                      Definitions of Managed Objects
                         for Frame Relay Service

Status of this Memo

Copyright Notice

Abstract

   This memo defines an extension to the Management Information Base
   (MIB) for use with network management protocols in Transmission
   Control Protocol/Internet Protocol-based (TCP/IP) internets. In
   particular, it defines objects for managing the frame relay service.

   This document obsoletes RFC 1604.

Table of Contents

1. The SNMP Management Framework

   The SNMP Management Framework presently consists of five major
   components:

   o An overall architecture, described in RFC 2571 [1].

   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 [2], STD 16, RFC 1212 [3] and RFC 1215 [4]. The
      second version, called SMIv2, is described in STD 58, RFC 2578
      [5], STD 58, RFC 2579 [6] and STD 58, RFC 2580 [7].

   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 [8]. A second version of the SNMP
      message protocol, which is not an Internet standards track
      protocol, is called SNMPv2c and described in RFC 1901 [9] and RFC
      1906 [10]. The third version of the message protocol is called
      SNMPv3 and described in RFC 1906 [10], RFC 2572 [11] and RFC 2574
      [12].

   o Protocol operations for accessing management information. The
      first set of protocol operations and associated PDU formats is
      described in STD 15, RFC 1157 [8]. A second set of protocol
      operations and associated PDU formats is described in RFC 1905
      [13].

   o A set of fundamental applications described in RFC 2573 [14] and
      the view-based access control mechanism described in RFC 2575
      [15].

A more detailed introduction to the current SNMP Management Framework
can be found in RFC 2570 [16].

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

These objects are used to manage a frame relay Service. At present,
this applies to the following value of the ifType variable in the
IF-MIB [26]:

        frameRelayService (44)

This section provides an overview and background of how to use this
MIB and other potential MIBs to manage a frame relay service.

2.1. Scope of MIB

The Frame Relay Service MIB supports Customer Network Management
(CNM) of a frame relay network service. Through the use of this and
other related MIBs, a frame relay service customer's NMS can monitor
the customer's UNI/NNI logical ports and PVCs. It provides customers
with access to configuration data, performance monitoring
information, and fault detection for the delivered frame relay
service. As an option, an SNMP agent supporting the Frame Relay
Service MIB may allow customer-initiated PVC management operations
such as creation, deletion, modification, activation, and
deactivation of individual PVCs. However, internal aspects of the
network (e.g., switching elements, line cards, and network routing
tables) are beyond the scope of this MIB.

The Frame Relay Service MIB models all interfaces and PVCs delivered
by a frame relay service within a single virtual SNMP system for the
purpose of comprehensively representing the customer's frame relay
service. The customer's interfaces and PVCs may physically exist on
one or more devices within the network topology. An SNMP agent

providing support for the Frame Relay Service MIB as well as other
appropriate MIBs to model a single virtual frame relay network
service is referred to as a Frame Relay Service (FRS) agent.
Internal communication mechanisms between the FRS agent and
individual devices within the frame relay network delivering the
service are implementation specific and beyond the scope of this MIB.

The customer's NMS will typically access the SNMP agent implementing
the Frame Relay Service MIB over a frame relay permanent virtual
connection (PVC). SNMP access over a frame relay PVC is achieved
through the use of SNMP over UDP over IP encapsulated in Frame Relay
according to STD 55, RFC2427 and ITU X.36 Annex D [23]. Alternate
access mechanisms and SNMP agent implementations are possible.

This MIB will NOT be implemented on user equipment (e.g., DTE). Such
devices are managed using the Frame Relay DTE MIB (RFC2115[18]).
However, concentrators may use the Frame Relay Service MIB instead of
the Frame Relay DTE MIB.

This MIB does not define managed objects for the physical layer.
Existing physical layer MIBs (e.g., DS1 MIB) and Interface MIB will
be used as needed in FRS Agent implementations.

This MIB supports frame relay PVCs. This MIB may be extended at a
later time to handle frame relay SVCs.

A switch implementation may support this MIB for the purpose of
configuration and control of the frame relay service beyond the scope
of traditional customer network management applications. A number of
objects (e.g. frLportTypeAdmin) support administrative actions that
impact the operation of frame relay switch equipment in the network.
This is reflected in the differences between the two MIB compliance
modules:

    o   the frame relay service compliance module
        (frnetservCompliance), and

    o   the frame relay switch compliance module
        (frnetSwitchCompliance).

The frame relay service compliance module does not support the
administrative control objects used for switch management.

2.2. Transiting Multiple Frame Relay Networks

   This MIB is only used to manage a single frame relay service offering
   from one network service provider. Therefore, if a customer PVC
   traverses multiple networks, then the customer must poll a different
   FRS agent within each frame relay network to retrieve the end-to-end
   view of service.

   Figure 1 illustrates a customer ("User B") NMS accessing FRS agents
   in three different frame relay networks (I, J, and K).

```
                 +------------------------------------+
                 | Customer Network Management Station |
                 |           (SNMP based)             |
                 +------------------------------------+
                       ^               ^              ^
                       |               |              |
                       |               |              |
      UNI              |     NNI       |     NNI      |    UNI
       |               |      ^        |      ^       |      ^
       |   +----------+ |  +----------+ |  +----------+ |
       |   |          | |  |          | |  |          | |
Originating |   |   FR     | |  |   FR     | |  |   FR     | |Terminating
  +--------+ |  Network I  | | Network J | | Network K | | +--------+
  |       | | |          | | |          | | |          | | |        |
  |       | |---|        |    |---|       |    |---|    | |---| User B |
  |       | | | |        | | | |        | | | |        | | | |        |
  |   ///////////////////////////////////////////////////////////////// |
  |       | | | |        | | | |        | | | |        | | | |        |
  +--------+ | +----------+ | +----------+ | +----------+ | +--------+
             |              |              |              |
             |              |              |              |
             | PVC Segment 1 | PVC Segment 2 | PVC Segment 3 |
             |<------------->|<------------->|<------------->|
             |              |              |              |
             |            Multi-network PVC              |
             |<---------------------------------------->|
             |   NNI = Network-to Network Interface      |
                 UNI = User-to-Network Interface
```
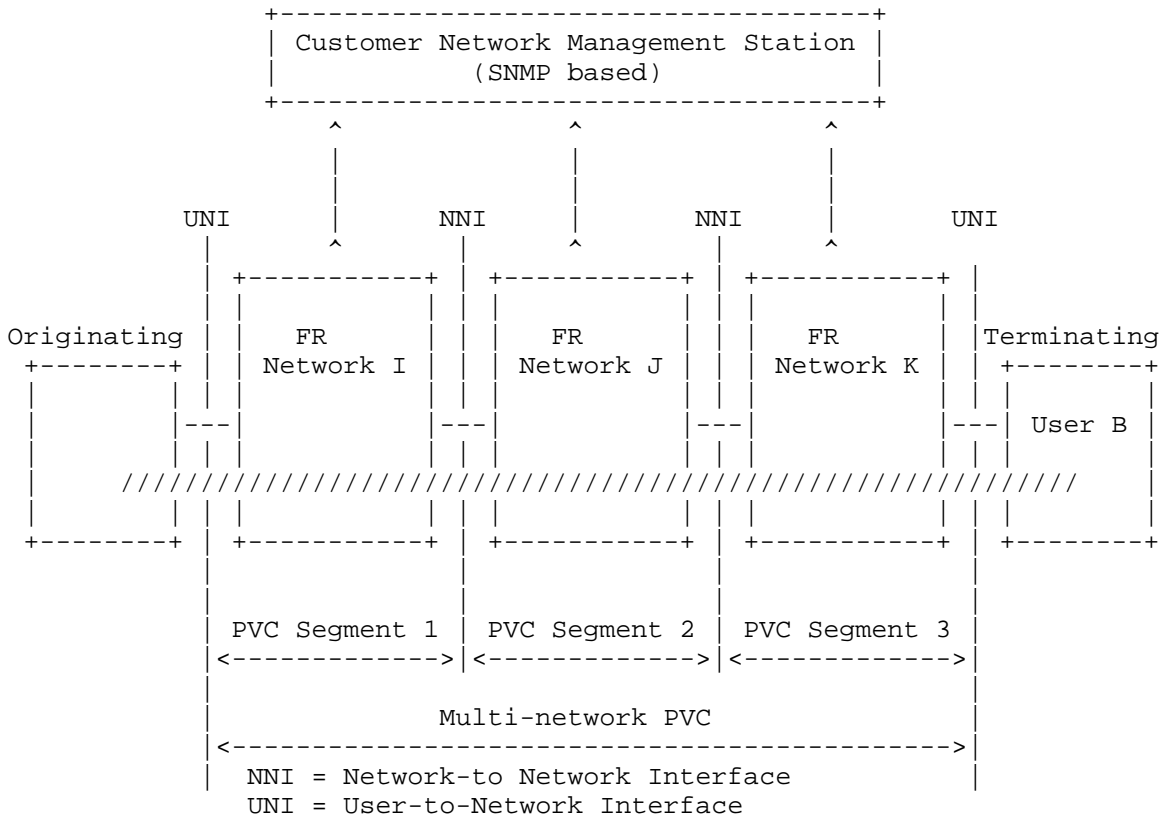
                    Figure 1, Multi-network PVC

2.3. Access Control

   A frame relay network is shared amongst many frame relay subscribers.
   Each subscriber will only have access to their information (e.g.,
   information with respect to their interfaces and PVCs). The FRS agent
   should provide instance level granularity for MIB views.

2.4. Frame Relay Service MIB Terminology

   Access Channel - An access channel generically refers to the DS1/E1
   or DS3/E3-based UNI access channel or NNI access channel across which
   frame relay data transits. An access channel is the access pathway
   for a single stream of user data.

   Within a given DS1 line, an access channel can denote any one of the
   following:

   o Unchannelized DS1 - the entire DS1 line is considered an access
      channel. Each access channel is comprised of 24 DS0 time slots.

   o Channelized DS1 - an access channel is any one of 24 channels.
      Each access channel is comprised of a single DS0 time slot.

   o Fractional DS1 - an access channel is a grouping of NxDS0 time
      slots (NX56/64 Kbps, where N = 1-23 DS0 Time slots per Fractional
      DS1 Access Channel) that may be assigned in consecutive or
      non-consecutive order.

   Within a given E1 line, a channel can denote any one of the following:

   o Unchannelized E1 - the entire E1 line is considered a single
      access channel. Each access channel is comprised of 31 E1 time
      slots.

   o Channelized E1 - an access channel is any one of 31 channels.
      Each access channel is comprised of a single E1 time slot.

   o Fractional E1 - an access channel is a grouping of N E1 time
      slots (NX64 Kbps, where N = 1-30 E1 time slots per FE1 access
      channel) that may be assigned in consecutive or non-consecutive
      order.

   Within a given unformatted line, the entire unformatted line is
   considered an access channel. Examples include RS-232, V.35, V.36 and
   X.21 (non-switched), and unframed E1 (G.703 without G.704).

   Access Rate - The data rate of the access channel, expressed in
   bits/second. The speed of the user access channel determines how
   rapidly the end user can inject data into the network.

   Bc - The Committed Burst Size (Bc) is the maximum amount of
   subscriber data (expressed in bits) that the network agrees to
   transfer, under normal conditions, during a time interval Tc.

Be - The Excess Burst Size (Be) is the maximum amount of subscriber
data (expressed in bits) in excess of Bc that the network will
attempt to deliver during the time interval Tc. This data (Be) is
delivered in general with a lower probability than Bc.

CIR - The Committed Information Rate (CIR) is the subscriber data
rate (expressed in bits/second) that the network commits to deliver
under normal network conditions. CIR is averaged over the time
interval Tc (CIR = Bc/Tc).

DLCI - Data Link Connection Identifier

Logical Port - This term is used to model the frame relay "interface"
on a device.

NNI - Network to Network Interface

Permanent Virtual Connection (PVC) - A virtual connection that has
its end-points and bearer capabilities defined at subscription time.

Time slot (E1) - An octet within the 256-bit information field in
each E1 frame is defined as a time slot. Time slots are position
sensitive within the 256-bit information field. Fractional E1 service
is provided in contiguous or non-contiguous time slot increments.

Time slot (DS0) - An octet within the 192-bit information field in
each DS1 frame is defined as a time slot. Time slots are position
sensitive within the 192-bit information field. Fractional DS1
service is provided in contiguous or non-contiguous time slot
increments.

UNI - User to Network Interface

N391 - Full status (status of all PVCs) polling counter

N392 - Error threshold

N393 - Monitored events count

T391 - Link integrity verification polling timer

T392 - Polling verification timer

nT3 - Status enquiry timer

nN3 - Maximum status enquiry counter

2.5. Relation to Other MIBs

2.5.1. System Group

   Use the System Group of the SNMPv2-MIB [27] to describe the Frame
   Relay Service (FRS) agent. The FRS agent may be monitoring many
   frame relay devices in one network. The System Group does not
   describe frame relay devices monitored by the FRS agent.

   sysDescr: ASCII string describing the FRS agent.
              Can be up to 255 characters long. This field is
              generally used to indicate the network providers
              identification and type of service offered.

   sysObjectID: Unique OBJECT IDENTIFIER (OID) for the
                 FRS agent.

   sysUpTime: Clock in the FRS agent; TimeTicks
               in 1/100s of a second. Elapsed type since
               the FRS agent came on line.

   sysContact: Contact for the FRS agent.
                ASCII string of up to 255 characters.

   sysName: Domain name of the FRS agent, for example,
             acme.com

   sysLocation: Location of the FRS agent.
                 ASCII string of up to 255 characters.

   sysServices: Services of the managed device. The value "2",
                 which implies that
                 the frame relay network is providing
                 a subnetwork level service, is recommended.

2.5.2. Interfaces Table (ifTable, ifXtable)

   This specifies how the Interfaces Group defined in the IF MIB [26]
   shall be used for the management of frame relay based interfaces, and
   in conjunction with the Frame Relay Service MIB module. This memo
   assumes the interpretation of the evolution of the Interfaces group
   to be in accordance with: "The interfaces table (ifTable) contains
   information on the managed resource's interfaces. Each sub-layer
   below the internetwork layer of a network interface is considered an
   interface." Thus, the ifTable allows the following frame relay-based
   interfaces to be represented as table entries:

-   Frame relay interfaces in equipment (e.g., switches, routers or
    networks) supporting frame relay. This level is concerned with
    generic frame counts and not with individual virtual connections.

In accordance with the guidelines of ifTable, frame counts per
virtual connection are not covered by ifTable, and are considered
interface specific and covered in the Frame Relay Service MIB module.
In order to interrelate the ifEntries properly, the Interfaces Stack
Group shall be supported.

Some specific interpretations of ifTable for frame relay follow.

```
Object               Use for the generic Frame Relay layer
======               ==========================================
ifIndex              Each frame relay port is represented by an
                     ifEntry.

ifDescr              Description of the frame relay interface.
                     ASCII string describing the UNI/NNI logical
                     port.  Can be up to 255 characters long.

ifType               The value allocated for Frame Relay Service
                     is equal to 44.

ifMtu                Set to maximum frame size in octets for this
                     frame relay logical port.

ifSpeed              Peak bandwidth in bits per second available
                     for use. This could be the speed of the
                     logical port and not the access rate. Actual
                     user information transfer rate (i.e., access
                     rate) of the UNI or NNI logical port in bits
                     per second (this is not the clocking speed).
                     For example, it is 1,536,000 bits per second
                     for a DS1-based UNI/NNI logical port and
                     1,984,000 bits per second for an E1-based
                     UNI/NNI logical port.

ifPhysAddress        The primary address for this logical port
                     assigned by the frame relay interface
                     provider.  An octet string of zero length if
                     no address is used for this logical port.

ifAdminStatus        The desired administrative status of the
                     frame relay logical port.
```

    ifOperStatus   The current operational status of the Frame
                          Relay UNI or NNI logical port.

    ifLastChange   The value of sysUptime at the last
                          re-initialization of the logical port. The
                          value of sysUpTime at the time the logical
                          port 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   The number of received octets. This counter
                          only counts octets from the beginning of the
                          frame relay header field to the end of user
                          data.

    ifInUcastPkts   The number of received unerrored, unicast
                          frames.

    ifInDiscards   The number of received frames discarded.
                          Specifically, frames discarded due to ingress
                          buffer congestion and traffic policing.

    ifInErrors   The number of received frames that are
                          discarded because of an error. Specifically,
                          frames that are too long or too short, frames
                          that are not a multiple of 8 bits in length,
                          frames with an invalid or unrecognized DLCI,
                          frames with an abort sequence, frames with
                          improper flag delimitation, and frame that
                          fail FCS.

    ifInUnknownProtos   The number of packets discarded because of an
                          unknown or unsupported protocol. For Frame
                          Relay Service interfaces, this counter will
                          always be zero.

    ifOutOctets   The number of transmitted octets. This
                          counter only counts octets from the beginning
                          of the frame relay header field to the end of
                          user data.

    ifOutUcastpkts   The number of unerrored, unicast frames sent.

    ifOutDiscards   The number of frames discarded in the egress
                          direction. Possible reasons are as follows:
                          policing, congestion.

   ifOutErrors   The number of frames discarded in the egress
                            direction because of an error. Specifically,
                            frames that are aborted due to a transmitter
                            underrun.

   ifName   This variable is not applicable for Frame
                            Relay Service interfaces, therefore, this
                            variable contains a zero-length string.

   ifInMulticastPkts   The number of received unerrored, multicast
                            frames.

   ifInBroadcastPkts   This variable is not applicable for Frame
                            Relay Service interfaces, therefore, this
                            counter is always zero.

   ifOutMulticastPkts   The number of sent unerrored, multicast
                            frames.

   ifOutBroadcastPkts   This variable is not applicable for Frame
                            Relay Service interfaces, therefore, this
                            counter is always zero.

   ifHCInOctets   Only used for DS3-based (and greater) Frame
                            Relay logical ports. The number of received
                            octets. This counter only counts octets
                            from the beginning of the frame relay header
                            field to the end of user data.

   ifHCOutOctets   Only used for DS3-based (and greater) Frame
                            Relay logical ports. The number of
                            transmitted octets. This counter only counts
                            octets from the beginning of the frame relay
                            header field to the end of user data.

   ifLinkUpDownTrapEnable Set to true(1). It is recommended that the
                            underlying physical layer notifications be
                            disabled since both are not required.
                            Notifications are enabled at the frame relay
                            service layer specifically because PVC
                            notifications are not to be sent if the frame
                            relay interface fails. Without a
                            linkUp/linkDown notification, the management
                            station would receive no notification of the
                            failure.

        ifHighSpeed   Set to the user data rate of the frame relay
                                logical port 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).

   Frame relay network service interfaces support the Interface Stack
   Group. Frame relay network service interfaces do not support any
   other groups or objects in the Interfaces group of the IF MIB.

2.5.3. Stack Table for DS1/E1 Environment

   This section describes by example how to use ifStackTable to
   represent the relationship of frame relay service to ds0 and
   ds0Bundles with ds1 interfaces [20].

   Example: A frame relay service is being carried on 4 ds0s of a ds1.
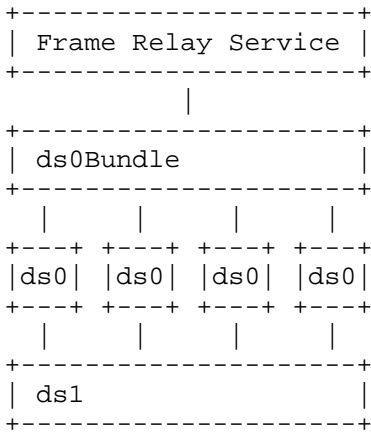
```
   +--------------------+
   | Frame Relay Service |
   +--------------------+
             |
   +--------------------+
   | ds0Bundle          |
   +--------------------+
     |     |     |     |
   +---+ +---+ +---+ +---+
   |ds0| |ds0| |ds0| |ds0|
   +---+ +---+ +---+ +---+
     |     |     |     |
   +--------------------+
   | ds1                |
   +--------------------+
```

   The assignment of the index values could for example be:

        ifIndex   Description
        1   FrameRelayService (type 44)
        2   ds0Bundle   (type 82)
        3   ds0 #1   (type 81)
        4   ds0 #2   (type 81)
        5   ds0 #3   (type 81)
        6   ds0 #4   (type 81)
        7   ds1   (type 18)

   The ifStackTable is then used to show the relationships between the
   various interfaces.

         ifStackTable Entries

         HigherLayer   LowerLayer
         0   1
         1   2
         2   3
         2   4
         2   5
         2   6
         3   7
         4   7
         5   7
         6   7
         7   0

   In the case where the frame relay service is using a single ds0, then
   the ds0Bundle is not required.
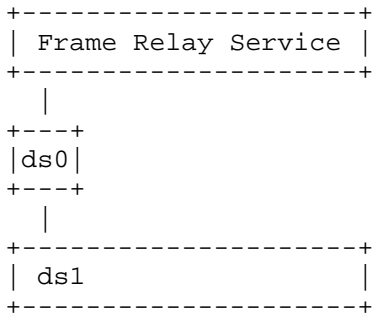
```
    +--------------------+
    | Frame Relay Service |
    +--------------------+
       |
    +---+
    |ds0|
    +---+
       |
    +--------------------+
    | ds1                |
    +--------------------+
```

   The assignment of the index values could for example be:

         ifIndex   Description
         1   FrameRelayService (type 44)
         2   ds0   (type 81)
         3   ds1   (type 18)

   The ifStackTable is then used to show the relationships between the
   various interfaces.

          ifStackTable Entries

          HigherLayer   LowerLayer
          0   1
          1   2
          2   3
          3   0

2.5.4.   Stack Table for V.35 Environments

     This section describes by example how to use ifStackTable to
     represent the relationship of frame relay service with V.35
     interfaces.

          +--------------------+
          | Frame Relay Service |
          +--------------------+
                    |
          +--------------------+
          | v35 |
          +--------------------+

     An example of index values in this case could be:

             ifIndex   Description
             1   FrameRelayService (type 44)
             2   v35   (type 33)

     Note type 33 (RS232-like MIB) is used instead of type 45 (V.35). V35
     does not pertain to this environment.

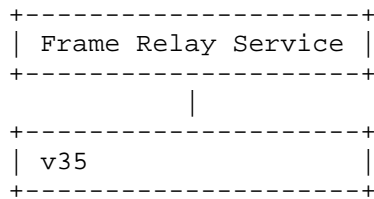   The ifStackTable is then used to show the relationships between the
     various interfaces.

             ifStackTable Entries

          HigherLayer   LowerLayer
          0   1
          1   2
          2   0

2.5.5.   The Frame Relay/ATM PVC Service Interworking MIB

     Connections between two frame relay endpoints are represented with an
     entry in the frPVCConnectTable of this MIB. Both endpoints are
     represented with rows in the frPVCEndptTable. The
     frPVCEndptConnectIdentifier object of each endpoint points to the
     frPVCConnectTable cross-connect table row for the connection.

   In contrast, a connection that spans frame relay and ATM endpoints is
   represented with an entry in the frAtmIwfConnectionTable of the
   FR/ATM PVC Service Interworking MIB defined in [28].

   In the case of an inter-worked connection, the
   frPVCEndptConnectIdentifier object is set to zero. Instead, the
   frPVCEndptAtmIwfConnIndex object is set to the index of the FR/ATM
   IWF cross-connect table row.

   The frame relay PVC cross-connect table (frPVCConnectTable) does not
   contain an entry for the FR/ATM inter-worked connection.

2.6. Textual Convention Change

   Version 1 of the Frame Relay Service MIB contains MIB objects defined
   with the DisplayString textual convention. In version 2 of this MIB,
   the syntax for these objects has been updated to use the (now
   preferred) SnmpAdminString textual convention. The new TC provides
   support for a greater variety of international character sets.

   The working group realizes that this change is not strictly supported
   by SMIv2. In our judgment, the alternative of deprecating the old
   objects and defining new objects would have a more adverse impact on
   backward compatibility and interoperability, given the particular
   semantics of these objects.

3. Object Definitions

FRNETSERV-MIB DEFINITIONS ::= BEGIN

IMPORTS
            MODULE-IDENTITY, OBJECT-TYPE,
            NOTIFICATION-TYPE, transmission,
            Counter32, Integer32   FROM SNMPv2-SMI
            TimeStamp, RowStatus   FROM SNMPv2-TC
            MODULE-COMPLIANCE, OBJECT-GROUP,
            NOTIFICATION-GROUP   FROM SNMPv2-CONF
            InterfaceIndex, ifIndex   FROM IF-MIB
            SnmpAdminString   FROM SNMP-FRAMEWORK-MIB;

        frnetservMIB MODULE-IDENTITY
            LAST-UPDATED "200009280000Z" -- September 28, 2000
            ORGANIZATION "IETF Frame Relay Service MIB Working Group"
            CONTACT-INFO
                "WG Charter:
                   http://www.ietf.org/html.charters/frnetmib-charter
                 WG-email:
                   frnetmib@sunroof.eng.sun.com

```
             Subscribe:
               frnetmib-request@sunroof.eng.sun.com
             Email Archive:
               ftp://ftp.ietf.org/ietf-mail-archive/frnetmib

             Chair:      Andy Malis
                         Vivace Networks, Inc.
             Email:      Andy.Malis@vivacenetworks.com

             WG editor:  Kenneth Rehbehn
                         Megisto Systems, Inc.
             Email:      krehbehn@megisto.com

             Co-author:  David Fowler
                         Syndesis Limited,
             EMail:      fowler@syndesis.com"

        DESCRIPTION
             "The MIB module to describe generic objects for
              Frame Relay Network Service."

        --
        -- Revision History
        --
        REVISION "200009280000Z"
        DESCRIPTION
             "Published as RFC 2954.

             The major new features of this revision include:

                 o  Support for read-write capability to
                    provision switch components providing service,

                 o  Support for cross-connection via a frame relay
                    to ATM service interworking function,

                 o  Support for frame relay fragmentation,

                 o  Additional frame counters to track frame
                    loss.

             Refer to Appendix A for a comprehensive list of
             changes since RFC 1604."

        REVISION "199311161200Z"
        DESCRIPTION
             "Published as RFC 1604."
        ::= { transmission 44 }
```

```
        frnetservObjects
            OBJECT IDENTIFIER ::= { frnetservMIB 1 }

        frnetservTraps
            OBJECT IDENTIFIER ::= { frnetservMIB 2 }

        frnetservTrapsPrefix
            OBJECT IDENTIFIER ::= { frnetservTraps 0 }

    --
    -- The Frame Relay Service Logical Port
    --
    frLportTable OBJECT-TYPE
        SYNTAX       SEQUENCE OF FrLportEntry
        MAX-ACCESS   not-accessible
        STATUS       current
        DESCRIPTION
                "The Frame Relay Logical Port Information table is
                 an interface-specific addendum to the generic
                 ifTable of the Interface MIB."
        ::= { frnetservObjects 1 }

    frLportEntry OBJECT-TYPE
        SYNTAX       FrLportEntry
        MAX-ACCESS   not-accessible
        STATUS       current
        DESCRIPTION
                "An entry in the Frame Relay Logical Port
                 Information table."
        INDEX    {   ifIndex }
        ::= { frLportTable 1 }

    FrLportEntry ::=
        SEQUENCE {
            frLportNumPlan            INTEGER,
            frLportContact            SnmpAdminString,
            frLportLocation           SnmpAdminString,
            frLportType               INTEGER,
            frLportAddrDLCILen        INTEGER,
            frLportVCSigProtocol      INTEGER,
            frLportVCSigPointer       OBJECT IDENTIFIER,
            frLportDLCIIndexValue     Integer32,
            frLportTypeAdmin          INTEGER,
            frLportVCSigProtocolAdmin INTEGER,
            frLportFragControl        INTEGER,
            frLportFragSize           Integer32
        }
```

```
        frLportNumPlan OBJECT-TYPE
            SYNTAX       INTEGER {
                         other(1),
                         e164(2),
                         x121(3),
                         none(4)
                         }
            MAX-ACCESS   read-only
            STATUS       current
            DESCRIPTION
                    "The value of this object identifies the network
                    address numbering plan for this UNI/NNI logical
                    port. The network address is the object
                    ifPhysAddress. The value none(4) implies that
                    there is no ifPhysAddress.  The FRS agent will
                    return an octet string of zero length for
                    ifPhysAddress.  The value other(1) means that an
                    address has been assigned to this interface, but
                    the numbering plan is not enumerated here."
            REFERENCE    "E.164 [29]
                          X.121 [30]"
            ::= { frLportEntry 1 }

        frLportContact  OBJECT-TYPE
            SYNTAX       SnmpAdminString
            MAX-ACCESS   read-only
            STATUS       current
            DESCRIPTION
                    "The value of this object identifies the network
                    contact for this UNI/NNI logical port."
            ::= { frLportEntry 2 }

        frLportLocation OBJECT-TYPE
            SYNTAX       SnmpAdminString
            MAX-ACCESS   read-only
            STATUS       current
            DESCRIPTION
                    "The value of this object identifies the frame
                    relay network location for this UNI/NNI logical
                    port."
            ::= { frLportEntry 3 }

        frLportType     OBJECT-TYPE
            SYNTAX       INTEGER {
                         uni(1),
                         nni(2)
                         }
            MAX-ACCESS   read-only
```

```
            STATUS       current
            DESCRIPTION
                    "The value of this object identifies the type of
                    network interface for this logical port."
            ::= { frLportEntry 4 }

        frLportAddrDLCILen OBJECT-TYPE
            SYNTAX       INTEGER {
                        twoOctets10Bits(1),
                        threeOctets10Bits(2),
                        threeOctets16Bits(3),
                        fourOctets17Bits(4),
                        fourOctets23Bits(5)
                        }
            UNITS        "Octets"
            MAX-ACCESS   read-only
            STATUS       current
            DESCRIPTION
                    "The value of this object identifies the Q.922
                    Address field length and DLCI length for this
                    UNI/NNI logical port."
            REFERENCE    "Q.922 [25]"
            ::= { frLportEntry 5 }

        frLportVCSigProtocol OBJECT-TYPE
            SYNTAX       INTEGER {
                        none(1),
                        lmi(2),
                        ansiT1617D(3),
                        ansiT1617B(4),
                        ccittQ933A(5)
                        }
            MAX-ACCESS   read-only
            STATUS       current
            DESCRIPTION
                    "The value of this object identifies the Local
                    In-Channel Signaling Protocol that is used for
                    this frame relay UNI/NNI logical port.

                    none(1):         Interface does not use a PVC
                                     signaling protocol

                    lmi(2):          Interface operates the Stratacom/
                                     Nortel/DEC Local Management
                                     Interface Specification protocol

                    ansiT1617D(3):   Interface operates the ANSI T1.617
                                     Annex D PVC status protocol
```

                       ansiT1617B(4): Interface operates the ANSI
T1.617
                                       Annex B procedures

                       ccittQ933A(5): Interface operates the ITU Q.933
                                       Annex A PVC status protocol"
            REFERENCE   "LMI [24]
                          T1.617 Annex D [17],
                          Q.933 Annex A [22]"
            ::= { frLportEntry 6 }

        frLportVCSigPointer OBJECT-TYPE
            SYNTAX   OBJECT IDENTIFIER
            MAX-ACCESS   read-only
            STATUS   deprecated
            DESCRIPTION
                    "The value of this object is used as a pointer to
                    the table that contains the Local In-Channel
                    Signaling Protocol parameters and errors for this
                    UNI/NNI logical port.

                    This object has been deprecated to reflect the
                    fact that the local in-channel signaling
                    parameters are accessed from a single table
                    (frMgtVCSigTable) that includes parameters for all
                    possible signaling protocols. Early design
                    anticipated multiple tables, one for each
                    signaling protocol."
            ::= { frLportEntry 7 }

        frLportDLCIIndexValue OBJECT-TYPE
            SYNTAX   Integer32 (16..4194303)
            MAX-ACCESS   read-only
            STATUS   current
            DESCRIPTION
                    "This object contains a hint to be used for
                    frPVCEndptDLCIIndex when creating entries in the
                    frPVCEndptTable. The SYNTAX of this object
                    matches the SYNTAX of the frPVCEndptDLCIIndex - an
                    object that is restricted to legal Q.922 DLCI
                    values for the size of the address field.

                    The value 0 indicates that no unassigned entries
                    are available.

                    To obtain the frPVCEndptDLCIIndex value for a new
                    entry, the manager issues a management protocol
                    retrieval operation to obtain the current value of

                    this object. After each retrieval, the agent must
                    modify the value to the next unassigned index to
                    prevent assignment of the same value to multiple
                    management systems.

                    A management system should repeat the read to
                    obtain a new value should an attempt to create the
                    new row using the previously returned hint fail."
          REFERENCE   "Q.922 [25]"
          ::= { frLportEntry 8 }

      frLportTypeAdmin   OBJECT-TYPE
          SYNTAX   INTEGER {
                       uni(1),
                       nni(2)
                       }
          MAX-ACCESS   read-write
          STATUS   current
          DESCRIPTION
                    "The value of this object desired identifies the
                    type of network interface for this logical port."
            ::= { frLportEntry 9 }

      frLportVCSigProtocolAdmin OBJECT-TYPE
          SYNTAX   INTEGER {
                       none(1),
                       lmi(2),
                       ansiT1617D(3),
                       ansiT1617B(4),
                       ccittQ933A(5)
                       }
          MAX-ACCESS   read-write
          STATUS   current
          DESCRIPTION
                    "The value of this object identifies the desired
                    Local In-Channel Signaling Protocol that is used
                    for this frame relay UNI/NNI logical port. This
                    value must be made the active protocol as soon as
                    possible on the device.

                    Refer to frLportVCSigProtocol for a description of
                    each signaling protocol choices."
          REFERENCE   "LMI [24]
                       T1.617 Annex D [17],
                       Q.933 Annex A [22]"
          ::= { frLportEntry 10 }

      frLportFragControl OBJECT-TYPE

```
         SYNTAX       INTEGER {
                      on(1),
                      off(2)
                      }
         MAX-ACCESS   read-write
         STATUS       current
         DESCRIPTION
                 "This object controls the transmission and
                 reception of fragmentation frames for this UNI or
                 NNI interface.

                 on(1)    Frames are fragmented using the interface
                          fragmentation format
                          Note:  The customer side of the interface
                          must also be configured to fragment
                          frames.

                 off(2)   Frames are not fragmented using the
                          interface fragmentation format."
         REFERENCE    "FRF.12 [21]"
         DEFVAL { off }
         ::= { frLportEntry 11 }

     frLportFragSize OBJECT-TYPE
         SYNTAX       Integer32 (0..4096)
         UNITS        "Octets"
         MAX-ACCESS   read-write
         STATUS       current
         DESCRIPTION
                 "The value of this object is the size in octets of
                 the maximum size of each fragment to be sent when
                 fragmenting.  This object is only used by the
                 fragmentation transmitter, and the two sides of
                 the interface may differ.  The fragment size
                 includes the octets for the frame relay header,
                 the UI octet, the NLPID, the fragmentation header,
                 and the fragment payload. If frLportFragControl is
                 set to off, this value should be zero."
         REFERENCE    "FRF.12 [21]"
         DEFVAL { 0 }
         ::= { frLportEntry 12 }

     --
     -- Frame Relay Management VC Signaling
     --

     frMgtVCSigTable OBJECT-TYPE
         SYNTAX       SEQUENCE OF FrMgtVCSigEntry
```

```
        MAX-ACCESS   not-accessible
        STATUS       current
        DESCRIPTION
                "The Frame Relay Management VC Signaling
                 Parameters and Errors table."
        ::= { frnetservObjects 2 }

    frMgtVCSigEntry OBJECT-TYPE
        SYNTAX       FrMgtVCSigEntry
        MAX-ACCESS   not-accessible
        STATUS       current
        DESCRIPTION
                "An entry in the Frame Relay Management VC
                 Signaling Parameters Errors table."
        INDEX       { ifIndex }
        ::= { frMgtVCSigTable 1 }

    FrMgtVCSigEntry ::=
        SEQUENCE {
            frMgtVCSigProced          INTEGER,
            frMgtVCSigUserN391        INTEGER,
            frMgtVCSigUserN392        INTEGER,
            frMgtVCSigUserN393        INTEGER,
            frMgtVCSigUserT391        INTEGER,
            frMgtVCSigNetN392         INTEGER,
            frMgtVCSigNetN393         INTEGER,
            frMgtVCSigNetT392         INTEGER,
            frMgtVCSigNetnN4          INTEGER,
            frMgtVCSigNetnT3          INTEGER,
            frMgtVCSigUserLinkRelErrors  Counter32,
            frMgtVCSigUserProtErrors     Counter32,
            frMgtVCSigUserChanInactive   Counter32,
            frMgtVCSigNetLinkRelErrors   Counter32,
            frMgtVCSigNetProtErrors      Counter32,
            frMgtVCSigNetChanInactive    Counter32,
            frMgtVCSigProcedAdmin     INTEGER,
            frMgtVCSigUserN391Admin   INTEGER,
            frMgtVCSigUserN392Admin   INTEGER,
            frMgtVCSigUserN393Admin   INTEGER,
            frMgtVCSigUserT391Admin   INTEGER,
            frMgtVCSigNetN392Admin    INTEGER,
            frMgtVCSigNetN393Admin    INTEGER,
            frMgtVCSigNetT392Admin    INTEGER,
            frMgtVCSigNetnT3Admin     INTEGER
        }

    frMgtVCSigProced OBJECT-TYPE
        SYNTAX       INTEGER {
```

```
                        u2nnet(1),
                        bidirect(2),
                        u2nuser(3)
                        }
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
                "The value of this object identifies the local
                in-channel signaling procedural role that is used
                for this UNI/NNI logical port.  Bidirectional
                procedures implies that both user-side and
                network-side procedural roles are used.

                u2nnet(1)    Logical port operates user to network
                             procedure in the role of the network
                             side

                bidirect(2) Logical port operates the
                             bidirectional procedure (both user
                             and network side roles)

                u2nuser(3)  Logical port operates user to network
                             procedure in the role of the user
                             side"
        REFERENCE   "Q.933 Annex A [22],
                    T1.617 Annex D [17]"
        ::= { frMgtVCSigEntry 1 }

    frMgtVCSigUserN391 OBJECT-TYPE
        SYNTAX      INTEGER (1..255)
        UNITS       "Polls"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
                "The value of this object identifies the User-side
                N391 full status polling cycle value for this
                UNI/NNI logical port.  If the logical port is not
                performing user-side (bidirectional) procedures,
                then this object is not instantiated and an
                attempt to read will result in the noSuchInstance
                exception response."
        REFERENCE   "Q.933 Annex A [22],
                    T1.617 Annex D [17]"
        DEFVAL { 6 }
        ::= { frMgtVCSigEntry 2 }

    frMgtVCSigUserN392 OBJECT-TYPE
        SYNTAX      INTEGER (1..10)
```

```
        UNITS        "Events"
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
                "The value of this object identifies the User-side
                N392 error threshold value for this UNI/NNI
                logical port.  If the logical port is not
                performing user-side (bidirectional) procedures,
                then this object is not instantiated."
        REFERENCE    "Q.933 Annex A [22],
                      T1.617 Annex D [17]"
        DEFVAL { 3 }
        ::= { frMgtVCSigEntry 3 }

    frMgtVCSigUserN393 OBJECT-TYPE
        SYNTAX       INTEGER (1..10)
        UNITS        "Events"
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
                "The value of this object identifies the User-side
                N393 monitored events count value for this UNI/NNI
                logical port.  If the logical port is not
                performing user-side (bidirectional) procedures,
                then this object is not instantiated."
        REFERENCE    "Q.933 Annex A [22],
                      T1.617 Annex D [17]"
        DEFVAL { 4 }
        ::= { frMgtVCSigEntry 4 }

    frMgtVCSigUserT391 OBJECT-TYPE
        SYNTAX       INTEGER (5..30)
        UNITS        "Seconds"
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
                "The value of this object identifies the User-side
                T391 link integrity verification polling timer
                value for this UNI/NNI logical port.  If the
                logical port is not performing user-side
                procedures, then this object is not instantiated."
        REFERENCE    "Q.933 Annex A [22],
                      T1.617 Annex D [17]"
        DEFVAL { 10 }
        ::= { frMgtVCSigEntry 5 }

    frMgtVCSigNetN392 OBJECT-TYPE
        SYNTAX       INTEGER (1..10)
```

```
        UNITS        "Events"
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
                "The value of this object identifies the Network-
                side N392 error threshold value (nN2 for LMI) for
                this UNI/NNI logical port.  If the logical port is
                not performing network-side procedures, then this
                object is not instantiated."
        REFERENCE    "Q.933 Annex A [22],
                      T1.617 Annex D [17],
                      LMI [24]"
        DEFVAL { 3 }
        ::= { frMgtVCSigEntry 6 }

    frMgtVCSigNetN393 OBJECT-TYPE
        SYNTAX       INTEGER (1..10)
        UNITS        "Events"
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
                "The value of this object identifies the Network-
                side N393 monitored events count value (nN3 for
                LMI) for this UNI/NNI logical port.  If the
                logical port is not performing network-side
                procedures, then this object is not instantiated."
        REFERENCE    "Q.933 Annex A [22],
                      T1.617 Annex D [17],
                      LMI [24]"
        DEFVAL { 4 }
        ::= { frMgtVCSigEntry 7 }

    frMgtVCSigNetT392 OBJECT-TYPE
        SYNTAX       INTEGER (5..30)
        UNITS        "Seconds"
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
                "The value of this object identifies the Network-
                side T392 polling verification timer value (nT2
                for LMI)  for this UNI/NNI logical port.  If the
                logical port is not performing network-side
                procedures, then this object is not instantiated."
        REFERENCE    "Q.933 Annex A [22],
                      T1.617 Annex D [17],
                      LMI [24]"
        DEFVAL { 15 }
        ::= { frMgtVCSigEntry 8 }
```

```
     frMgtVCSigNetnN4 OBJECT-TYPE
         SYNTAX       INTEGER (5..5)
         UNITS        "Events"
         MAX-ACCESS   read-only
         STATUS       current
         DESCRIPTION
                 "The value of this object identifies the Network-
                 side nN4 maximum status enquires received value
                 for this UNI/NNI logical port.  If the logical
                 port is not performing network-side procedures or
                 is not performing LMI procedures, then this object
                 is not instantiated.

                 This object applies only to LMI and always has a
                 value of 5."
         REFERENCE    "LMI [24]"
         ::= { frMgtVCSigEntry 9 }

     frMgtVCSigNetnT3 OBJECT-TYPE
         SYNTAX       INTEGER (5 | 10 | 15 | 20 | 25 | 30)
         UNITS        "Seconds"
         MAX-ACCESS   read-only
         STATUS       current
         DESCRIPTION
                 "The value of this object identifies the Network-
                 side nT3 timer (for nN4 status enquires received)
                 value for this UNI/NNI logical port.  If the
                 logical port is not performing network-side
                 procedures or is not performing LMI procedures,
                 then this object is not instantiated.

                   This object applies only to LMI."
          REFERENCE    "LMI [24]"
          DEFVAL { 20 }
          ::= { frMgtVCSigEntry 10 }

     frMgtVCSigUserLinkRelErrors OBJECT-TYPE
         SYNTAX       Counter32
         UNITS        "Errors"
         MAX-ACCESS   read-only
         STATUS       current
         DESCRIPTION
                 "The number of user-side local in-channel
                 signaling link reliability errors (i.e., non-
                 receipt of Status/Status Enquiry messages or
                 invalid sequence numbers in a Link Integrity
                 Verification Information Element) for this UNI/NNI
                 logical port.  If the logical port is not
```

                    performing user-side procedures, then this object
                    is not instantiated."
             ::= { frMgtVCSigEntry 11 }

        frMgtVCSigUserProtErrors OBJECT-TYPE
            SYNTAX   Counter32
            UNITS   "Errors"
            MAX-ACCESS   read-only
            STATUS   current
            DESCRIPTION
                    "The number of user-side local in-channel
                    signaling protocol errors (i.e., protocol
                    discriminator, unnumbered information, message
                    type, call reference, and mandatory information
                    element errors) for this UNI/NNI logical port. If
                    the logical port is not performing user-side
                    procedures, then this object is not instantiated."
             ::= { frMgtVCSigEntry 12 }

      frMgtVCSigUserChanInactive OBJECT-TYPE
          SYNTAX   Counter32
          UNITS   "Events"
          MAX-ACCESS   read-only
            STATUS   current
            DESCRIPTION
                    "The number of times the user-side channel was
                    declared inactive (i.e., N392 errors in N393
                    events) for this UNI/NNI logical port. If the
                    logical port is not performing user-side
                    procedures, then this object is not instantiated."
             ::= { frMgtVCSigEntry 13 }

        frMgtVCSigNetLinkRelErrors OBJECT-TYPE
            SYNTAX   Counter32
            UNITS   "Errors"
            MAX-ACCESS   read-only
            STATUS   current
            DESCRIPTION
                    "The number of network-side local in-channel
                    signaling link reliability errors (i.e., non-
                    receipt of Status/Status Enquiry messages or
                    invalid sequence numbers in a Link Integrity
                    Verification Information Element) for this UNI/NNI
                    logical port."
             ::= { frMgtVCSigEntry 14 }

        frMgtVCSigNetProtErrors OBJECT-TYPE
            SYNTAX   Counter32

```
        UNITS       "Errors"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
                "The number of network-side local in-channel
                signaling protocol errors (i.e., protocol
                discriminator, message type, call reference, and
                mandatory information element errors) for this
                UNI/NNI logical port."
        ::= { frMgtVCSigEntry 15 }


    frMgtVCSigNetChanInactive OBJECT-TYPE
        SYNTAX      Counter32
        UNITS       "Events"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
                "The number of times the network-side channel was
                declared inactive (i.e., N392 errors in N393
                events) for this UNI/NNI logical port."
        ::= { frMgtVCSigEntry 16 }


    frMgtVCSigProcedAdmin OBJECT-TYPE
        SYNTAX      INTEGER {
                    u2nnet(1),
                    bidirect(2),
                    u2nuser(3)
                    }
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
                "The value of this object identifies the local
                in-channel signaling procedural role that is used
                for this UNI/NNI logical port.  Bidirectional
                procedures implies that both user-side and
                network-side procedural roles are used.

                u2nnet(1)   Logical port operates user to network
                            procedure in the role of the network
                            side

                bidirect(2) Logical port operates the
                            bidirectional procedure (both user
                            and network side roles)

                u2nuser(3)  Logical port operates user to network
                            procedure in the role of the user
                            side"
```

```
        REFERENCE     "Q.933 Annex A [22],
                       T1.617 Annex D [17]"
        DEFVAL { u2nnet }
        ::= { frMgtVCSigEntry 17 }

    frMgtVCSigUserN391Admin OBJECT-TYPE
        SYNTAX        INTEGER (1..255)
        UNITS         "Polls"
        MAX-ACCESS    read-write
        STATUS        current
        DESCRIPTION
                "The value of this object identifies the desired
                User-side N391 full status polling cycle value for
                this UNI/NNI logical port.  If the logical port is
                not performing user-side (bidirectional)
                procedures, then this object is not instantiated."
        REFERENCE     "Q.933 Annex A [22],
                       T1.617 Annex D [17]"
        ::= { frMgtVCSigEntry 18 }

    frMgtVCSigUserN392Admin OBJECT-TYPE
        SYNTAX        INTEGER (1..10)
        UNITS         "Events"
        MAX-ACCESS    read-write
        STATUS        current
        DESCRIPTION
                "The value of this object identifies the desired
                User-side N392 error threshold value for this
                UNI/NNI logical port.  If the logical port is not
                performing user-side (bidirectional) procedures,
                then this object is not instantiated."
        REFERENCE     "Q.933 Annex A [22],
                       T1.617 Annex D [17]"
        ::= { frMgtVCSigEntry 19 }

    frMgtVCSigUserN393Admin OBJECT-TYPE
        SYNTAX        INTEGER (1..10)
        UNITS         "Events"
        MAX-ACCESS    read-write
        STATUS        current
        DESCRIPTION
                "The value of this object identifies the desired
                User-side N393 monitored events count value for
                this UNI/NNI logical port.  If the logical port is
                not performing user-side (bidirectional)
                procedures, then this object is not instantiated."
        REFERENCE     "Q.933 Annex A [22],
                       T1.617 Annex D [17]"
```

```
          ::= { frMgtVCSigEntry 20 }


     frMgtVCSigUserT391Admin OBJECT-TYPE
         SYNTAX        INTEGER (5..30)
         UNITS         "Seconds"
         MAX-ACCESS    read-write
         STATUS        current
         DESCRIPTION
                 "The value of this object identifies the desired
                 User-side T391 link integrity verification polling
                 timer value for this UNI/NNI logical port.  If the
                 logical port is not performing user-side
                 procedures, then this object is not instantiated."
         REFERENCE     "Q.933 Annex A [22],
                       T1.617 Annex D [17]"
         ::= { frMgtVCSigEntry 21 }


     frMgtVCSigNetN392Admin OBJECT-TYPE
         SYNTAX        INTEGER (1..10)
         UNITS         "Events"
         MAX-ACCESS    read-write
         STATUS        current
         DESCRIPTION
                 "The value of this object identifies the desired
                 Network-side N392 error threshold value (nN2 for
                 LMI) for this UNI/NNI logical port.  If the
                 logical port is not performing network-side
                 procedures, then this object is not instantiated."
         REFERENCE     "Q.933 Annex A [22],
                       T1.617 Annex D [17],
                       LMI [24]"
         ::= { frMgtVCSigEntry 22 }


     frMgtVCSigNetN393Admin OBJECT-TYPE
         SYNTAX        INTEGER (1..10)
         UNITS         "Events"
         MAX-ACCESS    read-write
         STATUS        current
         DESCRIPTION
                 "The value of this object identifies the desired
                 Network-side N393 monitored events count value
                 (nN3 for LMI) for this UNI/NNI logical port.  If
                 the logical port is not performing network-side
                 procedures, then this object is not instantiated."
         REFERENCE     "Q.933 Annex A [22],
                       T1.617 Annex D [17],
                       LMI [24]"
         ::= { frMgtVCSigEntry 23 }
```

```
        frMgtVCSigNetT392Admin OBJECT-TYPE
            SYNTAX       INTEGER (5..30)
            UNITS        "Seconds"
            MAX-ACCESS   read-write
            STATUS       current
            DESCRIPTION
                    "The value of this object identifies the desired
                    Network-side T392 polling verification timer value
                    (nT2 for LMI)  for this UNI/NNI logical port. If
                    the logical port is not performing network-side
                    procedures, then this object is not instantiated."
            REFERENCE    "Q.933 Annex A [22],
                         T1.617 Annex D [17],
                         LMI [24]"
            ::= { frMgtVCSigEntry 24 }

        frMgtVCSigNetnT3Admin OBJECT-TYPE
            SYNTAX       INTEGER (5 | 10 | 15 | 20 | 25 | 30)
            UNITS        "Seconds"
            MAX-ACCESS   read-write
            STATUS       current
            DESCRIPTION
                    "The value of this object identifies the desired
                    Network-side nT3 timer (for nN4 status enquires
                    received) value for this UNI/NNI logical port. If
                    the logical port is not performing network-side
                    procedures or is not performing LMI procedures,
                    then this object is not instantiated.  This object
                    applies only to LMI."
            REFERENCE    "LMI [24]"
            ::= { frMgtVCSigEntry 25 }

        --
        -- Frame Relay PVC End-points
        --

        frPVCEndptTable OBJECT-TYPE
            SYNTAX       SEQUENCE OF FrPVCEndptEntry
            MAX-ACCESS   not-accessible
            STATUS       current
            DESCRIPTION
                    "The Frame Relay PVC End-Point table.  This table
                    is used to model a PVC end-point.  This table
                    contains the traffic parameters and statistics for
                    a PVC end-point.

                    This table is used to identify the traffic
                    parameters for a bi-directional PVC segment end-
```

point, and it also provides statistics for a PVC
segment end-point.

A PVC segment end-point is identified by a UNI/NNI
logical port index value and DLCI index value.

If the frame relay service provider allows the
frame relay CNM subscriber to create, modify or
delete PVCs using SNMP, then this table is used to
identify and reserve the requested traffic
parameters of each PVC segment end-point. The
Connection table is used to 'connect' the end-
points together. Not all implementations will
support the capability of
creating/modifying/deleting PVCs using SNMP as a
feature of frame relay CNM service.

Uni-directional PVCs are modeled with zero valued
traffic parameters in one of the directions (In or
 Out direction) in this table.

To create a PVC, the following procedures shall be
followed:

1)   Create the entries for the PVC segment
     endpoints in the frPVCEndptTable by specifying
     the traffic parameters for the bi-directional
     PVC segment endpoints. As shown in figure 2, a
     point-to-point PVC has two endpoints, thus two
     entries in this table. Uni-directional PVCs
     are modeled with zero valued traffic
     parameters in one direction; all the 'In'
     direction parameters for one frame relay PVC
     End-point or all the 'Out' direction
     parameters for the other frame relay PVC
     Endpoint.

```
    In    _____  Out
   >>>>>>|                              |>>>>>>>>
   _____|       Frame Relay Network    |_____
    Out  |                              | In
   <<<<<<|_____|<<<<<<<<
   Frame Relay                        Frame Relay
      PVC                                PVC
   Endpoint                           Endpoint
```

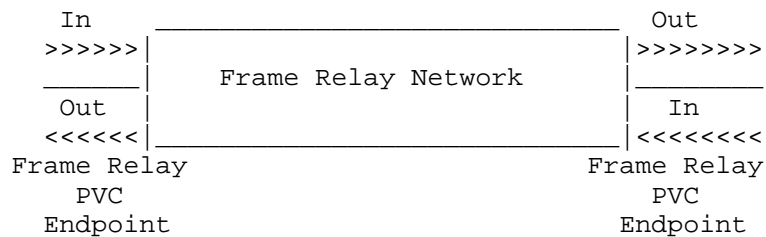   Figure 2, PVC Terminology

```
              2)  Go to the Frame Relay Connection Group."
        ::= { frnetservObjects 3 }

    frPVCEndptEntry OBJECT-TYPE
        SYNTAX        FrPVCEndptEntry
        MAX-ACCESS    not-accessible
        STATUS        current
        DESCRIPTION
                "An entry in the Frame Relay PVC Endpoint table."
        INDEX   { ifIndex, frPVCEndptDLCIIndex }
        ::= { frPVCEndptTable 1 }

    FrPVCEndptEntry ::=
        SEQUENCE {
            frPVCEndptDLCIIndex           Integer32,
            frPVCEndptInMaxFrameSize      Integer32,
            frPVCEndptInBc                Integer32,
            frPVCEndptInBe                Integer32,
            frPVCEndptInCIR               Integer32,
            frPVCEndptOutMaxFrameSize     Integer32,
            frPVCEndptOutBc               Integer32,
            frPVCEndptOutBe               Integer32,
            frPVCEndptOutCIR              Integer32,
            frPVCEndptConnectIdentifier   Integer32,
            frPVCEndptRowStatus           RowStatus,
            frPVCEndptRcvdSigStatus       INTEGER,
            frPVCEndptInFrames            Counter32,
            frPVCEndptOutFrames           Counter32,
            frPVCEndptInDEFrames          Counter32,
            frPVCEndptInExcessFrames      Counter32,
            frPVCEndptOutExcessFrames     Counter32,
            frPVCEndptInDiscards          Counter32,
            frPVCEndptInOctets            Counter32,
            frPVCEndptOutOctets           Counter32,
            frPVCEndptInDiscardsDESet     Counter32,
            frPVCEndptInFramesFECNSet     Counter32,
            frPVCEndptOutFramesFECNSet    Counter32,
            frPVCEndptInFramesBECNSet     Counter32,
            frPVCEndptOutFramesBECNSet    Counter32,
            frPVCEndptInCongDiscards      Counter32,
            frPVCEndptInDECongDiscards    Counter32,
            frPVCEndptOutCongDiscards     Counter32,
            frPVCEndptOutDECongDiscards   Counter32,
            frPVCEndptOutDEFrames         Counter32,
            frPVCEndptAtmIwfConnIndex     Integer32
            }
```

```
    frPVCEndptDLCIIndex OBJECT-TYPE
        SYNTAX        Integer32 (16..4194303)
        MAX-ACCESS    not-accessible
        STATUS        current
        DESCRIPTION
                "The value of this object is equal to the DLCI
                value for this PVC end-point.

                The values are restricted to the legal range for
                the size of address field supported by the logical
                port (frLportAddrDLCILen)."
        REFERENCE   "Q.922 [25]"
        ::= { frPVCEndptEntry 1 }

    frPVCEndptInMaxFrameSize OBJECT-TYPE
        SYNTAX        Integer32 (1..4096)
        UNITS         "Octets"
        MAX-ACCESS    read-create
        STATUS        current
        DESCRIPTION
                "The value of this object is the size in octets of
                the largest frame relay information field for this
                PVC end-point in the ingress direction (into the
                frame relay network).  The value of
                frPVCEndptInMaxFrameSize must be less than or
                equal to the corresponding ifMtu for this frame
                relay UNI/NNI logical port."
        REFERENCE   "FRF.1 [31]
                     Q.922 [25]
                     Q.933 [22]"
        DEFVAL { 1600 }
        ::= { frPVCEndptEntry 2 }

    frPVCEndptInBc OBJECT-TYPE
        SYNTAX        Integer32 (1..2147483647)
        UNITS         "Bits"
        MAX-ACCESS    read-create
        STATUS        current
        DESCRIPTION
                "The value of this object is equal to the
                committed burst size (Bc) parameter (measured in
                bits) for this PVC end-point in the ingress
                direction (into the frame relay network).

                Note that the max value of this range is lower
                than the max value allowed by Q.933 (16383 *
                10**6).
```

                    Note that the value is encoded in bits whilst the
                    Q.933 Link layer core parameters information
                    element encodes this information using octet
                    units."
            REFERENCE   "Q.933 [22]"
            ::= { frPVCEndptEntry 3 }

        frPVCEndptInBe OBJECT-TYPE
            SYNTAX   Integer32 (1..2147483647)
            UNITS   "Bits"
            MAX-ACCESS   read-create
            STATUS   current
            DESCRIPTION
                    "The value of this object is equal to the excess
                    burst size (Be) parameter (measured in bits) for
                    this PVC end-point in the ingress direction (into
                    the frame relay network).

                    Note that the max value of this range is lower
                    than the max value allowed by Q.933 (16383 *
                    10**6).

                    Note that the value is encoded in bits whilst the
                    Q.933 Link layer core parameters information
                    element encodes this information using octet
                    units."
            REFERENCE   "Q.933 [22]"
            ::= { frPVCEndptEntry 4 }

        frPVCEndptInCIR OBJECT-TYPE
            SYNTAX   Integer32 (1..2147483647)
            UNITS   "Bits per Second"
            MAX-ACCESS   read-create
            STATUS   current
            DESCRIPTION
                    "The value of this object is equal to the
                    committed information rate (CIR) parameter
                    (measured in bits per second) for this PVC end-
                    point in the ingress direction (into the frame
                    relay network).

                    Note that the max value of this range is lower
                    than the max value allowed by Q.933 (2047 *
                    10**6)."
            REFERENCE   "Q.933 [22]"
            ::= { frPVCEndptEntry 5 }

        frPVCEndptOutMaxFrameSize OBJECT-TYPE

```
        SYNTAX        Integer32 (1..4096)
        UNITS         "Octets"
        MAX-ACCESS    read-create
        STATUS        current
        DESCRIPTION
                "The value of this object is the size in octets of
                the largest frame relay information field for this
                PVC end-point in the egress direction (out of the
                frame relay network).  The value of
                frPVCEndptOutMaxFrameSize must be less than or
                equal to the corresponding ifMtu for this frame
                relay UNI/NNI logical port."
        REFERENCE    "FRF.1 [31]
                      Q.922 [25]
                      Q.933 [22]"
        DEFVAL { 1600 }
        ::= { frPVCEndptEntry 6 }

    frPVCEndptOutBc OBJECT-TYPE
        SYNTAX        Integer32 (1..2147483647)
        UNITS         "Bits"
        MAX-ACCESS    read-create
        STATUS        current
        DESCRIPTION
                "The value of this object is equal to the
                committed burst size (Bc) parameter (measured in
                bits) for this PVC end-point in the egress
                direction (out of the frame relay network).

                Note that the max value of this range is lower
                than the max value allowed by Q.933 (16383 *
                10**6).

                Note that the value is encoded in bits whilst the
                Q.933 Link layer core parameters information
                element encodes this information using octet
                units."
        REFERENCE    "Q.933 [22]"
        ::= { frPVCEndptEntry 7 }

    frPVCEndptOutBe OBJECT-TYPE
        SYNTAX        Integer32 (1..2147483647)
        UNITS         "Bits"
        MAX-ACCESS    read-create
        STATUS        current
        DESCRIPTION
                "The value of this object is equal to the excess
                burst size (Be) parameter (measured in bits) for
```

                        this PVC end-point in the egress direction (out of
                        the frame relay network).

                        Note that the max value of this range is lower
                        than the max value allowed by Q.933 (16383 *
                        10**6).

                        Note that the value is encoded in bits whilst the
                        Q.933 Link layer core parameters information
                        element encodes this information using octet
                        units."
                REFERENCE   "Q.933 [22]"
                ::= { frPVCEndptEntry 8 }

        frPVCEndptOutCIR OBJECT-TYPE
                SYNTAX   Integer32 (1..2147483647)
                UNITS   "Bits per Second"
                MAX-ACCESS   read-create
                STATUS   current
                DESCRIPTION
                        "The value of this object is equal to the
                        committed information rate (CIR) parameter
                        (measured in bits per second) for this PVC end-
                        point in the egress direction (out of the frame
                        relay network).

                        Note that the max value of this range is lower
                        than the max value allowed by Q.933 (2047 *
                        10**6)."
                REFERENCE   "Q.933 [22]"
                ::= { frPVCEndptEntry 9 }

        frPVCEndptConnectIdentifier OBJECT-TYPE
                SYNTAX   Integer32 (0..2147483647)
                MAX-ACCESS   read-only
                STATUS   current
                DESCRIPTION
                        "This object is used to associate PVC end-points
                        as being part of one PVC segment connection. This
                        value of this object is equal to the value of
                        frPVCConnectIndex, which is used as one of the
                        indices into the frPVCConnectTable.

                        A connection that has been cross-connected via the
                        FR/ATM PVC Service IWF cross-connect table will
                        return the value zero when this object is read. In
                        case of these interworked connections, the
                        frPVCEndptAtmIwfConnIndex object must be accessed

                    to select the entry in the FR/ATM PVC Service IWF
                    cross-connect table.

                    The value of this object is provided by the agent,
                    after the associated entries in the
                    frPVCConnectTable or frAtmIwfConnectionTable have
                    been created."
          ::= { frPVCEndptEntry 10 }

     frPVCEndptRowStatus OBJECT-TYPE
         SYNTAX   RowStatus
         MAX-ACCESS read-create
         STATUS   current
         DESCRIPTION
                    "This object is used to create new rows in this
                    table, modify existing rows, and to delete
                    existing rows. To create a new PVC, the entries
                    for the PVC segment end-points in the
                    frPVCEndptTable must first be created. Next, the
                    frPVCConnectTable is used to associate the frame
                    relay PVC segment end-points. In order for the
                    manager to have the necessary error diagnostics,
                    the frPVCEndptRowStatus object must initially be
                    set to 'createAndWait(5)'. While the
                    frPVCEndptRowStatus object is in the
                    'createAndWait(5)' state, the manager can set each
                    columnar object and get the necessary error
                    diagnostics. The frPVCEndptRowStatus object may
                    not be set to 'active(1)' unless the following
                    columnar objects exist in this row:
                    frPVCEndptInMaxFrameSize, frPVCEndptInBc,
                    frPVCEndptInBe, frPVCEndptInCIR,
                    frPVCEndptOutMaxFrameSize, frPVCEndptOutBc,
                    frPVCEndptOutBe, and frPVCEndptOutCIR."
          ::= { frPVCEndptEntry 11 }

     frPVCEndptRcvdSigStatus OBJECT-TYPE
         SYNTAX   INTEGER {
                    deleted(1),
                    active(2),
                    inactive(3),
                    none(4)
                    }
         MAX-ACCESS read-only
         STATUS   current
         DESCRIPTION
                    "The value of this object identifies the PVC
                    status received via the local in-channel signaling

procedures for this PVC end-point. This object is
only pertinent for interfaces that perform the
bidirectional procedures.

Each value has the following meaning:
  deleted(1):   This PVC is not listed in the full
                status reports received from the
                user device. The object retains
                this value for as long as the PVC
                is not listed in the full status
                reports

  active(2):   This PVC is reported as active, or
                operational, by the user device.

  inactive(3): This PVC is reported as inactive,
               or non-operational, by the user
               device.

  none(4):   This interface is only using the
                network-side in-channel signaling
                procedures, so this object does
                not apply."
    ::= { frPVCEndptEntry 12 }

frPVCEndptInFrames OBJECT-TYPE
    SYNTAX   Counter32
    UNITS   "Frames"
    MAX-ACCESS read-only
    STATUS   current
    DESCRIPTION
            "The number of frames received by the network
            (ingress) for this PVC end-point. This includes
            any frames discarded by the network due to
            submitting more than Bc + Be data or due to any
            network congestion recovery procedures."
    ::= { frPVCEndptEntry 13 }

frPVCEndptOutFrames OBJECT-TYPE
    SYNTAX   Counter32
    UNITS   "Frames"
    MAX-ACCESS read-only
    STATUS   current
    DESCRIPTION
            "The number of frames sent by the network (egress)
            regardless of whether they are Bc or Be frames for
            this PVC end-point."
    ::= { frPVCEndptEntry 14 }

```
        frPVCEndptInDEFrames OBJECT-TYPE
            SYNTAX          Counter32
            UNITS           "Frames"
            MAX-ACCESS      read-only
            STATUS          current
            DESCRIPTION
                    "The number of frames received by the network
                    (ingress) with the DE bit set to (1) for this PVC
                    end-point."
            ::= { frPVCEndptEntry 15 }

        frPVCEndptInExcessFrames OBJECT-TYPE
            SYNTAX          Counter32
            UNITS           "Frames"
            MAX-ACCESS      read-only
            STATUS          current
            DESCRIPTION
                    "The number of frames received by the network
                    (ingress) for this PVC end-point which were
                    treated as excess traffic.  Frames which are sent
                    to the network with DE set to zero are treated as
                    excess when more than Bc bits are submitted to the
                    network during the Committed Information Rate
                    Measurement Interval (Tc).  Excess traffic may or
                    may not be discarded at the ingress if more than
                    Bc + Be bits are submitted to the network during
                    Tc.  Traffic discarded at the ingress is not
                    recorded in frPVCEndptInExcessFrames.  Frames
                    which are sent to the network with DE set to one
                    are also treated as excess traffic."
            ::= { frPVCEndptEntry 16 }

        frPVCEndptOutExcessFrames OBJECT-TYPE
            SYNTAX          Counter32
            UNITS           "Frames"
            MAX-ACCESS      read-only
            STATUS          current
            DESCRIPTION
                    "The number of frames sent by the network (egress)
                    for this PVC end-point which were treated as
                    excess traffic.  (The DE bit may be set to one.)"
            ::= { frPVCEndptEntry 17 }

        frPVCEndptInDiscards OBJECT-TYPE
            SYNTAX          Counter32
            UNITS           "Frames"
            MAX-ACCESS      read-only
            STATUS          current
```

```
        DESCRIPTION
                "The number of frames received by the network
                (ingress) that were discarded due to traffic
                enforcement for this PVC end-point.  Congestion
                discards are not counted in this object."
        ::= { frPVCEndptEntry 18 }

    frPVCEndptInOctets OBJECT-TYPE
        SYNTAX      Counter32
        UNITS       "Octets"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
                "The number of octets received by the network
                (ingress) for this PVC end-point.  This counter
                should only count octets from the beginning of the
                frame relay header field to the end of user data.
                If the network supporting frame relay can not
                count octets, then this count should be an
                approximation."
        ::= { frPVCEndptEntry 19 }

    frPVCEndptOutOctets OBJECT-TYPE
        SYNTAX      Counter32
        UNITS       "Octets"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
                "The number of octets sent by the network (egress)
                for this PVC end-point.  This counter should only
                count octets from the beginning of the frame relay
                header field to the end of user data.  If the
                network supporting frame relay can not count
                octets, then this count should be an
                approximation."
        ::= { frPVCEndptEntry 20 }

    frPVCEndptInDiscardsDESet OBJECT-TYPE
        SYNTAX      Counter32
        UNITS       "Frames"
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
                "The number of frames received by the network
                (ingress) that were discarded with the DE bit set
                due to traffic enforcement for this PVC end-point.
                Congestion discards are not counted in this
                object."
```

```
        ::= { frPVCEndptEntry 21 }


    frPVCEndptInFramesFECNSet OBJECT-TYPE
        SYNTAX        Counter32
        UNITS         "Frames"
        MAX-ACCESS    read-only
        STATUS        current
        DESCRIPTION
                "The number of frames received by the network
                (ingress) that have the FECN bit set for this PVC
                end-point."
        ::= { frPVCEndptEntry 22 }


    frPVCEndptOutFramesFECNSet OBJECT-TYPE
        SYNTAX        Counter32
        UNITS         "Frames"
        MAX-ACCESS    read-only
        STATUS        current
        DESCRIPTION
                "The number of frames sent by the network (egress)
                that have the FECN bit set for this PVC end-
                point."
        ::= { frPVCEndptEntry 23 }


    frPVCEndptInFramesBECNSet OBJECT-TYPE
        SYNTAX        Counter32
        UNITS         "Frames"
        MAX-ACCESS    read-only
        STATUS        current
        DESCRIPTION
                "The number of frames received by the network
                (ingress) that have the BECN bit set for this PVC
                end-point."
        ::= { frPVCEndptEntry 24 }


    frPVCEndptOutFramesBECNSet OBJECT-TYPE
        SYNTAX        Counter32
        UNITS         "Frames"
        MAX-ACCESS    read-only
        STATUS        current
        DESCRIPTION
                "The number of frames sent by the network (egress)
                that have the BECN bit set for this PVC end-
                point."
        ::= { frPVCEndptEntry 25 }


    frPVCEndptInCongDiscards OBJECT-TYPE
        SYNTAX        Counter32
```

```
        UNITS        "Frames"
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
               "The number of frames received by the network
               (ingress) that were discarded due to input buffer
               congestion, rather than traffic enforcement, for
               this PVC end-point."
        ::= { frPVCEndptEntry 26 }

    frPVCEndptInDECongDiscards OBJECT-TYPE
        SYNTAX       Counter32
        UNITS        "Frames"
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
               "The number of frames counted by
               frPVCEndptInCongDiscards with the DE bit set to
               (1)."
        ::= { frPVCEndptEntry 27 }

    frPVCEndptOutCongDiscards OBJECT-TYPE
        SYNTAX       Counter32
        UNITS        "Frames"
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
               "The number of frames sent by the network (egress)
               that were discarded due to output buffer
               congestion for this PVC end-point."
        ::= { frPVCEndptEntry 28 }

    frPVCEndptOutDECongDiscards OBJECT-TYPE
        SYNTAX       Counter32
        UNITS        "Frames"
        MAX-ACCESS   read-only
        STATUS       current
        DESCRIPTION
               "The number of frames counted by
               frPVCEndptOutCongDiscards with the DE bit set to
               (1)."
        ::= { frPVCEndptEntry 29 }

    frPVCEndptOutDEFrames OBJECT-TYPE
        SYNTAX       Counter32
        UNITS        "Frames"
        MAX-ACCESS   read-only
        STATUS       current
```

```
        DESCRIPTION
                "The number of frames sent by the network (egress)
                with the DE bit set to (1) for this PVC end-
                point."
        ::= { frPVCEndptEntry 30 }

    frPVCEndptAtmIwfConnIndex OBJECT-TYPE
        SYNTAX       Integer32  (0..2147483647)
        MAX-ACCESS  read-only
        STATUS       current
        DESCRIPTION
                "This object contains the index value of the
                FR/ATM cross-connect table entry used to link the
                frame relay PVC with an ATM PVC.

                Each row of the frPVCEndptTable that is not
                cross-connected with an ATM PVC must return the
                value zero when this object is read.

                The value of this object is initialized by the
                agent after the associated entries in the
                frAtmIwfConnectionTable have been created.

                The value of this object is reset to zero
                following destruction of the associated entry in
                the frAtmIwfConnectionTable"
        ::= { frPVCEndptEntry 31 }

    --
    -- Frame Relay PVC Connections
    --

    frPVCConnectIndexValue OBJECT-TYPE
        SYNTAX       INTEGER (0..2147483647)
        MAX-ACCESS  read-only
        STATUS       current
        DESCRIPTION
                "This object returns a hint to be used for
                frPVCConnectIndex when creating entries in the
                frPVCConnectTable.

                The value 0 indicates that no unassigned entries
                are available.

                To obtain the frPVCConnectIndex value for a new
                entry, the manager issues a management protocol
                retrieval operation to obtain the current value of
                this object.  After each retrieval, the agent must
```

                      modify the value to the next unassigned index to
                      prevent assignment of the same value to multiple
                      management systems.

                      A management system should repeat the read to
                      obtain a new value should an attempt to create the
                      new row using the previously returned hint fail."
              ::= { frnetservObjects 4 }

      frPVCConnectTable OBJECT-TYPE
          SYNTAX   SEQUENCE OF FrPVCConnectEntry
          MAX-ACCESS   not-accessible
          STATUS   current
          DESCRIPTION
                      "The Frame Relay PVC Connect Table is used to
                      model the bi-directional PVC segment flows
                      including: point-to-point PVCs, point-to-
                      multipoint PVCs, and multipoint-to-multipoint
                      PVCs.

                      This table has read-create access and is used to
                      associate PVC end-points together as belonging to
                      one connection. The frPVCConnectIndex is used to
                      associate all the bi-directional flows. Not all
                      implementations will support the capability of
                      creating/modifying/deleting PVCs using SNMP as a
                      feature of frame relay CNM service.

                      Once the entries in the frPVCEndptTable are
                      created, the following step are used to associate
                      the PVC end-points as belonging to one PVC
                      connection:

                      1)   Obtain a unique frPVCConnectIndex
                           using the frPVCConnectIndexValue object.

                      2)   Connect the PVC segment endpoints together
                           with the applicable frPVCConnectIndex value
                           obtained via frPVCConnectIndexValue. The
                           entries in this table are created by using
                           the frPVCConnectRowStatus object.

                      3)   The agent will provide the value of the
                           corresponding instances of
                           frPVCEndptConnectIdentifier with the
                           frPVCConnectIndex value.

                      4)   Set frPVCConnectAdminStatus to 'active(1)' in

                        all rows for this PVC segment to turn the
                        PVC on.

                For example, the Frame Relay PVC Connection Group
                models a bi-directional, point-to-point PVC
                segment as one entry in this table.

                Frame Relay   Frame Relay
                 Network   Network
                 Low Port   High Port
                         _____
                        | |
                 _____ | >> from low to high PVC flow >> |_____
                        | << from high to low PVC flow << |
                        |_____|

                The terms low and high are chosen to represent
                numerical ordering of a PVC segment's endpoints
                for representation in this table. That is, the
                endpoint with the lower value of ifIndex is termed
                'low', while the opposite endpoint of the segment
                is termed 'high'. This terminology is to provide
                directional information; for example the
                frPVCConnectL2hOperStatus and
                frPVCConnectH2lOperStatus as illustrated above.

                If the Frame Relay Connection table is used to
                model a unidirectional PVC, then one direction
                (either from low to high or from high to low) has
                 its Operational Status equal to down.

                 A PVC segment is a portion of a PVC that traverses
                one Frame Relay Network, and a PVC segment is
                identified by its two end-points (UNI/NNI logical
                port index value and DLCI index value) through one
                Frame Relay Network."
        ::= { frnetservObjects 5 }

      frPVCConnectEntry OBJECT-TYPE
          SYNTAX   FrPVCConnectEntry
          MAX-ACCESS not-accessible
          STATUS   current
          DESCRIPTION
                "An entry in the Frame Relay PVC Connect table.
                This entry is used to model a PVC segment in two
                directions."
          INDEX { frPVCConnectIndex,
                frPVCConnectLowIfIndex,

```
                    frPVCConnectLowDLCIIndex,
                    frPVCConnectHighIfIndex,
                    frPVCConnectHighDLCIIndex }
            ::= { frPVCConnectTable 1 }

        FrPVCConnectEntry ::=
            SEQUENCE {
                frPVCConnectIndex            Integer32,
                frPVCConnectLowIfIndex       InterfaceIndex,
                frPVCConnectLowDLCIIndex     Integer32,
                frPVCConnectHighIfIndex      InterfaceIndex,
                frPVCConnectHighDLCIIndex    Integer32,
                frPVCConnectAdminStatus      INTEGER,
                frPVCConnectL2hOperStatus    INTEGER,
                frPVCConnectH2lOperStatus    INTEGER,
                frPVCConnectL2hLastChange    TimeStamp,
                frPVCConnectH2lLastChange    TimeStamp,
                frPVCConnectRowStatus        RowStatus,
                frPVCConnectUserName         SnmpAdminString,
                frPVCConnectProviderName     SnmpAdminString
        }

          frPVCConnectIndex OBJECT-TYPE
              SYNTAX      Integer32 (0..2147483647)
              MAX-ACCESS  not-accessible
              STATUS      current
              DESCRIPTION
                      "The value of this object is equal to the
                       frPVCConnectIndexValue obtained to uniquely
                       identify this PVC segment connection."
              ::= { frPVCConnectEntry 1 }

        frPVCConnectLowIfIndex OBJECT-TYPE
            SYNTAX      InterfaceIndex
            MAX-ACCESS  not-accessible
            STATUS      current
            DESCRIPTION
                    "The value of this object is equal to IF-MIB
                     ifIndex value of the UNI/NNI logical port for this
                     PVC segment.  The term low implies that this PVC
                     segment end-point has the numerically lower
                     ifIndex value than the connected/associated PVC
                     segment end-point.

                     RFC 1604 permitted a zero value for this object to
                     identify termination at a non-frame relay
                     interface.  However, this cross-connect table is
                     limited to frame relay connections.  See the frame
```

                    relay/ATM IWF MIB [28] for the cross-connect table
                    used for those types of connections."
            ::= { frPVCConnectEntry 2 }

        frPVCConnectLowDLCIIndex OBJECT-TYPE
            SYNTAX   Integer32 (16..4194303)
            MAX-ACCESS   not-accessible
            STATUS   current
            DESCRIPTION
                    "The value of this object is equal to the DLCI
                    value for this end-point of the PVC segment."
            REFERENCE   "Q.922 [25]"
            ::= { frPVCConnectEntry 3 }

        frPVCConnectHighIfIndex OBJECT-TYPE
            SYNTAX   InterfaceIndex
            MAX-ACCESS   not-accessible
            STATUS   current
            DESCRIPTION
                    "The value of this object is equal to IF-MIB
                    ifIndex value for the UNI/NNI logical port for
                    this PVC segment. The term high implies that this
                    PVC segment end-point has the numerically higher
                    ifIndex value than the connected/associated PVC
                    segment end-point."
            ::= { frPVCConnectEntry 4 }

        frPVCConnectHighDLCIIndex OBJECT-TYPE
            SYNTAX   Integer32 (16..4194303)
            MAX-ACCESS   not-accessible
            STATUS   current
            DESCRIPTION
                    "The value of this object is equal to the egress
                    DLCI value for this end-point of the PVC segment."
            REFERENCE   "Q.922 [25]"
            ::= { frPVCConnectEntry 5 }

        frPVCConnectAdminStatus OBJECT-TYPE
            SYNTAX   INTEGER {
                    active(1),
                    inactive(2),
                    testing(3)
                    }
            MAX-ACCESS   read-create
            STATUS   current
            DESCRIPTION
                    "The value of this object identifies the desired
                    administrative status of this bi-directional PVC

                segment. The active(1) state means the PVC
                segment is currently operational; the inactive(2)
                state means the PVC segment is currently not
                operational; the testing(3) state means the PVC
                segment is currently undergoing a test. This
                state is set by an administrative entity. This
                value affects the PVC status indicated across the
                ingress NNI/UNI of both end-points of the bi-
                directional PVC segment. When a PVC segment
                connection is created using this table, this
                object is initially set to 'inactive(2)'. After
                the frPVCConnectRowStatus object is set to
                'active(1)' (and the corresponding/associated
                entries in the frPVCEndptTable have their
                frPVCEndptRowStatus object set to 'active(1)'),
                the frPVCConnectAdminStatus object may be set to
                'active(1)' to turn on the PVC segment
                connection."
        ::= { frPVCConnectEntry 6 }

    frPVCConnectL2hOperStatus OBJECT-TYPE
        SYNTAX   INTEGER {
                    active(1),
                    inactive(2),
                    testing(3),
                    unknown(4)
                    }
        MAX-ACCESS read-only
        STATUS   current
        DESCRIPTION
                "The value of this object identifies the current
                operational status of the PVC segment connection
                in one direction; (i.e., in the low to high
                direction). This value affects the PVC status
                indicated across the ingress NNI/UNI (low side) of
                the PVC segment.

                The values mean:

                active(1)   - PVC is currently operational

                inactive(2) - PVC is currently not operational.
                              This may be because of an underlying
                              LMI or DS1 failure.

                testing(3) - PVC is currently undergoing a test.
                              This may be because of an underlying
                              frLport or DS1 undergoing a test.

                     unknown(4) - the status of the PVC currently can
                                   not be determined."
            ::= { frPVCConnectEntry 7 }

        frPVCConnectH2lOperStatus OBJECT-TYPE
            SYNTAX   INTEGER {
                         active(1),
                         inactive(2),
                         testing(3),
                         unknown(4)
                         }
            MAX-ACCESS   read-only
            STATUS   current
            DESCRIPTION
                     "The value of this object identifies the current
                     operational status of the PVC segment connection
                     in one direction; (i.e., in the high to low
                     direction).. This value affects the PVC status
                     indicated across the ingress NNI/UNI (high side)
                     of the PVC segment.

                     The values mean:

                     active(1)   - PVC is currently operational

                     inactive(2) - PVC is currently not operational.
                                   This may be because of an underlying
                                   LMI or DS1 failure.

                     testing(3) - PVC is currently undergoing a test.
                                   This may be because of an underlying
                                   frLport or DS1 undergoing a test.

                     unknown(4) - the status of the PVC currently can
                                   not be determined."
            ::= { frPVCConnectEntry 8 }

        frPVCConnectL2hLastChange OBJECT-TYPE
            SYNTAX   TimeStamp
            MAX-ACCESS   read-only
            STATUS   current
            DESCRIPTION
                     "The value of the Interface MIB's sysUpTime object
                     at the time this PVC segment entered its current
                     operational state in the low to high direction.
                     If the current state was entered prior to the last
                     re-initialization of the FRS agent, then this
                     object contains a zero value."

```
        ::= { frPVCConnectEntry 9 }

frPVCConnectH2lLastChange OBJECT-TYPE
    SYNTAX        TimeStamp
    MAX-ACCESS  read-only
    STATUS        current
    DESCRIPTION
            "The value of the Interface MIB's sysUpTime object
            at the time this PVC segment entered its current
            operational state in the high to low direction.
            If the current state was entered prior to the last
            re-initialization of the FRS agent, then this
            object contains a zero value."
    ::= { frPVCConnectEntry 10 }

frPVCConnectRowStatus OBJECT-TYPE
    SYNTAX        RowStatus
    MAX-ACCESS  read-create
    STATUS        current
    DESCRIPTION
            "The status of this entry in the
            frPVCConnectTable.  This variable is used to
            create new connections for the PVC end-points and
            to change existing connections of the PVC end-
            points.  This object must be initially set to
            'createAndWait(5)'. In this state, the agent
            checks the parameters in the associated entries in
            the frPVCEndptTable to verify that the PVC end-
            points can be connected (i.e., the In parameters
            for one PVC end-point are equal to the Out
            parameters for the other PVC end-point).  This
            object can not be set to 'active(1)' unless the
            following columnar object exists in this row:
            frPVCConnectAdminStatus.  The agent also supplies
            the associated value of frPVCConnectIndex for the
            frPVCEndptConnectIdentifier instances.  To turn on
            a PVC segment connection, the
            frPVCConnectAdminStatus is set to 'active(1)'."
    ::= { frPVCConnectEntry 11 }

frPVCConnectUserName OBJECT-TYPE
    SYNTAX        SnmpAdminString
    MAX-ACCESS  read-create
    STATUS        current
    DESCRIPTION
            "This is a service user assigned textual
            representation of a PVC."
    ::= { frPVCConnectEntry 12 }
```

        frPVCConnectProviderName OBJECT-TYPE
            SYNTAX   SnmpAdminString
            MAX-ACCESS   read-create
            STATUS   current
            DESCRIPTION
                    "This is a system supplied textual representation
                    of PVC. It is assigned by the service provider."
            ::= { frPVCConnectEntry 13 }

        --
        -- The Frame Relay Accounting
        --

        frAccountPVCTable OBJECT-TYPE
            SYNTAX   SEQUENCE OF FrAccountPVCEntry
            MAX-ACCESS   not-accessible
            STATUS   current
            DESCRIPTION
                    "The Frame Relay Accounting PVC table. This table
                    is used to perform accounting on a PVC segment
                    end-point basis."
            ::= { frnetservObjects 6 }

        frAccountPVCEntry OBJECT-TYPE
            SYNTAX   FrAccountPVCEntry
            MAX-ACCESS   not-accessible
            STATUS   current
            DESCRIPTION
                    "An entry in the Frame Relay Accounting PVC
                    table."
            INDEX   {   ifIndex,
                        frAccountPVCDLCIIndex }
            ::= { frAccountPVCTable 1 }

        FrAccountPVCEntry ::=
            SEQUENCE {
                frAccountPVCDLCIIndex   Integer32,
                frAccountPVCSegmentSize   Integer32,
                frAccountPVCInSegments   Counter32,
                frAccountPVCOutSegments   Counter32
            }

        frAccountPVCDLCIIndex OBJECT-TYPE
            SYNTAX   Integer32 (16..4194303)
            MAX-ACCESS   not-accessible
            STATUS   current
            DESCRIPTION
                    "The value of this object is equal to the DLCI

                        value for this PVC segment end-point."
            REFERENCE   "Q.922 [25]"
            ::= { frAccountPVCEntry 1 }

        frAccountPVCSegmentSize OBJECT-TYPE
            SYNTAX   Integer32
            UNITS   "Octets"
            MAX-ACCESS   read-only
            STATUS   current
            DESCRIPTION
                    "The value of this object is equal to the Segment
                    Size for this PVC segment end-point."
            ::= { frAccountPVCEntry 2 }

        frAccountPVCInSegments OBJECT-TYPE
            SYNTAX   Counter32
            UNITS   "Segments"
            MAX-ACCESS   read-only
            STATUS   current
            DESCRIPTION
                    "The value of this object is equal to the number
                    of segments received by this PVC segment end-
                    point."
            ::= { frAccountPVCEntry 3 }

        frAccountPVCOutSegments OBJECT-TYPE
            SYNTAX   Counter32
            UNITS   "Segments"
            MAX-ACCESS   read-only
            STATUS   current
            DESCRIPTION
                    "The value of this object is equal to the number
                    of segments sent by this PVC segment end-point."
            ::= { frAccountPVCEntry 4 }

        --
        -- Accounting on a Frame Relay Logical Port
        --

        frAccountLportTable OBJECT-TYPE
            SYNTAX   SEQUENCE OF FrAccountLportEntry
            MAX-ACCESS   not-accessible
            STATUS   current
            DESCRIPTION
                    "The Frame Relay Accounting Logical Port table.
                    This table is used to perform accounting on a
                    UNI/NNI Logical Port basis."
            ::= { frnetservObjects 7 }

```
        frAccountLportEntry OBJECT-TYPE
            SYNTAX       FrAccountLportEntry
            MAX-ACCESS   not-accessible
            STATUS       current
            DESCRIPTION
                    "An entry in the Frame Relay Accounting Logical
                    Port table."
            INDEX   {   ifIndex }
            ::= { frAccountLportTable 1 }

        FrAccountLportEntry ::=
            SEQUENCE {
                frAccountLportSegmentSize
                    Integer32,
                frAccountLportInSegments
                    Counter32,
                frAccountLportOutSegments
                    Counter32
            }

        frAccountLportSegmentSize OBJECT-TYPE
            SYNTAX       Integer32
            UNITS        "Octets"
            MAX-ACCESS   read-only
            STATUS       current
            DESCRIPTION
                    "The value of this object is equal to the Segment
                    Size for this UNI/NNI logical port."
            ::= { frAccountLportEntry 1 }

        frAccountLportInSegments OBJECT-TYPE
            SYNTAX       Counter32
            UNITS        "Segments"
            MAX-ACCESS   read-only
            STATUS       current
            DESCRIPTION
                    "The value of this object is equal to the number
                    of segments received by this UNI/NNI logical
                    port."
            ::= { frAccountLportEntry 2 }

        frAccountLportOutSegments OBJECT-TYPE
            SYNTAX       Counter32
            UNITS        "Segments"
            MAX-ACCESS   read-only
            STATUS       current
            DESCRIPTION
                    "The value of this object is equal to the number
```

```
                   of segments sent by this UNI/NNI logical port."
           ::= { frAccountLportEntry 3 }


        --
        -- Frame Relay Network Service Notifications
        --


     frPVCConnectStatusChange  NOTIFICATION-TYPE
         OBJECTS { frPVCConnectIndex,
                   frPVCConnectLowIfIndex,
                   frPVCConnectLowDLCIIndex,
                   frPVCConnectHighIfIndex,
                   frPVCConnectHighDLCIIndex,
                   frPVCConnectL2hOperStatus,
                   frPVCConnectH2lOperStatus,
                   frPVCEndptRcvdSigStatus }
         STATUS  deprecated
         DESCRIPTION
                 "Refer to the description of the
                 frPVCConnectStatusNotif notification that has
                 replaced this notification.  The notification is
                 deprecated due to the incorrect inclusion of index
                 values and to take advantage of the trap prefix
                 for automatic conversion from SMIv2 to SMIv1 by
                 making the one but last sub-ID a zero (i.e. the
                 so-called trap prefix)."
         ::= { frnetservTraps 1 }

     frPVCConnectStatusNotif  NOTIFICATION-TYPE
         OBJECTS { frPVCConnectL2hOperStatus,
                   frPVCConnectH2lOperStatus,
                   frPVCEndptRcvdSigStatus }
         STATUS  current
         DESCRIPTION
                 "This notification indicates that the indicated
                 PVC has changed state.

                 This notification is not sent if an FR-UNI changes
                 state; a linkDown or linkUp notification should be
                 sent instead.  The first instance of
                 frPVCEndptRcvdSigStatus is for the endpoint with
                 LowIfIndex, LowDLCIIndex.  The second instance of
                 frPVCEndptRcvdSigStatus is for the endpoint with
                 HighIfIndex, HighDLCIIndex"
         ::= { frnetservTrapsPrefix 2 }

     -- Conformance Information
```

```
     frnetservConformance  OBJECT IDENTIFIER
         ::= { frnetservMIB 3 }


     frnetservGroups       OBJECT IDENTIFIER
         ::= { frnetservConformance 1 }
     frnetservCompliances  OBJECT IDENTIFIER
         ::= { frnetservConformance 2 }



     --
     --  Service (Read-only) Modules
     --
     frnetservCompliance2 MODULE-COMPLIANCE
         STATUS      current
         DESCRIPTION
                 "The compliance statement for SNMP entities which
                 have Frame Relay Network Service Interfaces.

                 The distinction between 'service' and 'switch' is
                 that a 'switch' is configured via this MIB.
                 Hence, the various read/write objects have write
                 capability.  A 'service' represents a passive
                 monitor-only customer network management
                 interface.  The various read/write objects are
                 restricted to read-only capability."
         MODULE -- this module
             MANDATORY-GROUPS { frnetservLportGroup2,
                                frnetservMgtVCSigGroup,
                                frnetservPVCEndptGroup,
                                frnetservPVCEndptGroup2,
                                frnetservPVCConnectGroup,
                                frnetservPVCConnectNamesGroup,
                                frnetservPVCNotifGroup2 }

             GROUP       frnetservAccountPVCGroup
             DESCRIPTION
                 "This group is optional for frame relay
                 interfaces.  It is mandatory if and only if
                 accounting is performed on a PVC basis this frame
                 relay interface."
             GROUP       frnetservAccountLportGroup
             DESCRIPTION
                 "This group is optional for frame relay
                 interfaces.  It is mandatory if and only if
                 accounting is performed on a logical port basis
                 this frame relay interface."

             OBJECT      frPVCEndptInMaxFrameSize
```

```
            MIN-ACCESS  read-only
            DESCRIPTION
                "Write access is not required."

            OBJECT      frPVCEndptInBc
            MIN-ACCESS  read-only
            DESCRIPTION
                "Write access is not required."

            OBJECT      frPVCEndptInBe
            MIN-ACCESS  read-only
            DESCRIPTION
                "Write access is not required."

            OBJECT      frPVCEndptInCIR
            MIN-ACCESS  read-only
            DESCRIPTION
                "Write access is not required."

            OBJECT      frPVCEndptOutMaxFrameSize
            MIN-ACCESS  read-only
            DESCRIPTION
                "Write access is not required."

            OBJECT      frPVCEndptOutBc
            MIN-ACCESS  read-only
            DESCRIPTION
                "Write access is not required."

            OBJECT      frPVCEndptOutBe
            MIN-ACCESS  read-only
            DESCRIPTION
                "Write access is not required."

            OBJECT      frPVCEndptOutCIR
            MIN-ACCESS  read-only
            DESCRIPTION
                "Write access is not required."

            OBJECT      frPVCEndptRowStatus
        -- subset of RowStatus
            SYNTAX      INTEGER { active(1) }
            MIN-ACCESS  read-only
            DESCRIPTION
                "Write access is not required, and only one of the
                six enumerated values for the RowStatus textual
                convention need be supported, specifically:
                active(1)."
```

```
            OBJECT       frPVCConnectAdminStatus
            MIN-ACCESS   read-only
            DESCRIPTION
                 "Write access is not required."

            OBJECT       frPVCConnectRowStatus
         -- subset of RowStatus
            SYNTAX       INTEGER { active(1) }
            MIN-ACCESS   read-only
            DESCRIPTION
                 "Write access is not required, and only one of the
                 six enumerated values for the RowStatus textual
                 convention need be supported, specifically:
                 active(1)."

            OBJECT       frLportFragControl
            MIN-ACCESS   read-only
            DESCRIPTION
                 "Write access is not required."

            OBJECT       frLportFragSize
            MIN-ACCESS   read-only
            DESCRIPTION
                 "Write access is not required."

            OBJECT       frPVCConnectUserName
            MIN-ACCESS   read-only
            DESCRIPTION
                 "Write access is not required."

            OBJECT       frPVCConnectProviderName
            MIN-ACCESS   read-only
            DESCRIPTION
                 "Write access is not required."

          ::= { frnetservCompliances 2 }

      --
      -- Switch (Configuration) Compliance
      --
      frnetSwitchCompliance MODULE-COMPLIANCE
          STATUS       current
          DESCRIPTION
                 "The compliance statement for SNMP entities which
                 have Frame Relay Network Switch objects.

                 The distinction between 'service' and 'switch' is
                 that a 'switch' is configured via this MIB.
```

                    Hence, the various read/write objects have write
                    capability. A 'service' represents a passive
                    monitor-only customer network management
                    interface. The various read/write objects are
                    restricted to read-only capability."
            MODULE -- this module
                MANDATORY-GROUPS { frnetservLportGroup2,
                                   frnetservLportAdminGroup,
                                   frnetservMgtVCSigGroup,
                                   frnetservMgtVCSigAdminGroup,
                                   frnetservPVCEndptGroup,
                                   frnetservPVCEndptGroup2,
                                   frnetservPVCConnectGroup,
                                   frnetservPVCConnectNamesGroup,
                                   frnetservPVCNotifGroup2 }

            GROUP   frnetservAccountPVCGroup
            DESCRIPTION
                "This group is optional for frame relay
                interfaces. It is mandatory if and only if
                accounting is performed on a PVC basis this frame
                relay interface."
            GROUP   frnetservAccountLportGroup
            DESCRIPTION
                "This group is optional for frame relay
                interfaces. It is mandatory if and only if
                accounting is performed on a logical port basis
                this frame relay interface."

            ::= { frnetservCompliances 3 }

      --
      -- Historical RFC 1604 Compliance Modules
      --

      frnetservCompliance MODULE-COMPLIANCE
          STATUS deprecated
          DESCRIPTION
                  "The compliance statement for SNMP entities which
                  have Frame Relay Network Service Interfaces.

                  This compliance statement has been deprecated in
                  favor of frnetservCompliance2. The new compliance
                  module expands the mandatory groups to include
                  notification and other new objects."

          MODULE -- this module
              MANDATORY-GROUPS { frnetservLportGroup,

```
                            frnetservMgtVCSigGroup,
                            frnetservPVCEndptGroup,
                            frnetservPVCConnectGroup }

        GROUP       frnetservAccountPVCGroup
        DESCRIPTION
            "This group is optional for Frame Relay
            interfaces.  It is mandatory if and only if
            accounting is performed on a PVC basis this Frame
            Relay interface."

        GROUP       frnetservAccountLportGroup
        DESCRIPTION
            "This group is optional for Frame Relay
            interfaces.  It is mandatory if and only if
            accounting is performed on a logical port basis
            this Frame Relay interface."

        OBJECT      frPVCEndptInMaxFrameSize
        MIN-ACCESS  read-only
        DESCRIPTION
            "Write access is not required."

        OBJECT      frPVCEndptInBc
        MIN-ACCESS  read-only
        DESCRIPTION
            "Write access is not required."
        OBJECT      frPVCEndptInBe
        MIN-ACCESS  read-only
        DESCRIPTION
            "Write access is not required."

        OBJECT      frPVCEndptInCIR
        MIN-ACCESS  read-only
        DESCRIPTION
            "Write access is not required."

        OBJECT      frPVCEndptOutMaxFrameSize
        MIN-ACCESS  read-only
        DESCRIPTION
            "Write access is not required."

        OBJECT      frPVCEndptOutBc
        MIN-ACCESS  read-only
        DESCRIPTION
            "Write access is not required."

        OBJECT      frPVCEndptOutBe
```

```
            MIN-ACCESS  read-only
            DESCRIPTION
                "Write access is not required."

            OBJECT      frPVCEndptOutCIR
            MIN-ACCESS  read-only
            DESCRIPTION
                "Write access is not required."

            OBJECT      frPVCEndptRowStatus
        -- subset of RowStatus
            SYNTAX      INTEGER { active(1) }
            MIN-ACCESS  read-only
            DESCRIPTION
                "Write access is not required, and only one of the
                six enumerated values for the RowStatus textual
                convention need be supported, specifically:
                active(1)."

            OBJECT      frPVCConnectAdminStatus
            MIN-ACCESS  read-only
            DESCRIPTION
                "Write access is not required."

            OBJECT      frPVCConnectRowStatus
        -- subset of RowStatus
            SYNTAX      INTEGER { active(1) }
            MIN-ACCESS  read-only
            DESCRIPTION
                "Write access is not required, and only one of the
                six enumerated values for the RowStatus textual
                convention need be supported, specifically:
                active(1)."
        ::= { frnetservCompliances 1 }

    --
    -- Frame Relay Service MIB Object Groups
    --
    frnetservLportGroup  OBJECT-GROUP
        OBJECTS { frLportNumPlan, frLportContact, frLportLocation,
                  frLportType,
                  frLportAddrDLCILen, frLportVCSigProtocol,
                  frLportVCSigPointer }
        STATUS  deprecated
        DESCRIPTION
                "A collection of objects providing information
                 applicable to a Frame Relay Logical Port.  This
                 group has been deprecated to eliminate reference
```

```
                        to the object frLportVCSigPointer.

                        Use the new group frnetservLportGroup2 as a
                        replacement for this group."
                ::= { frnetservGroups 1 }

        frnetservMgtVCSigGroup  OBJECT-GROUP
            OBJECTS { frMgtVCSigProced,
                        frMgtVCSigUserN391,
                        frMgtVCSigUserN392,
                        frMgtVCSigUserN393,
                        frMgtVCSigUserT391,
                        frMgtVCSigNetN392,
                        frMgtVCSigNetN393,
                        frMgtVCSigNetT392,
                        frMgtVCSigNetnN4,
                        frMgtVCSigNetnT3,
                        frMgtVCSigUserLinkRelErrors,
                        frMgtVCSigUserProtErrors,
                        frMgtVCSigUserChanInactive,
                        frMgtVCSigNetLinkRelErrors,
                        frMgtVCSigNetProtErrors,
                        frMgtVCSigNetChanInactive }
            STATUS  current
            DESCRIPTION
                    "A collection of objects providing information
                    applicable to the Local In-Channel Signaling
                    Procedures used for a UNI/NNI logical port."
                ::= { frnetservGroups 2 }

        frnetservPVCEndptGroup  OBJECT-GROUP
            OBJECTS { frPVCConnectIndexValue,
                        frPVCEndptInMaxFrameSize,
                        frPVCEndptInBc,
                        frPVCEndptInBe,
                        frPVCEndptInCIR,
                        frPVCEndptOutMaxFrameSize,
                        frPVCEndptOutBc,
                        frPVCEndptOutBe,
                        frPVCEndptOutCIR,
                        frPVCEndptConnectIdentifier,
                        frPVCEndptRowStatus,
                        frPVCEndptRcvdSigStatus,
                        frPVCEndptInFrames,
                        frPVCEndptOutFrames,
                        frPVCEndptInDEFrames,
                        frPVCEndptInExcessFrames,
                        frPVCEndptOutExcessFrames,
```

```
                     frPVCEndptInDiscards,
                     frPVCEndptInOctets,
                     frPVCEndptOutOctets }
            STATUS   current
            DESCRIPTION
                    "A collection of objects providing information
                    applicable to a Frame Relay PVC end-point."
            ::= { frnetservGroups 3 }

        frnetservPVCConnectGroup   OBJECT-GROUP
            OBJECTS { frPVCConnectAdminStatus,
                     frPVCConnectL2hOperStatus,
                     frPVCConnectH2lOperStatus,
                     frPVCConnectL2hLastChange,
                     frPVCConnectH2lLastChange,
                     frPVCConnectRowStatus }
            STATUS   current
            DESCRIPTION
                    "A collection of objects providing information
                    applicable to a Frame Relay PVC connection."
            ::= { frnetservGroups 4 }

        frnetservAccountPVCGroup   OBJECT-GROUP
            OBJECTS { frAccountPVCSegmentSize,
                     frAccountPVCInSegments,
                     frAccountPVCOutSegments }
            STATUS   current
            DESCRIPTION
                    "A collection of objects providing accounting
                    information application to a Frame Relay PVC end-
                    point."
            ::= { frnetservGroups 5 }

        frnetservAccountLportGroup   OBJECT-GROUP
            OBJECTS { frAccountLportSegmentSize,
                     frAccountLportInSegments,
                     frAccountLportOutSegments }
            STATUS   current
            DESCRIPTION
                    "A collection of objects providing accounting
                    information application to a Frame Relay logical
                    port."
            ::= { frnetservGroups 6 }

        frnetservLportGroup2   OBJECT-GROUP
            OBJECTS { frLportNumPlan,
                     frLportContact,
                     frLportLocation,
```

```
                    frLportType,
                    frLportAddrDLCILen,
                    frLportVCSigProtocol,
                    frLportFragControl,
                    frLportFragSize }
        STATUS  current
        DESCRIPTION
                "A collection of objects providing information
                applicable to a Frame Relay Logical Port.

                This new version of the Logical Port Group
                eliminates the frLportVCSigPointer and adds
                support for fragmentation."
        ::= { frnetservGroups 7 }

frnetservPVCEndptGroup2  OBJECT-GROUP
        OBJECTS { frPVCEndptInDiscardsDESet,
                    frPVCEndptInFramesFECNSet,
                    frPVCEndptOutFramesFECNSet,
                    frPVCEndptInFramesBECNSet,
                    frPVCEndptOutFramesBECNSet,
                    frPVCEndptInCongDiscards,
                    frPVCEndptInDECongDiscards,
                    frPVCEndptOutCongDiscards,
                    frPVCEndptOutDECongDiscards,
                    frPVCEndptOutDEFrames,
                    frPVCEndptAtmIwfConnIndex }
        STATUS  current
        DESCRIPTION
                "Additions to the PVC end-point group.  These
                additions provide new frame counters to track
                frame loss.  In addition, the new FR/ATM IWF MIB
                cross-connect index is included."
        ::= { frnetservGroups 8 }

frnetservPVCConnectNamesGroup  OBJECT-GROUP
        OBJECTS { frPVCConnectUserName,
                    frPVCConnectProviderName }
        STATUS  current
        DESCRIPTION
                "Additions to the PVC Connect Group."
        ::= { frnetservGroups 9 }

frnetservLportAdminGroup OBJECT-GROUP
        OBJECTS { frLportDLCIIndexValue,
                    frLportTypeAdmin,
                    frLportVCSigProtocolAdmin }
        STATUS  current
```

```
         DESCRIPTION
                 "Administrative (R/W) objects for creating a
                 switch logical port."
         ::= { frnetservGroups 10 }


     frnetservMgtVCSigAdminGroup OBJECT-GROUP
         OBJECTS { frMgtVCSigProcedAdmin,
                   frMgtVCSigUserN391Admin,
                   frMgtVCSigUserN392Admin,
                   frMgtVCSigUserN393Admin,
                   frMgtVCSigUserT391Admin,
                   frMgtVCSigNetN392Admin,
                   frMgtVCSigNetN393Admin,
                   frMgtVCSigNetT392Admin,
                   frMgtVCSigNetnT3Admin }
         STATUS   current
         DESCRIPTION

                 "A collection of objects providing information
                 applicable to the Local In-Channel Signaling
                 Procedures used for a UNI/NNI logical port."
         ::= { frnetservGroups 11 }


     frnetservPVCNotifGroup NOTIFICATION-GROUP
         NOTIFICATIONS { frPVCConnectStatusChange }
         STATUS    deprecated
         DESCRIPTION
                 "Deprecated notification group.  The
                 frPVCConnectStatusChange notification was flawed
                 because it included redundant indexes and was not
                 properly encoded for SMIv1 conversion."
        ::= { frnetservGroups 12 }


     frnetservPVCNotifGroup2 NOTIFICATION-GROUP
         NOTIFICATIONS { frPVCConnectStatusNotif }
         STATUS    current
         DESCRIPTION
                 "A collection of notifications that apply to frame
                 relay PVC Connections "
        ::= { frnetservGroups 13 }


     END
```

4. Acknowledgments

   This document was produced by the Frame Relay Service MIB Working
   Group.

   The working group thanks Bert Wijnen, David Perkins, and Bob Stewart
   for their assistance in reviewing the MIB.

5. References

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

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

   [3] Rose, M. and K. McCloghrie, "Concise MIB Definitions", STD 16,
        RFC 1212, March 1991.

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

   [5] 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.

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

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

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

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

   [10] 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.

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

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

   [13] 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.

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

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

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

   [17] ANSI T1.617-1991, American National Standard for
        Telecommunications - Integrated Services Digital Network (ISDN)
        - Digital Subscriber Signaling System No. 1 (DSS1) - Signaling
        Specification for Frame Relay Bearer Service.

   [18] Brown, C. and F. Baker, "Management Information Base for Frame
        Relay DTEs", RFC 2115, September 1997.

   [19] Brown, C. and A. Malis, "Multi-Protocol Interconnect over Frame
        Relay", STD 55, RFC 2427, September 1998.

   [20] Fowler, D, "Definitions of Managed Objects for the DS0 and DS0
        Bundle Interface Types", RFC 2494, January 1999.

   [21] Frame Relay Forum, "Frame Relay Fragmentation Implementation
        Agreement", FRF.12, December 1997.

   [22] ITU-T Recommendation Q.933,Integrated Services Digital Network
        (ISDN) Digital Subscriber Signalling System No. 1 (DSS 1) -
        Signalling Specifications for Frame Mode Switched and Permanent
        Virtual Connection Control and Status Monitoring, December 1995

    [23] ITU-T Recommendation X.36, Interface Between Data Terminal
         Equipment (DTE) and Data Circuit-Terminating Equipment (DCE) For
         Public Data Networks Providing Frame Relay Data Transmission
         Service By Dedicated Circuit, April 1995

    [24] Digital Equipment Corporation, et. al., "Frame Relay
         Specification with Extensions Based on Proposed T1S1 Standards",
         Revision 1.0, September 18, 1990

    [25] ITU-T Recommendation Q.922, Integrated Services Digital Network
         (ISDN) Data Link Layer Specification For Frame Mode Bearer
         Services, February 1992

    [26] McCloghrie, K. and F. Kastenholz, "The Interfaces Group MIB",
         RFC 2863, June 2000.

    [27] 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.

    [28] Rehbehn, K., Nicklass, O. and G. Mouradian, "Definitions of
         Managed Objects for Monitoring and Controlling the Frame
         Relay/ATM PVC Service Interworking Function", RFC 2955, October
         2000.

    [29] ITU-T Recommendation E.164/I.331, The International Public
         Telecommunication Numbering Plan, May 1997

    [30] ITU-T Recommendation X.121, International Numbering Plan For
         Public Data Networks, October 1996

    [31] Frame Relay Forum, "The Frame Relay Forum User-to-Network
         Implementation Agreement (UNI)", FRF 1.2, July 2000.

6. 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.

    No managed objects in this MIB contain sensitive information.

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 [12] and the View-based
Access Control Model RFC 2575 [15] 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.

7. Authors' Addresses

Kenneth Rehbehn
Megisto Systems, Inc.
20251 Century Boulevard
Germantown, MD, USA 20874

Phone: (301) 515-3672
EMail: krehbehn@megisto.com

David Fowler
Syndesis Limited
28 Fulton Way
Richmond Hill, Ontario, Canada L4B 1J5

Phone: (905) 886-7818
EMail: fowler@syndesis.com

APPENDIX A Update Information

   The changes from RFC 1604 are the following:

   (1) Added the object frLportDLCIIndexValue to automatically generate
        index values for new DLC rows.

   (2) Add the following objects to support manager writing to objects:

        Logical Port Objects
            frLportTypeAdmin
            frLportVCSigProtocolAdmin

        VC Objects
            frMgtVCSigProcedAdmin
            frMgtVCSigUserN391Admin
            frMgtVCSigUserN392Admin
            frMgtVCSigUserN393Admin
            frMgtVCSigUserT391Admin
            frMgtVCSigNetN392Admin
            frMgtVCSigNetN393Admin
            frMgtVCSigNetT392Admin
            frMgtVCSigNetnT3Admin

   (3) Add objects to control fragmentation:

        frLportFragControl frLportFragSize

   (4) Added objects to track frames offered to network (in) and
        delivered (out) for increased visibility into policing-driven
        discards, congestion-driven discards, DE-bit setting, and
        congestion bit setting:

        frPVCEndptInDiscardsDESet
        frPVCEndptInFramesFECNSet
        frPVCEndptOutFramesFECNSet
        frPVCEndptInFramesBECNSet
        frPVCEndptOutFramesBECNSet
        frPVCEndptInCongDiscards
        frPVCEndptInDECongDiscards
        frPVCEndptOutCongDiscards
        frPVCEndptOutDECongDiscards
        frPVCEndptOutDEFrames

   (5) Added the PVC object frPVCEndptAtmIwfConnIndex to identify the
        type of connection, frame relay or ATM IWF; and to identify the
        cross-connect row of the FR/ATM IWF MIB.

   (6) Added objects to provide printable names of the connection user
        and service provider:

        frPVCConnectUserName
        frPVCConnectProviderName

   (7) Added a new notification to correct flaws in the RFC1604 trap.
        The flaws include improper OID suffix (SMIv1 compatibility
        issue) and the inclusion of redundant index fields

   (8) Updated compliance modules and object groups to reflect the new
        objects and notification:

        frnetservCompliance2 -   New service-centric (read-only)
                                    compliance module

        frnetSwitchCompliance -   New switch-centric (read-write)
                                    compliance module

        frnetservCompliance -   Original RFC 1604 Module, now
                                    deprecated

        frnetservLportGroup -   Original RFC 1604 logical port
                                    group, now deprecated

        frnetservLportGroup2 -   Replacement logical port group

        frnetservPVCEndptGroup2 -   Extension objects with this
                                    revision of the MIB

        frnetservPVCConnectNamesGroup - New group w/ display names
                                    for connections

        frnetservLportAdminGroup - New group w/ read-write objects
                                    for the logical port

        frnetservMgtVCSigAdminGroup - New group w/ read-write objects
                                    for the signaling protocol

        frnetservPVCNotifGroup -   Group deprecated to eliminate
                                    obsolete frPVCConnectStatusChange
                                    notification

        frnetservPVCNotifGroup2 -   New group added with w/
                                    frPVCConnectStatusNotif

   (9) Added UNITS and REFERENCE clauses for objects that needed the
        clarification.

(10) Changed references to "proxy-agent" to "FRS agent" to avoid
     confusion with other proxy-agent terminology.

(11) Changed objects using the DisplayString TC to use the
     SnmpAdminString TC.

(12) frMgtVCSigProced - Expanded to include the u2nuser(3)
     enumeration for the UNI protocol operation where the logical
     port operates in the user role.

(13) DESCRIPTION text added to specify agent response when object is
     not instantiated for the following objects:

     frMgtVCSigUserN391
     frMgtVCSigUserN393
     frMgtVCSigUserT391
     frMgtVCSigUserN392
     frMgtVCSigNetN391
     frMgtVCSigNetN393
     frMgtVCSigNetT391
     frMgtVCSigNetN392
     frMgtVCSigNetnN4
     frMgtVCSigNetnT3
     frMgtVCSigUserLinkRelErrors
     frMgtVCSigUserProtErrors
     frMgtVCSigUserChanInactive

(14) DESCRIPTION text addressing case of logical port not performing
     network-side procedures was removed from following objects:

     frMgtVCSigNetLinkRelErrors
     frMgtVCSigNetChanInactive
     frMgtVCSigNetProtErrors

(15) frPVCEndptConnectIdentifier - Operation described for the case
     of FR/ATM IWF cross-connect operation.

(16) frPVCEndptRcvdSigStatus - Added description of enumerated
     values.

(17) frPVCEndptInDiscards - Clarified DESCRIPTION to state that
     congestion discards are not counted by object.

(18) frPVCConnect{Low|High}IfIndex - Changed to use InterfaceIndex TC
     and changed reference to MIB-II to the new IF-MIB. Removed
     statement asserting that a zero value means the port is not a FR
     logical port.

   (19) frPVCConnectIndex - Added a range to the SYNTAX clause

   (20) frPVCConnect{L2h|H2l}OperStatus - Added DESCRIPTION text for
        each enumerated value.

   (21) frAccountPVCDLCIIndex - Added a range to the SYNTAX clause

   (22) frPVCCOnnectStatusChange Notification - STATUS change to
        deprecated. Obsoleted to eliminate inappropriate inclusion of
        index objects

   (23) frPVCConnectStatusNotif Notification - Replaces
        frPVCConnectStatusChange. In addition, the notification now
        requires 2 instances of the frPVCEndptRcvdSigStatus object, one
        for each endpoint of the connection.

   (24) Guidance added to recommend ifLinkUpDownTrapEnable be set on.

   (25) Behavior of the PVC status and endpoint signaling status is
        clarified for the case of underlying layer failure.

   (26) Overview text re-written for clarity.

   (27) Clarified role of system group.

   (28) Established maximum frame size of 4096 and default value of
        1600.

   (29) Clarified that DLC index range is restricted to valid range for
        the specific length of address field used on the logical port.

   (30) Figure 1 and accompanying text was removed to eliminate a
        confusing "MIB stack" concept. See the section titled "Relation
        to Other MIBs" for replacement text.

Intellectual Property Rights

   The IETF takes no position regarding the validity or scope of any
   intellectual property or other rights that might be claimed to
   pertain to the implementation or use of the technology described in
   this document or the extent to which any license under such rights
   might or might not be available; neither does it represent that it
   has made any effort to identify any such rights. Information on the
   IETF's procedures with respect to rights in standards-track and
   standards-related documentation can be found in BCP-11. Copies of
   claims of rights made available for publication and any assurances of
   licenses to be made available, or the result of an attempt made to
   obtain a general license or permission for the use of such
   proprietary rights by implementers or users of this specification can
   be obtained from the IETF Secretariat.

   The IETF invites any interested party to bring to its attention any
   copyrights, patents or patent applications, or other proprietary
   rights which may cover technology that may be required to practice
   this standard. Please address the information to the IETF Executive
   Director.

Full Copyright Statement

Acknowledgement