      -- the dot1dStp group
      -- ---------------------------------------------------------- --

      dot1dStpBridgeGroup OBJECT-GROUP
          OBJECTS {
              dot1dStpProtocolSpecification,
              dot1dStpPriority,
```

```
            dot1dStpTimeSinceTopologyChange,
            dot1dStpTopChanges,
            dot1dStpDesignatedRoot,
            dot1dStpRootCost,
            dot1dStpRootPort,
```

```
            dot1dStpMaxAge,
            dot1dStpHelloTime,
            dot1dStpHoldTime,
            dot1dStpForwardDelay,
            dot1dStpBridgeMaxAge,
            dot1dStpBridgeHelloTime,
            dot1dStpBridgeForwardDelay
        }
        STATUS      current
        DESCRIPTION
            "Bridge level Spanning Tree data for this device."
        ::= { dot1dGroups 3 }

    dot1dStpPortGroup OBJECT-GROUP
        OBJECTS {
            dot1dStpPortPriority,
            dot1dStpPortState,
            dot1dStpPortEnable,
            dot1dStpPortPathCost,
            dot1dStpPortDesignatedRoot,
            dot1dStpPortDesignatedCost,
            dot1dStpPortDesignatedBridge,
            dot1dStpPortDesignatedPort,
            dot1dStpPortForwardTransitions
        }
        STATUS      current
        DESCRIPTION
            "Spanning Tree data for each port on this device."
        ::= { dot1dGroups 4 }

    -- ------------------------------------------------------------ --
    -- the dot1dTp group
    -- ------------------------------------------------------------ --

    dot1dTpBridgeGroup OBJECT-GROUP
        OBJECTS {
            dot1dTpLearnedEntryDiscards,
            dot1dTpAgingTime
        }
        STATUS      current
        DESCRIPTION
            "Bridge level Transparent Bridging data."
        ::= { dot1dGroups 5 }

    dot1dTpFdbGroup OBJECT-GROUP
        OBJECTS {
            dot1dTpFdbPort,
            dot1dTpFdbStatus
        }
```

```
     STATUS      current
     DESCRIPTION
         "Filtering Database information for the Bridge."
     ::= { dot1dGroups 6 }
```

```
     dot1dTpGroup OBJECT-GROUP
         OBJECTS {
             dot1dTpPortMaxInfo,
             dot1dTpPortInFrames,
             dot1dTpPortOutFrames,
             dot1dTpPortInDiscards
         }
         STATUS      current
         DESCRIPTION
             "Dynamic Filtering Database information for each port of
             the Bridge."
         ::= { dot1dGroups 7 }


     -- -------------------------------------------------------------- --
     -- The Static (Destination-Address Filtering) Database
     -- -------------------------------------------------------------- --

     dot1dStaticGroup OBJECT-GROUP
         OBJECTS {
             dot1dStaticAllowedToGoTo,
             dot1dStaticStatus
         }
         STATUS      current
         DESCRIPTION
             "Static Filtering Database information for each port of
             the Bridge."
         ::= { dot1dGroups 8 }


     -- -------------------------------------------------------------- --
     -- The Trap Notfication Group
     -- -------------------------------------------------------------- --

     dot1dNotificationGroup NOTIFICATION-GROUP
         NOTIFICATIONS {
             newRoot,
             topologyChange
         }
         STATUS      current
         DESCRIPTION
             "Group of objects describing notifications (traps)."
         ::= { dot1dGroups 9 }


     -- -------------------------------------------------------------- --
     -- compliance statements
     -- -------------------------------------------------------------- --

     bridgeCompliance MODULE-COMPLIANCE
         STATUS      current
         DESCRIPTION
```

```
        "The compliance statement for device support of bridging
        services."

    MODULE
        MANDATORY-GROUPS {
```

```
                dot1dBaseBridgeGroup,
                dot1dBasePortGroup
            }

            GROUP    dot1dStpBridgeGroup
            DESCRIPTION
                "Implementation of this group is mandatory for
                bridges that support the Spanning Tree Protocol."

            GROUP    dot1dStpPortGroup
            DESCRIPTION
                "Implementation of this group is mandatory for
                bridges that support the Spanning Tree Protocol."

            GROUP    dot1dTpBridgeGroup
            DESCRIPTION
                "Implementation of this group is mandatory for
                bridges that support the transparent bridging
                mode.  A transparent or SRT bridge will implement
                this group."

            GROUP    dot1dTpFdbGroup
            DESCRIPTION
                "Implementation of this group is mandatory for
                bridges that support the transparent bridging
                mode.  A transparent or SRT bridge will implement
                this group."

            GROUP    dot1dTpGroup
            DESCRIPTION
                "Implementation of this group is mandatory for
                bridges that support the transparent bridging
                mode.  A transparent or SRT bridge will implement
                this group."

            GROUP    dot1dStaticGroup
            DESCRIPTION
                "Implementation of this group is optional."

            GROUP dot1dNotificationGroup
            DESCRIPTION
                "Implementation of this group is optional."
            ::= { dot1dCompliances 1 }

    END
```

4. Security Considerations

There are a number of management objects defined in this MIB that
have a MAX-ACCESS clause of read-write and/or read-create. Such
objects may be considered sensitive or vulnerable in some network
environments. The support for SET operations in a non-secure

   environment without proper protection can have a negative effect on
   network operations.

   SNMPv1 by itself is not a secure environment. Even if the network
   itself is secure (for example by using IPSec), even then, there is no
   control as to who on the secure network is allowed to access and
   GET/SET (read/change/create/delete) the objects in this MIB.

   It is recommended that the implementers consider the security
   features as provided by the SNMPv3 framework. Specifically, the use
   of the User-based Security Model RFC 2574 [RFC2574] and the View-
   based Access Control Model RFC 2575 [RFC2575] is recommended.

   It is then a customer/user responsibility to ensure that the SNMP
   entity giving access to an instance of this MIB, is properly
   configured to give access to the objects only to those principals
   (users) that have legitimate rights to indeed GET or SET
   (change/create/delete) them.

5. Acknowledgments

   The MIB presented in this memo is a direct translation of the BRIDGE
   MIB defined in [RFC1493], to the SMIv2 syntax required for current
   IETF MIB standards.

   The original authors were E. Decker, P. Langille, A Rijsinghani and
   K. McCloghrie. Further acknowledgement is given to the members of
   the original Bridge Working Group in [RFC1493].

   This document was produced on behalf of the Bridge MIB Working Group
   in the Operations and Management area of the Internet Engineering
   Task Force.

   The authors wish to thank the members of the Bridge MIB Working Group
   for their many comments and suggestions which improved this effort.

6. References

[RFC2571]   Harrington, D., Presuhn, R., and B. Wijnen, "An Architecture
            for Describing SNMP Management Frameworks", RFC 2571, April
            1999.

[RFC1155]   Rose, M., and K. McCloghrie, "Structure and Identification
            of Management Information for TCP/IP-based Internets", STD
            16, RFC 1155, May 1990.

[RFC1212]   Rose, M., and K. McCloghrie, "Concise MIB Definitions", STD

16, RFC 1212, March 1991.

[RFC1215]   M. Rose, "A Convention for Defining Traps for use with the
            SNMP", RFC 1215, March 1991.

[RFC2578]   McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J.,
             Rose, M., and S. Waldbusser, "Structure of Management
             Information Version 2 (SMIv2)", STD 58, RFC 2578, April
             1999.

[RFC2579]   McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J.,
             Rose, M., and S. Waldbusser, "Textual Conventions for
             SMIv2", STD 58, RFC 2579, April 1999.

[RFC2580]   McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J.,
             Rose, M., and S. Waldbusser, "Conformance Statements for
             SMIv2", STD 58, RFC 2580, April 1999.

[RFC1157]   Case, J., Fedor, M., Schoffstall, M., and J. Davin, "Simple
             Network Management Protocol", STD 15, RFC 1157, May 1990.

[RFC1901]   Case, J., McCloghrie, K., Rose, M., and S. Waldbusser,
             "Introduction to Community-based SNMPv2", RFC 1901, January
             1996.

[RFC1906]   Case, J., McCloghrie, K., Rose, M., and S. Waldbusser,
             "Transport Mappings for Version 2 of the Simple Network
             Management Protocol (SNMPv2)", RFC 1906, January 1996.

[RFC2572]   Case, J., Harrington D., Presuhn R., and B. Wijnen, "Message
             Processing and Dispatching for the Simple Network Management
             Protocol (SNMP)", RFC 2572, April 1999.

[RFC2574]   Blumenthal, U., and B. Wijnen, "User-based Security Model
             (USM) for version 3 of the Simple Network Management
             Protocol (SNMPv3)", RFC 2574, April 1999.

[RFC1905]   Case, J., McCloghrie, K., Rose, M., and S. Waldbusser,
             "Protocol Operations for Version 2 of the Simple Network
             Management Protocol (SNMPv2)", RFC 1905, January 1996.

[RFC2573]   Levi, D., Meyer, P., and B. Stewart, "SNMPv3 Applications",
             RFC 2573, April 1999.

[RFC2575]   Wijnen, B., Presuhn, R., and K. McCloghrie, "View-based
             Access Control Model (VACM) for the Simple Network
             Management Protocol (SNMP)", RFC 2575, April 1999.

[RFC2570]   Case, J., Mundy, R., Partain, D., and B. Stewart,
             "Introduction to Version 3 of the Internet-standard Network
             Management Framework", RFC 2570, April 1999.

[IEEE8021D] ANSI/IEEE Standard 802.1D-1990 MAC Bridges, IEEE Project 802
             Local and Metropolitan Area Networks, (March 8, 1991).

[ISO8021D] ISO DIS 10038 MAC Bridges.

[RFC2863]   McCloghrie, K. and F. Kastenholz, "The Interfaces Group

            MIB", [RFC 2863](), June 2000.

[RFC1907]   Case, J., McCloghrie, K., Rose, M. and S. Waldbusser,
             "Management Information Base for Version 2 of the Simple
             Network Management Protocol (SNMPv2)", [RFC 1907](), January
             1996.

[7](). Changes from [RFC 1493]()

   The following changes have been made from [RFC 1493]().

 (1)   Translated the MIB definition to use SMIv2.

 (2)   Updated the SNMP Framework and references to comply with the
        current IETF guidelines.

 (3)   Updated the Security section to comply with current IETF
        guidelines.

    The following chnages have been made from
        [draft-ietf-bridge-bridgemib-smiv2-00.txt]()

 (1)   Misc. description refernces to IEEE 802.1d documents

 (2)   dot1dNotificationGroup changed from dot1dTrapGroup

 (3)   Misc. additions to some descriptions

[8](). Authors' Addresses

   K.C. Norseth
   Enterasys Networks
   2691 S. Decker Lake Lane
   Salt Lake City, Utah 84119
   Phone: +1 801 887 9823
   Email: knorseth@enterasys.com

[9](). Full Copyright

Internet organizations, except as needed for the purpose of
developing Internet standards in which case the procedures for
copyrights defined in the Internet Standards process must be
followed, or as required to translate it into languages other than

English.

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

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

Table of Contents