NavisCore Enterprise MIB Definitions for MPT

Ascend Communications

November 1998

```
--
************************************************************
******************
--
-- Copyright (c) 1991 by
-- Ascend Communications Corporation, Westford, Mass.
-- All rights reserved.
--
-- This software is the property of Cascade Communications
Corp.
-- This software or any other copies thereof may not be
provided or
-- otherwise made available to any other person outside
Cascade
-- Communications Corp.
--
-- MODULE NAME
--
--          cascmpt.mib
--
-- COMPONENT:
--
--          mib
--
-- ABSTRACT:
--
--          Multipoint-to-Point Tunnel MIB.
--
-- MODIFICATION HISTORY:
--
--
--
--
************************************************************
******************

Ascend-MPT-MIB DEFINITIONS ::= BEGIN

IMPORTS
        IpAddress
                FROM RFC1155-SMI
        OBJECT-TYPE
                FROM RFC-1212
        mpt, nodeTrapSeverity, nodeTrapSequenceNumber
                FROM CASCADE-MIB
        DisplayString
                FROM RFC1213-MIB
        TRAP-TYPE
                FROM RFC-1215;

mptTable OBJECT-TYPE
            SYNTAX      SEQUENCE OF MptEntry
            ACCESS      not-accessible
            STATUS      mandatory
            DESCRIPTION
                        "A list of MPT node entries."
            ::= { mpt 1 }

mptEntry OBJECT-TYPE
            SYNTAX      MptEntry
            ACCESS      not-accessible
            STATUS      mandatory
            DESCRIPTION
                        "The Multipoint-to-Point Tunnel
entry contains objects
                        relevant to Multipoint-to-Point
Tunnels."
            INDEX       { mptDomain, mptNode }
            ::= { mptTable 1 }

MptEntry ::=
            SEQUENCE {
                        mptDomain
                                    INTEGER,
                        mptNode
                                    IpAddress,
                        mptState
                                    INTEGER,
                        mptHops
                                    INTEGER,
                        mptReset
                                    INTEGER,
                        mptPath
                                    OCTET STRING,
                        mptFailReason
                                    INTEGER,
                        mptFailNode
                                    IpAddress,
                        mptFailPort
                                    INTEGER,
```

```
                    mptPathFlags
                                INTEGER
        }

mptDomain OBJECT-TYPE
        SYNTAX      INTEGER {
                   mpt   (1),
              mptPt (2)
            }
        ACCESS      read-only
        STATUS      mandatory
        DESCRIPTION
                    "The domain of a given MPT
destination node."
        ::= { mptEntry 1 }

mptNode OBJECT-TYPE
        SYNTAX      IpAddress
        ACCESS      read-only
        STATUS      mandatory
        DESCRIPTION
                    "The node identifier of a given
MPT destination node."
        ::= { mptEntry 2 }

mptState OBJECT-TYPE
        SYNTAX      INTEGER {
                                retry (1),
                                calling (2),
                                active (3),
                                inactive (4),
                                wcinact (5),
                                wcdel (6)
                    }
        ACCESS      read-only
        STATUS      mandatory
        DESCRIPTION
                    "The current state of the MPT
destination node path."
        ::= { mptEntry 3 }

mptHops OBJECT-TYPE
        SYNTAX      INTEGER
        ACCESS      read-only
        STATUS      mandatory
        DESCRIPTION
                    "The count of hops of the MPT
destination node path."
        ::= { mptEntry 4 }

mptReset OBJECT-TYPE
        SYNTAX      INTEGER {
                                reset (1)
                    }
        ACCESS      read-write
        STATUS      mandatory
        DESCRIPTION
                    "Reset the MPT destination node
path.  A zero is returned
                    when this object is read."
        ::= { mptEntry 5 }

mptPath OBJECT-TYPE
        SYNTAX      OCTET STRING
        ACCESS      read-only
        STATUS      mandatory
        DESCRIPTION
                    "The circuit path consisting of a
sequence of node
                    identifiers and outbound interface
indexes at nodes
                    along the established circuit.
The format is
                    node:interface:node:interface...,
where node and
                    interface are two-octet binary
values."
        ::= { mptEntry 6 }

mptFailReason OBJECT-TYPE
        SYNTAX      INTEGER {
                    none (1),
                    tpcalling (2),
                    vcalling (3),
                    tpdead (4),
                    routelookup (5),
                    confirmtimeout (6),
                    pathclear (7),
                    trunkdown (8),
                    dead (9),
```

```
                        grooming (10),
                        pathregister (11),
                        impurepath (12),
                        rvcdied (13)
                }
                ACCESS      read-only
                STATUS      mandatory
                ::= { mptEntry 7 }

mptFailNode OBJECT-TYPE
                SYNTAX      IpAddress
                ACCESS      read-only
                STATUS      mandatory
                ::= { mptEntry 8 }

mptFailPort OBJECT-TYPE
                SYNTAX      INTEGER
                ACCESS      read-only
                STATUS      mandatory
                ::= { mptEntry 9 }

mptPathFlags OBJECT-TYPE
                SYNTAX      INTEGER
                ACCESS      read-only
                STATUS      mandatory
                ::= { mptEntry 10 }

mptPtTable OBJECT-TYPE
        SYNTAX  SEQUENCE OF MptPtEntry
        ACCESS  not-accessible
        STATUS  mandatory
        DESCRIPTION
                "A list of MPT point-point node entries."
        ::= { mpt 2 }

mptPtEntry OBJECT-TYPE
        SYNTAX  MptPtEntry
        ACCESS  not-accessible
        STATUS  mandatory
        DESCRIPTION
                "The Multipoint-to-Point Tunnel point-
point entry contains objects
                relevant to configured MPT point-point
connections."
        INDEX   { mptPtNode }
                ::= { mptPtTable 1 }

MptPtEntry ::=
        SEQUENCE {
                        mptPtNode
                           IpAddress,
                        mptPtOperStatus
                          INTEGER,
                        mptPtAdminStatus
                          INTEGER,
                        mptPtDefinedPath
                           DisplayString,
                        mptPtDefinedPathHopCnt
                            INTEGER,
                        mptPtDefinedPathEnable
                            INTEGER,
                        mptPtDefinedPathAltOption
                            INTEGER,
                        mptPtUsingDefinedPath
                            INTEGER,
                        mptPtAtmTdIndx
                            INTEGER,
                        mptPtPriority
                            INTEGER,
                        mptPtCir
                            INTEGER,
                        mptPtBc
                            INTEGER,
                        mptPtBe
                            INTEGER,
                        mptPtHops
                            INTEGER,
                        mptPtReset
                            INTEGER,
                        mptPtPath
                            OCTET STRING,
                        mptPtFailReason
                            INTEGER,
                        mptPtFailNode
                            IpAddress,
                        mptPtFailPort
                            INTEGER,
                        mptPtPathFlags
                            INTEGER
}
```

```
mptPtNode OBJECT-TYPE
        SYNTAX  IpAddress
        ACCESS  read-only
        STATUS  mandatory
        DESCRIPTION
               "The node identifier of a given MPT
destination node."
        ::= { mptPtEntry 1 }

mptPtOperStatus OBJECT-TYPE
        SYNTAX INTEGER {
                up (1),
                down (2)
                }
        ACCESS read-only
        STATUS mandatory
        DESCRIPTION
                "The current operational status of the
mptPtEntry."
        ::= { mptPtEntry 2 }

mptPtAdminStatus OBJECT-TYPE
        SYNTAX INTEGER {
                invalid (1),
                activate (2)
                }
        ACCESS read-write
        STATUS mandatory
        DESCRIPTION
                "The admin status of the mptPtEntry."
        ::= { mptPtEntry 3 }

mptPtDefinedPath OBJECT-TYPE
        SYNTAX DisplayString
        ACCESS read-write
        STATUS  mandatory
       DESCRIPTION
                "The circuit path consisting of a sequence
of node
                        identifiers and outbound interface
indexes at nodes
                        along the established circuit.
The format is
                        node:interface:node:interface..., where node
and
                        interface are two-octet binary
values."
        ::= { mptPtEntry 4 }

mptPtDefinedPathHopCnt OBJECT-TYPE
        SYNTAX  INTEGER
        ACCESS  read-only
        STATUS  mandatory
        DESCRIPTION
                "User defined path hop count."
        ::= { mptPtEntry 5 }

mptPtDefinedPathEnable OBJECT-TYPE
        SYNTAX INTEGER {
                disable (1),
                enable (2)
                }
        ACCESS  read-write
        STATUS  mandatory
        DESCRIPTION
                "If set to 1, it means there is an user
defined
                        path for this circuit and it is
enabled."
        ::= { mptPtEntry 6 }

mptPtDefinedPathAltOption OBJECT-TYPE
        SYNTAX INTEGER {
                disable (1),
                enable (2)
                }
        ACCESS  read-write
        STATUS  mandatory
        DESCRIPTION
                "If set to 1, it means that if the user
defined path fails,
                use the ospf-determined path."
        ::= { mptPtEntry 7 }

mptPtUsingDefinedPath OBJECT-TYPE
        SYNTAX  INTEGER
        ACCESS  read-only
        STATUS  mandatory
```

```
        DESCRIPTION
                "If set to 1, it indicates the PVC is
currently using the
                defined path."
        ::= { mptPtEntry 8 }

mptPtAtmTdIndx OBJECT-TYPE
        SYNTAX INTEGER (1..256)
        ACCESS read-write
        STATUS  mandatory
        DESCRIPTION
                " The Quality of Service of the ATM
connection."
        ::= { mptPtEntry 9}


mptPtPriority OBJECT-TYPE
        SYNTAX INTEGER
        ACCESS read-write
        STATUS  mandatory
        DESCRIPTION
               "The priority level (0 through 3) for this
circuit.
                When port service type is defined as
'mono-class', this
                priority means the forward priority of
the circuit.
                When port service type is defined as
'multi-class', this
                priority means the discard priority of
the circuit. "
        ::= { mptPtEntry 10 }

mptPtCir OBJECT-TYPE
        SYNTAX INTEGER
        ACCESS read-write
        STATUS  mandatory
        DESCRIPTION
                "The average number of user data (bits)
that the
                        network agrees to transfer over
the circuit in one
                        direction, measured over the
measurement interval
                        T = cktBc/cktCir."
        ::= { mptPtEntry 11 }

mptPtBc OBJECT-TYPE
        SYNTAX INTEGER
        ACCESS read-write
        STATUS  mandatory
        DESCRIPTION
                "The maximum amount of data (bits) that
the network agrees
                to transfer over the circuit under normal
conditions, during
                the measurement interval."
        ::= { mptPtEntry 12 }

mptPtBe OBJECT-TYPE
        SYNTAX INTEGER
        ACCESS read-write
        STATUS  mandatory
        DESCRIPTION
                "The maximum amount of uncommitted data
(bits) that
                        the network will attempt to
transfer over the circuit
                        during the measurement interval.
By default, if not
                        configured when creating the
entry, the Excess
                        Information Burst Size is set to
the value of ifSpeed."
        ::= { mptPtEntry 13 }

mptPtHops OBJECT-TYPE
            SYNTAX      INTEGER
            ACCESS      read-only
            STATUS      mandatory
            DESCRIPTION
                        "The count of hops of the MPT
destination node path."
            ::= { mptPtEntry 14 }

mptPtReset OBJECT-TYPE
            SYNTAX      INTEGER {
                                    reset (1)
                        }
            ACCESS      read-write
```

```
            STATUS      mandatory
            DESCRIPTION
                        "Reset the MPT destination node
path.  A zero is returned
                        when this object is read."
            ::= { mptPtEntry 15 }

mptPtPath OBJECT-TYPE
            SYNTAX      OCTET STRING
            ACCESS      read-only
            STATUS      mandatory
            DESCRIPTION
                        "The circuit path consisting of a
sequence of node
                        identifiers and outbound interface
indexes at nodes
                        along the established circuit.
The format is
                        node:interface:node:interface...,
where node and
                        interface are two-octet binary
values."
            ::= { mptPtEntry 16 }

mptPtFailReason OBJECT-TYPE
            SYNTAX      INTEGER {
                        none (1),
                        tpcalling (2),
                        vcalling (3),
                        tpdead (4),
                        routelookup (5),
                        confirmtimeout (6),
                        pathclear (7),
                        trunkdown (8),
                        dead (9),
                        grooming (10),
                        pathregister (11),
                        impurepath (12),
                        rvcdied (13)
            }
            ACCESS      read-only
            STATUS      mandatory
            ::= { mptPtEntry 17 }

mptPtFailNode OBJECT-TYPE
            SYNTAX      IpAddress
            ACCESS      read-only
            STATUS      mandatory
            ::= { mptPtEntry 18 }

mptPtFailPort OBJECT-TYPE
            SYNTAX      INTEGER
            ACCESS      read-only
            STATUS      mandatory
            ::= { mptPtEntry 19 }

mptPtPathFlags OBJECT-TYPE
            SYNTAX      INTEGER
            ACCESS      read-only
            STATUS      mandatory
            ::= { mptPtEntry 20 }


mptFwdBase  OBJECT IDENTIFIER  ::= { mpt 3 }

mptFwdAdminStatusOBJECT-TYPE
                                    SYNTAX  INTEGER {
                                        enabled
(1),
                                        disabled
(2)
                                    }
                                    ACCESS      read-write
                                    STATUS  mandatory
                                    DESCRIPTION


"Administrative status for the
                                        Multicast
Forwarding MPT component.
                                        The
default is enabled."
        ::= { mptFwdBase 1 }

mptFwdCnt               OBJECT-TYPE
                                    SYNTAX  INTEGER
                                    ACCESS      read-only
                                    STATUS  mandatory
                                    DESCRIPTION
                                        "Total
number of FMPT circuits currently
```

```
established."
        ::= { mptFwdBase 2 }

mptFwdActiveGrpsOBJECT-TYPE
                                SYNTAX  INTEGER
                                ACCESS      read-only
                                STATUS  mandatory
                                DESCRIPTION
                                        "Total
number of IP multicast groups currently
                                        forwarding
data over FMPT circuits."
        ::= { mptFwdBase 3 }

mptFwdTable OBJECT-TYPE
                        SYNTAX  SEQUENCE OF MptFwdEntry
                        ACCESS        not-accessible
                        STATUS  mandatory
                        DESCRIPTION
                                "A list of existing
Multicast Forwarding MPT
                                ckt entries."
        ::= { mpt 4 }

mptFwdEntry OBJECT-TYPE
        SYNTAX  MptFwdEntry
        ACCESS  not-accessible
        STATUS  mandatory
        DESCRIPTION
                "The Multipoint-to-Point Tunnel point-
point entry contains objects
                relevant to configured MPT point-point
connections."
        INDEX   { mptFwdId }
        ::= { mptFwdTable 1 }

MptFwdEntry ::=
        SEQUENCE {
                        mptFwdId
                           INTEGER,
                        mptFwdState
                          INTEGER,
                        mptFwdLeafCnt
                          INTEGER,
                                                mptFwdGrpCnt
                                                  INTEGER,
                                                mptFwdAge
                                                  INTEGER,
                                                mptFwdVCId
                                                  INTEGER,
                                                mptFwdOutBytes
                                                  INTEGER
}

mptFwdId    OBJECT-TYPE
                                SYNTAX  INTEGER
                                ACCESS  read-only
                                STATUS  mandatory
                                DESCRIPTION
                                        "The identifier of the
root MPT circuit."
        ::= { mptFwdEntry 1 }

mptFwdState OBJECT-TYPE
                                SYNTAX  INTEGER
                                ACCESS  read-only
                                STATUS  mandatory
                                DESCRIPTION
                                        "Current state of the
MPT circuit."
        ::= { mptFwdEntry 2 }

 mptFwdLeafCnt OBJECT-TYPE
                                SYNTAX  INTEGER
                                ACCESS  read-only
                                STATUS  mandatory
                                DESCRIPTION
                                        "The number of
destination MPT leaf switches
                                         connected through
this MPT circuit."
        ::= { mptFwdEntry 3 }

 mptFwdGrpCntOBJECT-TYPE
                                SYNTAX  INTEGER
                                ACCESS  read-only
                                STATUS  mandatory
                                DESCRIPTION
```

```
                                    "The number of IP
multicast groups that use
                                     this MPT circuit for
data forwarding."
        ::= { mptFwdEntry 4 }

 mptFwdAge  OBJECT-TYPE
        SYNTAX  INTEGER
        ACCESS  read-only
        STATUS  mandatory
        DESCRIPTION
               "The age, in seconds, used to determine
whether the MPT
                        circuit is to be cleared and
deleted.  Only inactive
                        circuits are aged, and are taken
down when the age
                        reaches 3600 seconds."
        ::= { mptFwdEntry 5 }

 mptFwdVCId OBJECT-TYPE
        SYNTAX  INTEGER
        ACCESS  read-only
        STATUS  mandatory
        DESCRIPTION
               "The VC ID of the root VC of this MPT
circuit."
        ::= { mptFwdEntry 6 }

 mptFwdOutBytes OBJECT-TYPE
                        SYNTAX  INTEGER
                        ACCESS  read-only
                        STATUS  mandatory
                        DESCRIPTION
                                    "The total # of bytes
forwarded out of the MPT
                                     root VC."
        ::= { mptFwdEntry 7 }


--      The Traps Group
--
--      Definitions for Cascade MPT Specific Traps.
--

mptPtStatusChange TRAP-TYPE
        ENTERPRISE mpt
        VARIABLES  { mptDomain, mptPtNode,
mptPtOperStatus,
                        nodeTrapSeverity,
nodeTrapSequenceNumber }
        DESCRIPTION
                "This trap indicates that an MPT ckt has
changed the state."
        ::= 1



END
```