                    Definitions of Managed Objects
                    for SMDS Interfaces using SMIv2

Status of this Memo

   This document specifies an Internet standards track protocol for the
   Internet community, and requests discussion and suggestions for
   improvements. Please refer to the current edition of the "Internet
   Official Protocol Standards" (STD 1) for the standardization state
   and status of this protocol. Distribution of this memo is unlimited.

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 objects for SMDS
   access interfaces. This includes the following access protocols:

                SIP [13]
                SIP/DXI [18] and [20]
                SIP/FR [19]
                SIP/ATM [24]

   This memo replaces RFC 1304 [12], and defines a MIB module which is
   both compliant to the SNMPv2 SMI and semantically-identical to the
   existing RFC 1304-based definitions.

   This memo also assumes application of the MIB II Interfaces group as
   defined in [9].

Table of Contents

1. The SNMPv2 Network Management Framework

    The SNMPv2 Network Management Framework consists of four major
    components. They are:

        o RFC 1442 [1] which defines the SMI, the mechanisms used for
          describing and naming objects for the purpose of
          management.

        o STD 17, RFC 1213 [6] defines MIB-II, the core set of managed
          objects for the Internet suite of protocols. Reference [12]
          defines the evolution of the Interfaces Group of MIB II in
          terms of extensions and precise applications of the objects.

        o RFC 1445 [4] which defines the administrative and other
          architectural aspects of the framework.

        o RFC 1448 [5] which defines the protocol used for network
          access to managed objects.

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

    This specification makes also use of:

        o RFC 1443 [2] which defines textual conventions for the
          specification of managed objects.

        o RFC 1444 [3] which defines conformance statements for the
          specification of managed objects.

2. Objects

   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) [7]
   defined in the SMI. In particular, each object has a name, a syntax,
   and an encoding. The name is an object identifier, an
   administratively assigned name, which specifies an object type. 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 OBJECT
   DESCRIPTOR, to also refer to the object type.

   The syntax of an object type defines the abstract data structure
   corresponding to that object type. The ASN.1 language is used for
   this purpose. However, the SMI RFC 1442 purposely restricts the
   ASN.1 constructs which may be used. These restrictions are
   explicitly made for simplicity.

   The encoding of an object type is simply how that object type is
   represented using the object type's syntax. Implicitly tied to the
   notion of an object type's syntax and encoding is how the object type
   is represented when being transmitted on the network. The SMI
   specifies the use of the basic encoding rules of ASN.1 [8], subject
   to the additional requirements imposed by the SNMP.

2.1. Format of Definitions

   Section 4 contains contains the specification of all object types
   contained in this MIB module. The object types are defined using the
   conventions defined in the SMI, as amended by the extensions
   specified in the SNMPv2 SMI.

3. Overview

     SMDS is a service that can be provided by numerous interface
     protocols as shown in the following figure:

```
         +------------------+   +------------------+
         | SIP Level 3* [13] |   | SIP Level 3* [13] |
         +------------------+   +------------------+
         | SIP Level 2* [13] |   | DXI Level 2* [20] |
         +------------------+   +------------------+
         | SIP PLCP* [14]   |   |                  |
         +------------------+   | DXI Level 1 [20] |
         | SIP Level 1 [14] |   |                  |
         +------------------+   +------------------+
            SIP based access       DXI based access


         +------------------+   +------------------+
         | SIP Level 3* [13] |   | SIP Level 3* [13] |
         +------------------+   +------------------+
         | ATM  [21]        |   | Frame Relay [19] |
         +------------------+   +------------------+
         | ATM PLCP [21]    |   |                  |
         +------------------+   | Frame Relay [19] |
         | ATM Level 1 [21] |   |   Level 1        |
         +------------------+   +------------------+
            ATM based access       FR based access
```
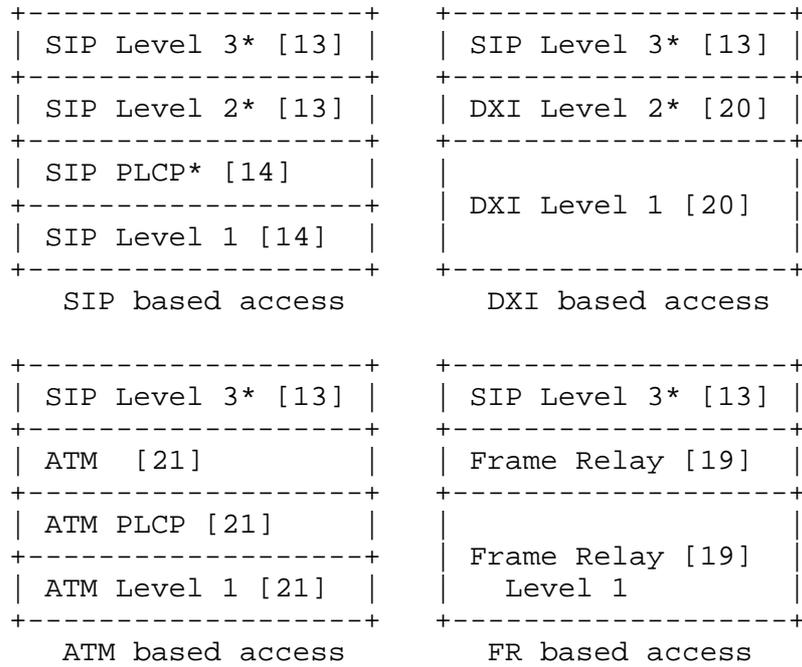
   In the figure below, managed objects for the protocol levels marked
   with a (*) are defined in this memo. Additional managed objects that
   must be used to manage SMDS interfaces are defined in other MIB
   modules as indicated in the figure.

```
         +------------------+   +------------------+
         | SIP Level 3*     |   | SIP Level 3*     |
         +------------------+   +------------------+
         | SIP Level 2*     |   | DXI Level 2*     |
         +------------------+   +------------------+
         | SIP PLCP*        |   |                  |
         +------------------+   | DXI Level 1      |
         | SIP Level 1      |   |                  |
         |    [10] or [11]  |   |     [10]         |
         +------------------+   +------------------+
           SIP based access       DXI based access


         +------------------+   +------------------+
         | SIP Level 3*     |   | SIP Level 3*     |
         +------------------+   +------------------+
         | ATM   [22]       |   | Frame Relay [23] |
         +------------------+   +------------------+
         | ATM PLCP/TC [22] |   |                  |
         +------------------+   | Frame Relay      |
         | ATM Level 1 [10] |   |   Level 1        |
         | [11], or [25]    |   |    [10] or [11]  |
         +------------------+   +------------------+
            ATM based access       FR based access
```
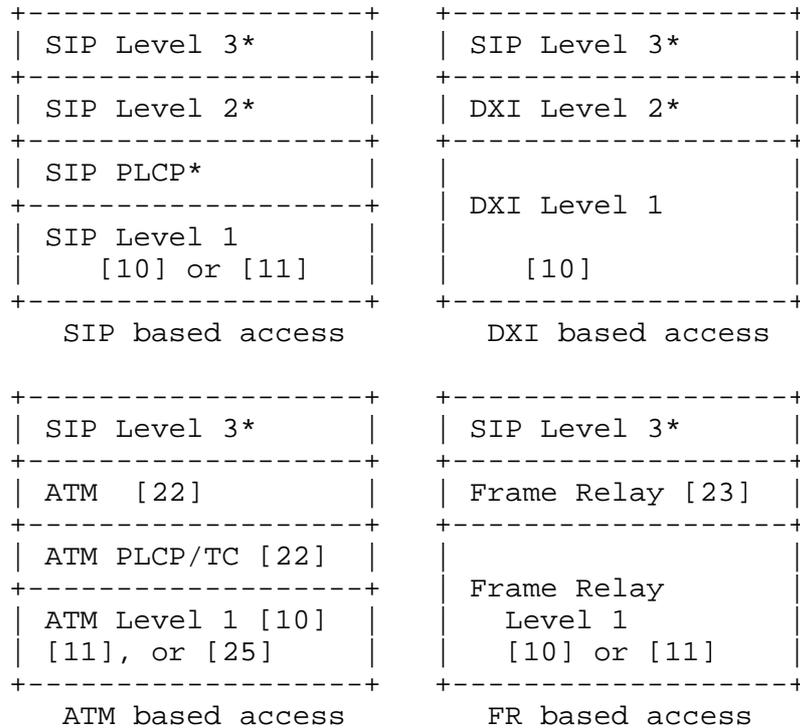
   With the improved interpretation of the MIB II interfaces group [9],
   some objects can be represented by ifTable. This means that these
   objects have been deprecated from the MIB module defined in RFC 1304,
   and ifTable is used instead. No semantical changes have been made to
   these objects. Only the object identifiers and object descriptors
   have been changed to the objects defined in ifTable.

   Implementation experience has shown that the objects
   sipL3UnrecognizedIndividualDAs and sipL3UnrecognizedGAs were not
   supported.

3.1. SIP Level 3

   Objects for SIP Level 3 apply to all methods to access SMDS shown in
   the figures above. With the improved interpretation of the MIB II
   interfaces group, most objects can be represented by ifTable. The
   appropriate mapping is defined below.

   This document does not specify objects for the management of
   subscription or configuration of Subscriber-Network Interfaces
   (SNIs). Those objects are defined in Definitions of Managed Objects
   for SMDS Subscription [17]. Bellcore requirements on these objects
   are specified in TR-TSV-001062 [16].

```
        ifTable Object      Use for
        ========================================================
         ifIndex            Interface index.

         ifDescr            Interface description.
                            For example, SIP Level 3 sublayer
                            of a SNI.

         ifType             Set to 31.

         ifMtu              Set to 9232.

         ifSpeed            Peak bandwidth in bits per second available
                            for use as provided by the supporting Level
                            2 protocol. For example, 1.17 Mbps when
                            using SIP based DS1 SNIs, and 1.536 Mbps
                            when using DXI-based DS1 DXI-SNI.

         ifPhysAddress      OCTET STRING of Size 8. Value is
                            a 16-digit Binary Coded Decimal
                            SMDS address that is
                            assigned to this interface.

         ifAdminStatus      The desired administrative status of the
                            SMDS interface.

         ifOperStatus       The current operational status of the
                            SMDS interface.

         ifLastChange       The elapsed time since the last
                            re-initialization of the interface.
                            The value of sysUpTime at the time the
                            interface entered its current
                            operational state. If the current
                            state was entered prior to the last
                            re-initialization of the local
                            network management subsystem, then
                            this object contains a zero value.

         ifInOctets         Number of received octets at SIP Level 3.
                            For SIP based SNIs, this is the number of
                            sipL2ReceivedCounts multiplied by 44.
```

                ifInUcastPkts   The total number of individually addressed
                                    SIP Level 3 PDUs received from the remote
                                    system across the SNI. The total includes
                                    only unerrored SIP Level 3 PDUs.
                                    [identical to RFC1304:
                                    sipL3ReceivedIndividualDAs]

                ifInDiscards   The number of received SIP Level 3 PDUs
                                    discarded. For SMDS interfaces, this
                                    counter will always be zero.

                ifInErrors   The total number of SIP Level 3 PDUs
                                    received from the remote system that were
                                    discovered to have errors (including
                                    protocol processing and bit errors but
                                    excluding addressing-related errors) and
                                    were discarded. Includes both group
                                    addressed SIP Level 3 PDUs and SIP Level
                                    3 PDUs containing an individual
                                    destination address.
                                    [identical to RFC1304: sipL3Errors]

                ifInUnknownProtos   The number of SIP Level 3 PDUs received
                                    from the remote system with a Source or
                                    Destination Address_Type subfields, (the
                                    four most significant bits of the 64 bit
                                    address field), not equal to the value
                                    1100 or 1110. Also, an error is
                                    considered to have occurred if the
                                    Address_Type field for a Source Address
                                    is equal to 1110 (a group address).
                                    [identical to RFC1304:
                                    sipL3InvalidSMDSAddressTypes]

                ifOutOctets   Number of received octets for transmission
                                    at SIP Level 3. For SIP based
                                    SNIs, this is the number
                                    of sipL2SentCounts multiplied by 44.

                ifOutUcastPkts   The number of individually addressed SIP
                                    Level 3 PDUs that have been sent by this
                                    system across the interface.
                                    [identical to RFC1304:
                                    sipL3SentIndividualDAs]

                ifOutDiscards   The number of SIP Level 3 PDUs discarded in
                                    the egress direction. For SMDS interfaces,
                                    this counter will always be zero.

        ifOutErrors   The number of SIP Level 3 PDUs
                           discarded in the egress
                           direction, because of errors.
                           For SMDS interfaces, this counter will
                           always be zero.

        ifName   The textual name of the interface.
                           If not used, this variable contains
                           a zero-length string.

        ifInMulticastPkts The total number of group addressed SIP
                           Level 3 PDUs received from the remote
                           system across the interface. The total
                           includes only unerrored SIP Level 3 PDUs.
                           [identical to RFC1304: sipL3ReceivedGAs]

        ifInBroadcastPkts This variable is not applicable for SMDS
                           interfaces. Therefore, this counter is
                           always zero.

        ifOutMulticastPkts The number of group addressed SIP Level 3
                           PDUs that have been sent by this system
                           across the interface.
                           [identical to RFC1304: sipL3SentGAs]

        ifOutBroadcastPkts This variable is not applicable for SMDS
                           interfaces. Therefore, this counter is
                           always zero.

        ifLinkUpDownTrapEnble The value of this object is
                                disabled(2) for SIP Level 3 interfaces.

        ifHighSpeed   Set to the user data rate of the
                           interface in millions of bits per second.
                           If the user data rate is less than 1 Mbps,
                           then this value is zero.

        ifPromiscuousMode Set to false(2).

        ifConnectorPresent Set to false(2).

   Consult the Evolution of the Interfaces Group [9] for when to use the
   HC (High Capacity) counters (e.g., ifHCInOctets is a 64-bit counter).

4. Object Definitions

          SIP-MIB DEFINITIONS ::= BEGIN

          IMPORTS
             MODULE-IDENTITY, OBJECT-TYPE, Counter32,
             Integer32, IpAddress   FROM SNMPv2-SMI
             TimeStamp, TEXTUAL-CONVENTION   FROM SNMPv2-TC
             MODULE-COMPLIANCE, OBJECT-GROUP   FROM SNMPv2-CONF
             transmission, ifIndex, mib-2   FROM RFC1213-MIB;

          -- This is the MIB module for the SMDS Interface objects.

          sipMIB MODULE-IDENTITY
              LAST-UPDATED "9403311818Z"
              ORGANIZATION "IETF Interfaces Working Group"
              CONTACT-INFO
                 "   Tracy Brown
                  Postal: Bell Communications Research
                          331 Newman Springs Road
                          P.O. Box 7020
                          Red Bank, NJ 07701-7020
                          US

                     Tel: +1 908 758-2107
                     Fax: +1 908 758-4177
                  E-mail: tacox@mail.bellcore.com

                          Kaj Tesink
                  Postal: Bell Communications Research
                          331 Newman Springs Road
                          P.O. Box 7020
                          Red Bank, NJ 07701-7020
                          US

                     Tel: +1 908 758 5254
                     Fax: +1 908 758 4177
                  E-mail: kaj@cc.bellcore.com."
                DESCRIPTION
                    "The MIB module to describe
                     SMDS interfaces objects."
                ::= { mib-2 36 }

          SMDSAddress ::= TEXTUAL-CONVENTION
              DISPLAY-HINT "1h:"
              STATUS   current
              DESCRIPTION
                  "The 60-bit SMDS address,

```
                 preceded by 4 bits with the following values:
                 1100 when representing an individual address
                 1110 when representing a group address."
             SYNTAX  OCTET STRING (SIZE (8))

     IfIndex ::= TEXTUAL-CONVENTION
       STATUS  current
       DESCRIPTION
         "The value of this object identifies the
         interface for which this entry contains
         management information.  The value of this
         object for a particular interface has the same
         value as the ifIndex object, defined in RFC
         1213, for the same interface."
       SYNTAX Integer32


     sip     OBJECT IDENTIFIER ::= { transmission 31 }
     sipMIBObjects   OBJECT IDENTIFIER ::= { sipMIB 1 }


     -- The SIP Level 3 Group

     sipL3Table  OBJECT-TYPE
         SYNTAX       SEQUENCE OF SipL3Entry
         MAX-ACCESS   not-accessible
         STATUS       current
         DESCRIPTION
             "This table contains SIP L3 parameters and
             state variables, one entry per SIPL3 interface."
         ::= { sip 1 }

     sipL3Entry  OBJECT-TYPE
         SYNTAX       SipL3Entry
         MAX-ACCESS   not-accessible
         STATUS       current
         DESCRIPTION
             "This list contains SIP L3 parameters and
             state variables."
         INDEX    { sipL3Index }
         ::= { sipL3Table 1 }

     SipL3Entry  ::= SEQUENCE {
         sipL3Index                      IfIndex,
         sipL3ReceivedIndividualDAs      Counter32,
         sipL3ReceivedGAs                Counter32,
         sipL3UnrecognizedIndividualDAs  Counter32,
         sipL3UnrecognizedGAs            Counter32,
```

```
         sipL3SentIndividualDAs             Counter32,
         sipL3SentGAs                       Counter32,
         sipL3Errors                        Counter32,
         sipL3InvalidSMDSAddressTypes       Counter32,
         sipL3VersionSupport                Integer32
        }

    sipL3Index  OBJECT-TYPE
        SYNTAX       IfIndex
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
           "The value of this object identifies the SIP
           L3 interface for which this entry contains
           management information. "
        ::= { sipL3Entry 1 }

    sipL3ReceivedIndividualDAs OBJECT-TYPE
        SYNTAX       Counter32
        MAX-ACCESS   read-only
        STATUS       deprecated
    -- Moved to ifTable
    -- ifInUcastPkts defined in [9] must be used instead.
        DESCRIPTION
           "The total number of individually addressed SIP
           Level 3 PDUs received from the remote system
           across the SNI.  The total includes only
           unerrored L3PDUs."
        ::= { sipL3Entry 2 }

    sipL3ReceivedGAs OBJECT-TYPE
        SYNTAX       Counter32
        MAX-ACCESS   read-only
        STATUS       deprecated
    -- Moved to ifTable
    -- ifInMulticastPkts defined in [9] must be used instead.
        DESCRIPTION
           "The total number of group addressed SIP Level 3
           PDUs received from the remote system across the
           SNI.  The total includes only unerrored L3PDUs."
        ::= { sipL3Entry 3 }

    sipL3UnrecognizedIndividualDAs OBJECT-TYPE
        SYNTAX       Counter32
        MAX-ACCESS   read-only
        STATUS       deprecated
        DESCRIPTION
           "The number of SIP Level 3 PDUs received from the
```

```
                    remote system with invalid or unknown individual
                    destination addresses (Destination Address
                    Screening violations are not included).  See SMDS
                    Subscription MIB module."
                 ::= { sipL3Entry 4 }


         sipL3UnrecognizedGAs OBJECT-TYPE
             SYNTAX        Counter32
             MAX-ACCESS    read-only
             STATUS        deprecated
             DESCRIPTION
                 "The number of SIP Level 3 PDUs received from the
                 remote system with invalid or unknown group
                 addresses.  (Destination Address Screening
                 violations are not included).  See SMDS
                 Subscription MIB module."
                 ::= { sipL3Entry 5 }


         sipL3SentIndividualDAs OBJECT-TYPE
             SYNTAX        Counter32
             MAX-ACCESS    read-only
             STATUS        deprecated
     -- Moved to ifTable
     -- ifOutUcastPkts defined in [9] must be used instead.
             DESCRIPTION
                 "The number of individually addressed SIP Level 3
                 PDUs that have been sent by this system across the
                 SNI."
                 ::= { sipL3Entry 6 }


         sipL3SentGAs OBJECT-TYPE
             SYNTAX        Counter32
             MAX-ACCESS    read-only
             STATUS        deprecated
     -- Moved to ifTable
     -- ifOutMulticastPkts defined in [9] must be used instead.
             DESCRIPTION
                 "The number of group addressed SIP L3PDUs that
                 have been sent by this system across the SNI."
                 ::= { sipL3Entry 7 }


     -- The total number of SIP L3PDU errors can be calculated as
     -- (Syntactic errors + Semantic Service errors )
     -- Syntactic errors include:
     --     sipL3Errors
     -- Latest occurrences of syntactic error types are logged in
     --     sipL3PDUErrorTable.
     -- Semantic Service errors include:
```

```
             --      sipL3UnrecognizedIndividualDAs
             --      sipL3UnrecognizedGAs
             --      sipL3InvalidSMDSAddressTypes
             -- Note that public networks supporting SMDS may discard
             -- SIP L3PDUs due to subscription violations.  Related
             -- managed objects are defined in Definitions of Managed
             -- Objects for SMDS Subscription.

             sipL3Errors OBJECT-TYPE
                 SYNTAX       Counter32
                 MAX-ACCESS   read-only
                 STATUS       deprecated
             -- Moved to ifTable
             -- ifInErrors defined in [9] must be used instead.
                 DESCRIPTION
                     "The total number of SIP Level 3 PDUs received
                     from the remote system that were discovered to
                     have errors (including protocol processing and bit
                     errors but excluding addressing-related errors)
                     and were discarded.  Includes both group addressed
                     L3PDUs and L3PDUs containing an individual
                     destination address."
                 ::= { sipL3Entry 8 }

             sipL3InvalidSMDSAddressTypes OBJECT-TYPE
                 SYNTAX       Counter32
                 MAX-ACCESS   read-only
                 STATUS       deprecated
             -- Moved to ifTable
             -- ifInUnknownProtos defined in [9] must be used instead.
                 DESCRIPTION
                     "The number of SIP Level 3 PDUs received from the
                     remote system that had the Source or Destination
                     Address_Type subfields, (the four most significant
                     bits of the 64 bit address field), not equal to
                     the value 1100 or 1110.  Also, an error is
                     considered to have occurred if the Address_Type
                     field for a Source Address, the four most
                     significant bits of the 64 bits, is equal to 1110
                     (a group address)."
                 ::= { sipL3Entry 9 }

             sipL3VersionSupport  OBJECT-TYPE
                 SYNTAX       Integer32
                 MAX-ACCESS   read-only
                 STATUS       current
                 DESCRIPTION
                     "A value which indicates the version(s) of SIP
```

that this interface supports. The value is a sum.
This sum initially takes the value zero. For each
version, V, that this interface supports, 2 raised
to (V - 1) is added to the sum. For example, a
port supporting versions 1 and 2 would have a
value of (2^(1-1)+2^(2-1))=3. The
sipL3VersionSupport is effectively a bit mask with
Version 1 equal to the least significant bit
(LSB)."
::= { sipL3Entry 10 }

-- The SIP Level 2 Group

sipL2Table OBJECT-TYPE
    SYNTAX   SEQUENCE OF SipL2Entry
    MAX-ACCESS not-accessible
    STATUS   current
    DESCRIPTION
       "This table contains SIP L2PDU parameters and
       state variables, one entry per SIP L2 interface."
    ::= { sip 2 }

sipL2Entry OBJECT-TYPE
    SYNTAX   SipL2Entry
    MAX-ACCESS not-accessible
    STATUS   current
    DESCRIPTION
       "This list contains SIP L2 parameters and state
       variables."
    INDEX   { sipL2Index }
    ::= { sipL2Table 1 }

SipL2Entry ::= SEQUENCE {
    sipL2Index   IfIndex,
    sipL2ReceivedCounts   Counter32,
    sipL2SentCounts   Counter32,
    sipL2HcsOrCRCErrors   Counter32,
    sipL2PayloadLengthErrors   Counter32,
    sipL2SequenceNumberErrors   Counter32,
    sipL2MidCurrentlyActiveErrors Counter32,
    sipL2BomOrSSMsMIDErrors   Counter32,
    sipL2EomsMIDErrors   Counter32
   }

sipL2Index OBJECT-TYPE
    SYNTAX   IfIndex
    MAX-ACCESS read-only

```
           STATUS        current
           DESCRIPTION
              "The value of this object identifies the SIP
              interface for which this entry contains management
              information."
           ::= { sipL2Entry 1 }

       sipL2ReceivedCounts OBJECT-TYPE
           SYNTAX        Counter32
           MAX-ACCESS    read-only
           STATUS        current
           DESCRIPTION
              "The number of SIP Level 2 PDUs received from the
              remote system across the SNI. The total includes
              only unerrored L2PDUs."
           ::= { sipL2Entry 2 }

       sipL2SentCounts OBJECT-TYPE
           SYNTAX        Counter32
           MAX-ACCESS    read-only
           STATUS        current
           DESCRIPTION
              "The number of SIP Level 2 PDUs that have been
              sent by this system across the SNI."
           ::= { sipL2Entry 3 }

       -- The following error types are counted, and
       -- preclude sipL2ReceivedCounts to be incremented:
       --     sipL2HcsOrCRCErrors
       --     sipL2PayloadLengthErrors
       --     sipL2SequenceNumberErrors
       --     sipL2BomOrSSMsMIDErrors
       --     sipL2EomsMIDErrors
       -- The receipt of SIP Level 2 PDUs which are BOMs and
       -- for with a MID that is already active will cause
       -- sipL2MidCurrentlyActiveErrors to increment.
       -- Any already accumulated (correct) segmentation
       -- units are discarded.The sipL2ReceivedCounts
       -- is incremented by 1. Thus,
       -- sipL2ReceivedCounts defines the number of
       -- correct SIP Level 2 PDUs delivered to the reassembly
       -- process.

       sipL2HcsOrCRCErrors  OBJECT-TYPE
           SYNTAX        Counter32
           MAX-ACCESS    read-only
           STATUS        current
           DESCRIPTION
```

```
               "The number of received SIP Level 2 PDUs that were
               discovered to have either a Header Check Sequence
               error or a Payload CRC violation."
           ::= { sipL2Entry 4 }

       sipL2PayloadLengthErrors  OBJECT-TYPE
           SYNTAX        Counter32
           MAX-ACCESS    read-only
           STATUS        current
           DESCRIPTION
               "The number of received SIP Level 2 PDUs that had
               Payload Length errors that fall in the following
               specifications:
               - SSM L2_PDU payload length field value less
               - than 28 octets or greater than 44 octets,

               - BOM or COM L2_PDU payload length field not
               - equal to 44 octets,
               - EOM L2_PDU payload length field value less
               - than 4 octets or greater than 44 octets."
           ::= { sipL2Entry 5 }

       sipL2SequenceNumberErrors  OBJECT-TYPE
           SYNTAX        Counter32
           MAX-ACCESS    read-only
           STATUS        current
           DESCRIPTION
               "The number of received SIP Level 2 PDUs that had
               a sequence number within the L2PDU not equal to
               the expected sequence number of the SMDS SS
               receive process."
           ::= { sipL2Entry 6 }

       sipL2MidCurrentlyActiveErrors  OBJECT-TYPE
           SYNTAX        Counter32
           MAX-ACCESS    read-only
           STATUS        current
           DESCRIPTION
               "The number of received SIP Level 2 PDUs that are
               BOMs for which an active receive process is
               already started."
           ::= { sipL2Entry 7 }

       sipL2BomOrSSMsMIDErrors  OBJECT-TYPE
           SYNTAX        Counter32
           MAX-ACCESS    read-only
           STATUS        current
           DESCRIPTION
```

```
            "The number of received SIP Level 2 PDUs that are
            SSMs with a MID not equal to zero or are BOMs with
            MIDs equal to zero."
        ::= { sipL2Entry 8 }

    sipL2EomsMIDErrors  OBJECT-TYPE
        SYNTAX       Counter32
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
            "The number of received SIP Level 2 PDUs that are
            EOMs for which there is no active receive process
            for the MID (i.e., the receipt of an EOM which
            does not correspond to a BOM) OR the EOM has a MID
            equal to zero."
        ::= { sipL2Entry 9 }


    -- The SIP PLCP Group

    sipPLCP OBJECT IDENTIFIER ::= { sip 3 }


    -- The DS1 PLCP Group

    sipDS1PLCPTable  OBJECT-TYPE
        SYNTAX       SEQUENCE OF SipDS1PLCPEntry
        MAX-ACCESS   not-accessible
        STATUS       current
        DESCRIPTION
            "This table contains SIP DS1 PLCP parameters and
            state variables, one entry per SIP port."
        ::= { sipPLCP 1 }

    sipDS1PLCPEntry  OBJECT-TYPE
        SYNTAX       SipDS1PLCPEntry
        MAX-ACCESS   not-accessible
        STATUS       current
        DESCRIPTION
            "This list contains SIP DS1 PLCP parameters and
            state variables."
        INDEX   { sipDS1PLCPIndex }
        ::= { sipDS1PLCPTable 1 }

    SipDS1PLCPEntry  ::= SEQUENCE {
        sipDS1PLCPIndex        IfIndex,
        sipDS1PLCPSEFSs        Counter32,
        sipDS1PLCPAlarmState   INTEGER,
```

```
            sipDS1PLCPUASs         Counter32
            }

    sipDS1PLCPIndex  OBJECT-TYPE
        SYNTAX       IfIndex
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
           "The value of this object identifies the
           interface for which this entry contains management
           information. "
        ::= { sipDS1PLCPEntry 1 }

    sipDS1PLCPSEFSs OBJECT-TYPE
        SYNTAX       Counter32
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
           "A DS1 Severely Errored Framing Second (SEFS) is a
           count of one-second intervals containing one or
           more SEF events.  A Severely Errored Framing (SEF)
           event is declared when an error in the A1 octet
           and an error in the A2 octet of a framing octet
           pair (i.e., errors in both framing octets), or two
           consecutive invalid and/or nonsequential Path
           Overhead Identifier octets are detected."
        ::= { sipDS1PLCPEntry 2 }

    sipDS1PLCPAlarmState OBJECT-TYPE
        SYNTAX   INTEGER     {
                    noAlarm (1),
                    receivedFarEndAlarm (2),
                    incomingLOF (3)
                }
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
           "This variable indicates if there is an alarm
           present for the DS1 PLCP.  The value
           receivedFarEndAlarm means that the DS1 PLCP has
           received an incoming Yellow Signal, the value
           incomingLOF means that the DS1 PLCP has declared a
           loss of frame (LOF) failure condition, and the
           value noAlarm means that there are no alarms
           present.  See TR-TSV-000773 for a description of
           alarm states."
        ::= { sipDS1PLCPEntry 3 }
```

```
        sipDS1PLCPUASs OBJECT-TYPE
            SYNTAX      Counter32
            MAX-ACCESS  read-only
            STATUS      current
            DESCRIPTION
              "The counter associated with the number of
              Unavailable Seconds, as defined by TR-TSV-000773,
              encountered by the PLCP."
            ::= { sipDS1PLCPEntry 4 }


        -- The DS3 PLCP Group

        sipDS3PLCPTable  OBJECT-TYPE
            SYNTAX      SEQUENCE OF SipDS3PLCPEntry
            MAX-ACCESS  not-accessible
            STATUS      current
            DESCRIPTION
              "This table contains SIP DS3 PLCP parameters and
              state variables, one entry per SIP port."
            ::= { sipPLCP 2 }

        sipDS3PLCPEntry  OBJECT-TYPE
            SYNTAX      SipDS3PLCPEntry
            MAX-ACCESS  not-accessible
            STATUS      current
            DESCRIPTION
              "This list contains SIP DS3 PLCP parameters and
              state variables."
            INDEX   { sipDS3PLCPIndex }
            ::= { sipDS3PLCPTable 1 }

        SipDS3PLCPEntry  ::= SEQUENCE {
            sipDS3PLCPIndex       IfIndex,
            sipDS3PLCPSEFSs       Counter32,
            sipDS3PLCPAlarmState  INTEGER,
            sipDS3PLCPUASs        Counter32
            }

        sipDS3PLCPIndex  OBJECT-TYPE
            SYNTAX      IfIndex
            MAX-ACCESS  read-only
            STATUS      current
            DESCRIPTION
              "The value of this object identifies the
              interface for which this entry contains management
              information.  "
            ::= { sipDS3PLCPEntry 1 }
```

```
        sipDS3PLCPSEFSs OBJECT-TYPE
            SYNTAX      Counter32
            MAX-ACCESS  read-only
            STATUS      current
            DESCRIPTION
                "A DS3 Severely Errored Framing Second (SEFS) is a
                count of one-second intervals containing one or
                more SEF events.  A Severely Errored Framing (SEF)
                event is declared when an error in the A1 octet
                and an error in the A2 octet of a framing octet
                pair (i.e., errors in both framing octets), or two
                consecutive invalid and/or nonsequential Path
                Overhead Identifier octets are detected."
            ::= { sipDS3PLCPEntry 2 }

        sipDS3PLCPAlarmState OBJECT-TYPE
            SYNTAX   INTEGER     {
                        noAlarm (1),
                        receivedFarEndAlarm (2),
                        incomingLOF (3)
                    }
            MAX-ACCESS  read-only
            STATUS      current
            DESCRIPTION
                "This variable indicates if there is an alarm
                present for the DS3 PLCP.  The value
                receivedFarEndAlarm means that the DS3 PLCP has
                received an incoming Yellow Signal, the value
                incomingLOF means that the DS3 PLCP has declared a
                loss of frame (LOF) failure condition, and the
                value noAlarm means that there are no alarms
                present.  See TR-TSV-000773 for a description of
                alarm states."
            ::= { sipDS3PLCPEntry 3 }

        sipDS3PLCPUASs OBJECT-TYPE
            SYNTAX      Counter32
            MAX-ACCESS  read-only
            STATUS      current
            DESCRIPTION
                "The counter associated with the number of
                Unavailable Seconds, as defined by TR-TSV-000773,
                encountered by the PLCP."
            ::= { sipDS3PLCPEntry 4 }


        -- The SMDS Applications group
        -- Applications that have been identified for this group are:
```

```
            --             * IP-over-SMDS (details are specified in RFC 1209)

            smdsApplications OBJECT IDENTIFIER ::= { sip 4 }

            ipOverSMDS OBJECT IDENTIFIER ::= { smdsApplications 1 }

            -- Although the objects in this group are read-only, at the
            -- agent's discretion they may be made read-write so that the
            -- management station, when appropriately authorized, may
            -- change the addressing information related to the
            -- configuration of a logical IP subnetwork implemented on
            -- top of SMDS.

            -- This table is necessary to support RFC1209 (IP-over-SMDS)
            -- and gives information on the Group Addresses and ARP
            -- Addresses used in the Logical IP subnetwork.
            -- One SMDS address may be associated with multiple IP
            -- addresses.  One SNI may be associated with multiple LISs.

            ipOverSMDSTable OBJECT-TYPE
                SYNTAX       SEQUENCE OF IpOverSMDSEntry
                MAX-ACCESS   not-accessible
                STATUS       current
                DESCRIPTION
                   "The table of addressing information relevant to
                    this entity's IP addresses."
                ::= { ipOverSMDS 1 }

            ipOverSMDSEntry OBJECT-TYPE
                SYNTAX       IpOverSMDSEntry
                MAX-ACCESS   not-accessible
                STATUS       current
                DESCRIPTION
                   "The addressing information for one of this
                    entity's IP addresses."
                INDEX   { ipOverSMDSIndex, ipOverSMDSAddress }
                ::= { ipOverSMDSTable 1 }

            IpOverSMDSEntry ::=
                SEQUENCE {
                    ipOverSMDSIndex        IfIndex,
                    ipOverSMDSAddress      IpAddress,
                    ipOverSMDSHA           SMDSAddress,
                    ipOverSMDSLISGA        SMDSAddress,
                    ipOverSMDSARPReq       SMDSAddress
                    }

            ipOverSMDSIndex OBJECT-TYPE
```

```
        SYNTAX        IfIndex
        MAX-ACCESS    read-only
        STATUS        current
        DESCRIPTION
           "The value of this object identifies the
           interface for which this entry contains management
           information. "
        ::= { ipOverSMDSEntry 1 }

    ipOverSMDSAddress OBJECT-TYPE
        SYNTAX        IpAddress
        MAX-ACCESS    read-only
        STATUS        current
        DESCRIPTION
           "The IP address to which this entry's addressing
           information pertains."
        ::= { ipOverSMDSEntry 2 }

    ipOverSMDSHA OBJECT-TYPE
        SYNTAX        SMDSAddress
        MAX-ACCESS    read-only
        STATUS        current
        DESCRIPTION
           "The SMDS Individual address of the IP station."
        ::= { ipOverSMDSEntry 3 }

    ipOverSMDSLISGA OBJECT-TYPE
        SYNTAX        SMDSAddress
        MAX-ACCESS    read-only
        STATUS        current
        DESCRIPTION
           "The SMDS Group Address that has been configured
           to identify the SMDS Subscriber-Network Interfaces
           (SNIs) of all members of the Logical IP Subnetwork
           (LIS) connected to the network supporting SMDS."
        ::= { ipOverSMDSEntry 4 }

    ipOverSMDSARPReq OBJECT-TYPE
        SYNTAX        SMDSAddress
        MAX-ACCESS    read-only
        STATUS        current
        DESCRIPTION
           "The SMDS address (individual or group) to which
           ARP Requests are to be sent."
        ::= { ipOverSMDSEntry 5 }


    -- The SMDS Carrier Selection group
```

```
            -- This group is used as a place holder
            -- for carrier selection objects.

            smdsCarrierSelection OBJECT IDENTIFIER ::= { sip 5 }


            -- The SIP Error Log

            sipErrorLog OBJECT IDENTIFIER ::= { sip 6 }

            sipL3PDUErrorTable OBJECT-TYPE
                SYNTAX       SEQUENCE OF SipL3PDUErrorEntry
                MAX-ACCESS   not-accessible
                STATUS       current
                DESCRIPTION
                    "A table that contains the latest occurrence of
                    the following syntactical SIP L3PDU errors:

                    - Destination Address Field Format Error,

                    The following pertains to the 60 least significant
                    bits of the 64 bit address field.  The 60 bits
                    contained in the address subfield can be used to
                    represent addresses up to 15 decimal digits.  Each
                    decimal digit shall be encoded into four bits
                    using Binary Coded Decimal (BCD), with the most
                    significant digit occurring left-most.  If not all
                    15 digits are required, then the remainder of this
                    field shall be padded on the right with bits set
                    to one.  An error is considered to have occurred:
                    a).  if the first four bits of the address
                    subfield are not BCD, OR b).  if the first four
                    bits of the address subfield are populated with
                    the country code value 0001, AND the 40 bits which
                    follow are not Binary Coded Decimal (BCD) encoded
                    values of the 10 digit addresses, OR the remaining
                    16 least significant bits are not populated with
                    1's, OR c).  if the address subfield is not
                    correct according to another numbering plan which
                    is dependent upon the carrier assigning the
                    numbers and offering SMDS.

                    - Source Address Field Format Error,

                    The description of this parameter is the same as
                    the description of the Destination Address Field
                    Format Error.
```

- Invalid BAsize Field Value,

An error is considered to have occurred when the
BAsize field of an SIP L3PDU contains a value less
that 32, greater than 9220 octets without the
CRC32 field present, greater than 9224 octets with
the CRC32 field present, or not equal to a
multiple of 4 octets,

- Invalid Header Extension Length Field Value,

An error is considered to have occurred when the
Header Extension Length field value is not equal
3.

- Invalid Header Extension - Element Length,

An error is considered to have occurred when the
Header Extension - Element Length is greater than
12.

- Invalid Header Extension - Version Element
Position, Length, or Value,

An error is considered to have occurred when a
Version element with Length=3, Type=0, and Value=1
does not appear first within the Header Extension,
or an element Type=0 appears somewhere other than
within the first three octets in the Header
Extension.

- Invalid Header Extension - Carrier Selection
Element Position, Length, Value or Format,

An error is considered to have occurred when a
Carrier Selection element does not appear second
within the Header Extension, if the Element Type
does not equal 1, the Element Length does not
equal 4, 6, or 8, the Element Value field is not
four BCD encoded decimal digits used in specifying
the Carrier Identification Code (CIC), or the
identified CIC code is invalid.

- Header Extension PAD Error

An error is considered to have occurred when the
Header Extension PAD is 9 octets in length, or if
the Header Extension PAD is greater than zero

octets in length and the Header Extension PAD does
not follow all Header Extension elements or does
not begin with at least one octet of all zeros.

- BEtag Mismatch Error,

An error is considered to have occurred when the
Beginning-End Tags in the SIP L3PDU header and
trailer are not equal.

- BAsize Field not equal to Length Field Error,

An error is considered to have occurred when the
value of the BAsize Field does not equal the value
of the Length Field.

- Incorrect Length Error, and

An error is considered to have occurred when the
the Length field value is not equal to the portion
of the SIP L3PDU which extends from the
Destination Address field up to and including the
CRC32 field (if present) or up to and including
the PAD field (if the CRC32 field is not present).
As an optional check, an error is considered to
have occurred when the length of a partially
received SIP L3PDU exceeds the BAsize value.

- MRI Timeout Error.

An error is considered to have occurred when the
elapsed time between receipt of BOM and
corresponding EOM exceeds the value of the MRI
(Message Receive Interval) for a particular
transport signal format.

An entry is indexed by interface number and error
type, and contains Source Address, Destination
Address and a timestamp. All these errors are
counted in the sipL3Errors counter. When
sipL3PDUErrorTimeStamp is equal to zero, the
SipL3PDUErrorEntry does not contain any valid
information."
     ::= { sipErrorLog 1 }

   sipL3PDUErrorEntry OBJECT-TYPE
       SYNTAX   SipL3PDUErrorEntry
       MAX-ACCESS   not-accessible

```
            STATUS       current
            DESCRIPTION
               "An entry in the service disagreement table."
            INDEX   { sipL3PDUErrorIndex, sipL3PDUErrorType }
            ::= { sipL3PDUErrorTable 1 }

        SipL3PDUErrorEntry ::= SEQUENCE {
            sipL3PDUErrorIndex        IfIndex,
            sipL3PDUErrorType         INTEGER,
            sipL3PDUErrorSA           SMDSAddress,
            sipL3PDUErrorDA           SMDSAddress,
            sipL3PDUErrorTimeStamp    TimeStamp
            }

        sipL3PDUErrorIndex OBJECT-TYPE
            SYNTAX       IfIndex
            MAX-ACCESS   read-only
            STATUS       current
            DESCRIPTION
               "The value of this object identifies the
                interface for which this entry contains management
                information."
            ::= { sipL3PDUErrorEntry 1 }

        sipL3PDUErrorType OBJECT-TYPE
            SYNTAX   INTEGER {
              erroredDAFieldFormat (1),
              erroredSAFieldFormat (2),
              invalidBAsizeFieldValue (3),
              invalidHdrExtLength (4),
              invalidHdrExtElementLength (5),
              invalidHdrExtVersionElementPositionLenthOrValue (6),
         invalidHdrExtCarSelectElementPositionLenghtValueOrFormat (7),
              hePADError (8),
              beTagMismatch (9),
              baSizeFieldNotEqualToLengthField (10),
              incorrectLength (11),
              mriTimeout (12)
            }
            MAX-ACCESS  read-only
            STATUS      current
            DESCRIPTION
               "The type of error."
            ::= { sipL3PDUErrorEntry 2 }

        sipL3PDUErrorSA OBJECT-TYPE
            SYNTAX       SMDSAddress
            MAX-ACCESS   read-only
```

```
           STATUS        current
           DESCRIPTION
              "A rejected SMDS source address."
           ::= { sipL3PDUErrorEntry 3 }

       sipL3PDUErrorDA OBJECT-TYPE
           SYNTAX        SMDSAddress
           MAX-ACCESS    read-only
           STATUS        current
           DESCRIPTION
              "A rejected SMDS destination address."
           ::= { sipL3PDUErrorEntry 4 }

       sipL3PDUErrorTimeStamp OBJECT-TYPE
           SYNTAX        TimeStamp
           MAX-ACCESS    read-only
           STATUS        current
           DESCRIPTION
              "The timestamp for the service disagreement.  The
              timestamp contains the value of sysUpTime at the
              latest occurrence of this type of service
              disagreement.  See textual description under
              sipL3PDUErrorTable for boundary conditions."
           ::= { sipL3PDUErrorEntry 5 }


       -- The DXI Group

       sipDxiTable   OBJECT-TYPE
          SYNTAX           SEQUENCE OF SipDxiEntry
          MAX-ACCESS       not-accessible
          STATUS           current
          DESCRIPTION
             "The DXI table."
          ::= { sipMIBObjects 1 }

       sipDxiEntry   OBJECT-TYPE
          SYNTAX      SipDxiEntry
          MAX-ACCESS  not-accessible
          STATUS      current
          DESCRIPTION
                "An entry in the DXI table."
          INDEX { ifIndex }
          ::= { sipDxiTable 1 }

       SipDxiEntry ::=
          SEQUENCE {
             sipDxiCrc
```

```
                    INTEGER,
                sipDxiOutDiscards
                    Counter32,
                sipDxiInErrors
                    Counter32,
                sipDxiInAborts
                    Counter32,
                sipDxiInTestFrames
                    Counter32,
                sipDxiOutTestFrames
                    Counter32,
                sipDxiHbpNoAcks
                    Counter32
            }

        sipDxiCrc   OBJECT-TYPE
           SYNTAX   INTEGER {
                       crc16(1),
                       crc32(2)
                       }
           MAX-ACCESS read-only
           STATUS      current
           DESCRIPTION
                 "The value of this object indicates the type
                 of Frame Checksum used by DXI.  Current
                 choices include CCITT CRC16 or CRC32."
           ::= { sipDxiEntry 1 }

        sipDxiOutDiscards   OBJECT-TYPE
           SYNTAX      Counter32
           MAX-ACCESS read-only
           STATUS      current
           DESCRIPTION
            "The number of outbound frames discarded
            because of congestion."
           ::= { sipDxiEntry 2 }

        sipDxiInErrors   OBJECT-TYPE
           SYNTAX      Counter32
           MAX-ACCESS read-only
           STATUS      current
           DESCRIPTION
                 "The number of inbound frames discarded
                 because of errors such as frame checksum
                 (CRC) violations,
                 non-integral number of octets, address
                 and control field violations, and frame
                 size errors."
```

```
                     ::= { sipDxiEntry 3 }

            sipDxiInAborts  OBJECT-TYPE
               SYNTAX      Counter32
               MAX-ACCESS read-only
               STATUS      current
               DESCRIPTION
                     "The number of inbound frames discarded
                      because of an abort bit sequence (1111111)
                      received before closing flag."
               ::= { sipDxiEntry 4 }

            sipDxiInTestFrames    OBJECT-TYPE
               SYNTAX      Counter32
               MAX-ACCESS read-only
               STATUS      current
               DESCRIPTION
                     "The number of unerrored,
                      inbound Test frames received
                      (generally as part of Heart
                      Beat Poll procedure)."
               ::= { sipDxiEntry 5 }

            sipDxiOutTestFrames    OBJECT-TYPE
               SYNTAX      Counter32
               MAX-ACCESS read-only
               STATUS      current
               DESCRIPTION
                     "The number of unerrored,
                      outbound Test frames sent
                     (generally as part of Heart
                      Beat Poll procedure)."
               ::= { sipDxiEntry 6 }

            sipDxiHbpNoAcks  OBJECT-TYPE
               SYNTAX      Counter32
               MAX-ACCESS read-only
               STATUS      current
               DESCRIPTION
                     "The number of Heart Beat
                      Poll (HBP) No Ack timeouts."
               ::= { sipDxiEntry 7 }


            -- conformance information

            smdsConformance OBJECT IDENTIFIER ::= { sipMIB 2 }
```

```
         smdsGroups      OBJECT IDENTIFIER ::= { smdsConformance 1 }
         smdsCompliances OBJECT IDENTIFIER ::= { smdsConformance 2 }


         -- compliance statements

         smdsCompliance MODULE-COMPLIANCE
             STATUS   current
             DESCRIPTION
                "The compliance statement for SMDS interfaces."

             MODULE  -- this module
                 MANDATORY-GROUPS { sipLevel3Stuff }

                 GROUP       sipLevel2Stuff
                 DESCRIPTION
                     "This group is mandatory only for those
                     interfaces (SNIs) which run SIP Level 2."

                 GROUP       sipDS1PLCPStuff
                 DESCRIPTION
                     "This group is mandatory only for those
                     interfaces (SNIs) which run the DS1 PLCP."

                 GROUP       sipDS3PLCPStuff
                 DESCRIPTION
                     "This group is mandatory only for those
                     interfaces (SNIs) which run the DS3 PLCP."

                 GROUP       sipIPApplicationsStuff
                 DESCRIPTION
                     "This group is mandatory only for interfaces
                      operating IP over SMDS in accordance with
                      RFC1209."

                 GROUP       sipDxiStuff
                 DESCRIPTION
                     "This group is mandatory only for those interfaces
                     (DXI-SNI)
                     which run the DXI protocol."
             ::= { smdsCompliances 1 }

         -- units of conformance

         sipLevel3Stuff    OBJECT-GROUP
             OBJECTS { sipL3Index,
                       sipL3VersionSupport, sipL3PDUErrorIndex,
                       sipL3PDUErrorType,
```

```
                      sipL3PDUErrorSA, sipL3PDUErrorDA,
                      sipL3PDUErrorTimeStamp }
           STATUS   current
           DESCRIPTION
                   "A collection of objects providing information
                   applicable to all SMDS interfaces."
           ::= { smdsGroups 1 }

        sipLevel2Stuff    OBJECT-GROUP
           OBJECTS { sipL2Index, sipL2HcsOrCRCErrors,
                      sipL2PayloadLengthErrors,
                      sipL2SequenceNumberErrors,
                      sipL2MidCurrentlyActiveErrors,
                      sipL2BomOrSSMsMIDErrors,
                      sipL2EomsMIDErrors }
           STATUS   current
           DESCRIPTION
                   "A collection of objects providing information
                   specific to interfaces using the SIP Level 2."
           ::= { smdsGroups 2 }

        sipDS1PLCPStuff    OBJECT-GROUP
           OBJECTS { sipDS1PLCPIndex, sipDS1PLCPSEFSs,
                      sipDS1PLCPAlarmState, sipDS1PLCPUASs }
           STATUS   current
           DESCRIPTION
                   "A collection of objects providing information
                   specific to interfaces using the DS1 PLCP."
           ::= { smdsGroups 3 }

        sipDS3PLCPStuff    OBJECT-GROUP
           OBJECTS { sipDS3PLCPIndex, sipDS3PLCPSEFSs,
                      sipDS3PLCPAlarmState, sipDS3PLCPUASs }
           STATUS   current
           DESCRIPTION
                   "A collection of objects providing information
                   specific to interfaces using the DS3 PLCP."
           ::= { smdsGroups 4 }


        sipIPApplicationsStuff    OBJECT-GROUP
           OBJECTS { ipOverSMDSIndex, ipOverSMDSAddress,
                      ipOverSMDSHA, ipOverSMDSLISGA, ipOverSMDSARPReq }
           STATUS   current
           DESCRIPTION
                   "A collection of objects providing information
                   for running IP over SMDS."
           ::= { smdsGroups 5 }
```

```
        sipDxiStuff      OBJECT-GROUP
            OBJECTS  { sipDxiCrc, sipDxiOutDiscards,
                       sipDxiInErrors, sipDxiInAborts,
                       sipDxiInTestFrames, sipDxiOutTestFrames,
                       sipDxiHbpNoAcks }
            STATUS    current
            DESCRIPTION
                    "A collection of objects providing information
                     specific to interfaces using the DXI protocol."
            ::= { smdsGroups 6 }

        END
```

5. Acknowledgments

   This specification is a product of the ifMIB Working Group.

6. References

   [1] Case, J., McCloghrie, K., Rose, M., and S. Waldbusser, "Structure
       of Management Information for version 2 of the Simple Network
       Management Protocol (SNMPv2)", RFC 1442, SNMP Research, Inc.,
       Hughes LAN Systems, Dover Beach Consulting, Inc., Carnegie Mellon
       University, April 1993.

   [2] Case, J., McCloghrie, K., Rose, M., and S. Waldbusser, "Textual
       Conventions for version 2 of the the Simple Network Management
       Protocol (SNMPv2)", RFC 1443, SNMP Research, Inc., Hughes LAN
       Systems, Dover Beach Consulting, Inc., Carnegie Mellon
       University, April 1993.

   [3] Case, J., McCloghrie, K., Rose, M., and S. Waldbusser,
       "Conformance Statements for version 2 of the the Simple Network
       Management Protocol (SNMPv2)", RFC 1444, SNMP Research, Inc.,
       Hughes LAN Systems, Dover Beach Consulting, Inc., Carnegie Mellon
       University, April 1993.

   [4] Galvin, J., and K. McCloghrie, "Administrative Model for version
       2 of the Simple Network Management Protocol (SNMPv2)", RFC 1445,
       Trusted Information Systems, Hughes LAN Systems, April 1993.

   [5] Case, J., McCloghrie, K., Rose, M., and S. Waldbusser, "Protocol
       Operations for version 2 of the Simple Network Management
       Protocol (SNMPv2)", RFC 1448, SNMP Research, Inc., Hughes LAN
       Systems, Dover Beach Consulting, Inc., Carnegie Mellon
       University, April 1993.

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

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

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

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

  [10] Cox, T., and K. Tesink, Editors, "Definitions of Managed Objects
       for the DS3/E3 Interface Type", RFC 1407, Bellcore, January 1993.

  [11] Baker, F., and J. Watt, Editors, "Definitions of Managed Objects
       for the DS1/E1 Interface Type", RFC 1406, Advanced Computer
       Communications, Newbridge Networks Corporation, January 1993.

  [12] Cox, T., and K. Tesink, Editors, "Definition of Managed Objects
       for the SMDS Interface Type", RFC 1304, Bellcore, February 1992.

  [13] "Generic System Requirements in Support of Switched Multi-megabit
       Data Service", Bellcore Technical Reference, TR-TSV-000772, Issue
       1, May 1991.

  [14] "Local Access System Generic Requirements, Objectives, and
       Interfaces in Support of Switched Multi-megabit Data Service",
       Bellcore Technical Reference, TR-TSV-000773, Issue 1, June 1990.

  [15] Piscitello, D., and J. Lawrence, Editors, The Transmission of IP
       Datagrams over the SMDS Service", RFC 1209, Bell Communications
       Research, March 1991.

  [16] "Generic Requirements For SMDS Customer Network Management
       Service", Bellcore TR-TSV-001062, Issue 1, March 1993, and
       Supplement 1, December 1993.

  [17] Cox, R., and K. Tesink, "Definitions of Managed Objects for SMDS
       Subscription", Version 2.1, Bellcore, August 1992.

   [18] Frame Based Interface Protocol for SMDS Networks - Data Exchange
        Interface / Subscriber Network Interface Revision 1.0 - SMDS
        Interest Group SIG-TS-005/1993, February 2, 1993.

   [19] Frame Based Interface Protocol for SMDS Networks - SIP Relay
        Interface Revision 1.0 - SMDS Interest Group SIG-TS-006/1993,
        February 2, 1993.

   [20] "Generic Requirements For Low Speed SMDS Access", Bellcore TR-
        TSV-001239, Issue 1, December 1993.

   [21] ATM Forum, "ATM User Network Interface Specification", Version
        3.0, September 1993.

   [22] Ahmed, M., and K. Tesink, Editors, "Definitions of Managed
        Objects for ATM Management", RFC 1695, Bellcore, August 1994.

   [23] Brown, R., Editor, "Definitions of Managed Objects for Frame
        Relay Service", RFC 1604, Bellcore, March 1994.

   [24] Specification for Implementation of SMDS over an ATM-based Public
        UNI - Cedric Druce, Max Figueroa, Bellcore - SIG TWG-1993/043,
        SMDS Interest Group Technical Working Group, Work in Progress,
        August 24, 1993.

   [25] Brown, T. and K. Tesink, Editors), "Definitions of Managed
        Objects for the SONET Interface Type, RFC 1595, Bellcore, March
        1994.

7. Security Considerations

    Security issues are not discussed in this memo.

8. Authors' Addresses

    Tracy A. Brown
    Bell Communications Research
    331 Newman Springs Road
    P.O. Box 7020
    Red Bank, NJ   07701-7020

    Phone: (908) 758-2107
    EMail: tacox@mail.bellcore.com

    Kaj Tesink
    Bell Communications Research
    331 Newman Springs Road
    P.O. Box 7020
    Red Bank, NJ   07701-7020

    Phone: (908) 758-5254
    EMail: kaj@cc.bellcore.com