

Management Information Base for Frame Relay DTEs
[draft-ietf-iplpdm-frmib-dte-09.txt](#)

Wed Dec 4 17:02:10 PST 1996

Caralyn Brown

Cadia Networks, Inc.
1 Corporate Drive
Andover, Massachusetts 01810

cbrown@cadia.com

Fred Baker

Cisco Systems
519 Lado Drive
Santa Barbara, California 93111

fred@cisco.com

1. Status of this Memo

This document is an Internet-Draft. Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months. Internet-Drafts may be updated, replaced, or obsoleted by other documents at any time. It is not appropriate to use Internet-Drafts as reference material or to cite them other than as a ``working draft'' or ``work in progress.''

To learn the current status of any Internet-Draft, please check the `1id-abstracts.txt` listing contained in the Internet-Drafts Shadow Directories on `ds.internic.net`, `nic.nordu.net`, `ftp.nisc.sri.com`, or `munnari.oz.au`.

This draft document will be submitted to the RFC editor as an extension to the SNMP MIB. Please send comments to the authors, copying `frs-mib@newbridge.com` and `ion@nexen.com`. It expires June 1997.

2. Abstract

This memo defines a portion of the Management Information Base (MIB) for use with network management protocols in TCP/IP-based internets. In particular, it defines objects for managing Frame Relay interfaces on DTEs.

This memo does not specify a standard for the Internet community.

3. The SNMPv2 Network Management Framework

The major components of the SNMPv2 Network Management framework are described in the documents listed below.

- o [RFC 1902](#) [1] defines the Structure of Management Information (SMI), the mechanisms used for describing and naming objects for the purpose of management.
- o STD 17, [RFC 1213](#) [2] defines MIB-II, the core set of managed objects (MO) for the Internet suite of protocols.
- o [RFC 1905](#) [3] defines the protocol used for network access to managed objects.

The framework is adaptable/extensible by defining new MIBs to suit the requirements of specific applications/protocols/situations.

Managed objects are accessed via a virtual information store, the MIB. Objects in the MIB are defined using the subset of Abstract Syntax Notation One (ASN.1) defined in the SMI. In particular, each object type is named by an OBJECT

IDENTIFIER, which is an administratively assigned name. The object type together with an object instance serves to uniquely identify a specific instantiation of the object. For human convenience, often a textual string, termed the descriptor, is used to refer to the object type.

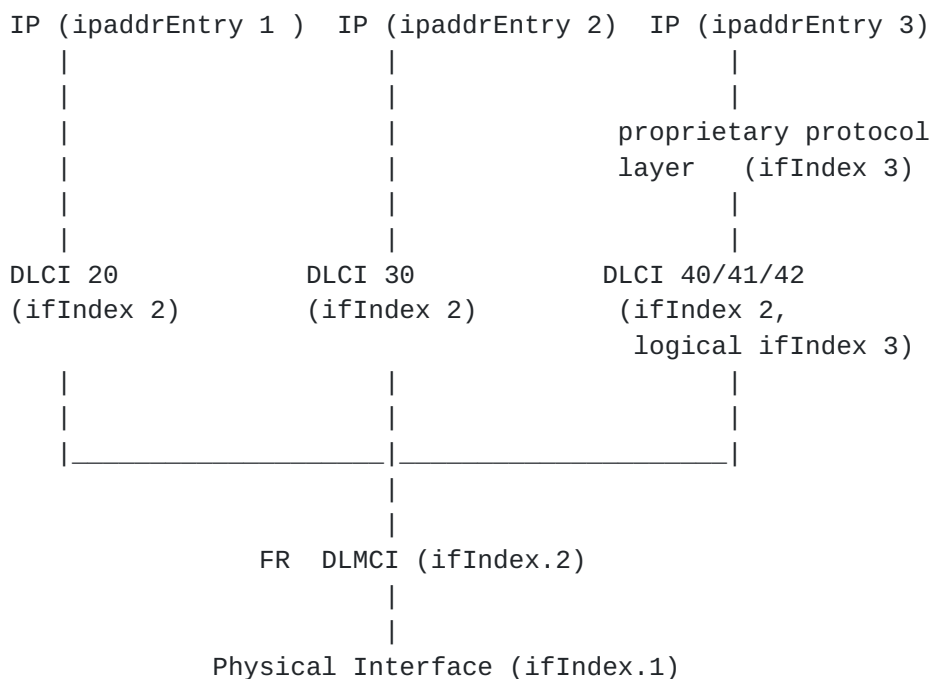
4. Overview

4.1. Frame Relay Operational Model

For the purposes of understanding this document, Frame Relay is viewed as a multi-access media, not as a group of point-to-point connections. This model proposes that Frame Relay is a single interface to the network (physical connection) with many destinations or neighbors (virtual connections). This view enables a network manager the ability to group all virtual connections with their corresponding physical connection thereby allowing simpler diagnostics and trouble shooting.

With the extension of the interfaces MIB, it is possible to configure frame relay DLCs as individual interfaces and create ifTable entries for each. This is not recommended and is not directly supported by this MIB. Additionally, in the presence of demand circuits creation of individual ifEntries for each is not possible.

Should the user wish to group DLCs together to associate them with a higher layer, or to associate a DLC with an unnumbered point-to-point service, the frame relay DTE MIB provides an entry in the frCircuitEntry record. For example, suppose one were to configure a company proprietary protocol to run above several of the frame relay VCs. The basic layering would look something like the following:



A configuration which specified that DLCI 40, 41, and 42 were associated with a proprietary protocol layer, while DLCI 20 and 30 were to run IP directly can now be expressed using a combination of frCircuitIfIndex and frCircuitLogicalIfIndex. In this particular case DLCIs 40, 41 and 42 would use frCircuitIfIndex equal to the frame relay interface level (2) while their frCircuitLogicalIfIndex would indicate the proprietary protocol (3). DLCIs 20 and 30 would have both instances set to the frame relay interface (2).

Object	Meaning for Frame Relay Interface
ifDescr	As per DESCRIPTION in RFC 1573 .
ifType	The value allocated for Frame Relay Interfaces - frameRelay (32).
ifMtu	Set to maximum frame size in octets for this frame relay interface.
ifSpeed	The access rate for the frame relay interface. This could be different from the speed of the underlying physical

interface, e.g. in a fractional T1 case the access rate could be 384 kbits/s (the value reported in this object) whereas the speed of the underlying interface would be 1.544 Mbits/s (the value reported in the instance of ifSpeed for the ifEntry with type ds1).

- ifPhysAddress The primary address for this interface assigned by the Frame Relay interface provider. An octet string of zero length if no address is used for this interface.
- ifAdminStatus As per DESCRIPTION in [RFC 1573](#).
- ifOperStatus As per DESCRIPTION in [RFC 1573](#).
- ifLastChange As per DESCRIPTION in [RFC 1573](#).
- ifInOctets The number of received octets. This includes not only the information field (user data) but also the frame relay header and CRC.
- ifInUcastPkts The number of frames received on non-multicast DLCIs
- ifInDiscards The number of frames that were successfully received but were discarded because of format errors or because the VC was not known. Format errors, in this case, are any errors which would prevent the system from recognizing the DLCI and placing the error in the frCircuitDiscard category.
- ifInErrors The number of received frames that are discarded, because of an error. Possible errors can be the following: the frame relay frames were too long or were too short, the frames had an invalid or unrecognized DLCI values, or incorrect header values.
- ifInUnknownProtos Number of unknown or unsupported upper layer protocol frames received

and discarded.

- `ifOutOctets` The number of received octets. This includes not only the information field (user data) but also the frame relay header and CRC.
- `ifOutUcastPkts` The number of frames sent.
- `ifOutDiscards` The number of frames discarded in the transmit direction.
- `ifOutErrors` The number of frames discarded in the egress direction, because of errors.
- `ifName` As per DESCRIPTION in [RFC 1573](#).
- `ifInMulticastPkts` The number of unerrored frames received on a multicast DLCI.
- `ifInBroadcastPkts` Always zero (0) as there are no broadcast frames.
- `ifOutMulticastPkts` The number of frames transmitted over a multicast DLCI.
- `ifOutBroadcastPkts` Always zero (0) as there are no broadcast frames.
- `ifHCInOctets` Only required when `ifSpeed` \geq 155 Mbits/s.
See details for `ifInOctets`.
- `ifHCOctets` Only required when `ifSpeed` \geq 155 Mbits/s.
See details for `ifInOctets`.
- `ifLinkUpDownTrapEnable` As per DESCRIPTION in [RFC 1573](#).
- `ifHighSpeed` The access rate of the frame relay interface measured in Mbits/s. If the access rate is less than 1 Mbits/s, this object returns 0.
- `ifPromiscuousMode` Set to false(2).

ifConnectorPresent Set to false(2).

4.2. Textual Conventions

One new data type is introduced as a textual convention in this MIB document. This textual convention enhances the readability of the specification and can ease comparison with other specifications if appropriate. It should be noted that the introduction of this textual conventions has no effect on either the syntax nor the semantics of any managed objects. The use of this is merely an artifact of the explanatory method used. Objects defined in terms of one of these methods are always encoded by means of the rules that define the primitive type. Hence, no changes to the SMI or the SNMP are necessary to accommodate this textual conventions which is adopted merely for the convenience of readers and writers in pursuit of the elusive goal of clear, concise, and unambiguous MIB documents.

The new data type is DLCI. DLCI refers to the range 0..DLCINumber, and is used to refer to the valid Data Link Connection Indices. DLCINumber is, by definition, the largest possible DLCI value possible under the configured Q.922 Address Format.

4.3. Structure of MIB

The MIB is composed of three groups, one defining the Data Link Connection Management Interface (DLCMI), one describing the Circuits, and a third describing errors.

During normal operation, Frame Relay virtual circuits will be added, deleted and change availability. The occurrence of such changes is of interest to the network manager and therefore, one trap is defined, intended to be corollary to the SNMP "Link Up" and "Link Down" traps.

5. Definitions

```
FRAME-RELAY-DTE-MIB DEFINITIONS ::= BEGIN
```

```
IMPORTS
```

```
    MODULE-IDENTITY, OBJECT-TYPE, Counter32,
    Integer32, NOTIFICATION-TYPE          FROM SNMPv2-SMI
    TEXTUAL-CONVENTION, RowStatus, TimeStamp FROM SNMPv2-TC
    MODULE-COMPLIANCE, OBJECT-GROUP      FROM SNMPv2-CONF
    transmission                          FROM RFC1213-MIB
    InterfaceIndex                        FROM IF-MIB;
```

```
-- Frame Relay DTE MIB
```

```
frameRelayDTE MODULE-IDENTITY
```

```
    LAST-UPDATED "9612041702Z" -- Wed Dec 4 17:02:10 PST 1996
```

```
    ORGANIZATION "IETF IPLPDN Working Group"
```

```
    CONTACT-INFO
```

```
        "      Caralyn Brown
    Postal: Cadia Networks, Inc.
            1 Corporate Drive
            Andover, Massachusetts 01810
    Tel:    +1 508 689 2400 x133
    E-Mail: cbrown@cadia.com
```

```
        Fred Baker
    Postal: Cisco Systems
            519 Lado Drive
            Santa Barbara, California 93111
    Tel:    +1 408 526 4257
    E-Mail: fred@cisco.com"
```

```
DESCRIPTION
```

```
    "The MIB module to describe the use of a Frame Relay
    interface by a DTE. This was converted from RFC 1315's
    SMI-V1 to SMI-V2; therefore, indices are read-only
    rather than being not-accessible."
```

```
 ::= { transmission 32 }
```

```
--
```

```
--     the range of a Data Link Connection Identifier
```

```
--
```

```
DLCI ::= TEXTUAL-CONVENTION
```

```
    STATUS      current
```

```
DESCRIPTION
```


"The range of DLCI values. Note that this varies by interface configuration; normally, interfaces may use 0..1023, but may be configured to use ranges as large as 0..2²³."

SYNTAX INTEGER (0..8388607)

--

-- Data Link Connection Management Interface

-- The variables that configure the DLC Management Interface.

frDlcmiTable OBJECT-TYPE

SYNTAX SEQUENCE OF FrDlcmiEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The Parameters for the Data Link Connection Management Interface for the frame relay service on this interface."

REFERENCE

"American National Standard T1.617-1991, Annex D"

::= { frameRelayDTE 1 }

frDlcmiEntry OBJECT-TYPE

SYNTAX FrDlcmiEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The Parameters for a particular Data Link Connection Management Interface."

INDEX { frDlcmiIfIndex }

::= { frDlcmiTable 1 }

FrDlcmiEntry ::=

SEQUENCE {

frDlcmiIfIndex InterfaceIndex,

frDlcmiState INTEGER,

frDlcmiAddress INTEGER,

frDlcmiAddressLen INTEGER,

frDlcmiPollingInterval INTEGER,

frDlcmiFullEnquiryInterval INTEGER,

frDlcmiErrorThreshold INTEGER,

frDlcmiMonitoredEvents INTEGER,

frDlcmiMaxSupportedVCs DLCI,

frDlcmiMulticast INTEGER,

frDlcmiStatus INTEGER,

frDlcmiRowStatus RowStatus

}

frDlcmiIfIndex OBJECT-TYPE


```
SYNTAX      InterfaceIndex
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "The ifIndex value of the corresponding ifEntry."
 ::= { frDlcmiEntry 1 }
```

frDlcmiState OBJECT-TYPE

```
SYNTAX INTEGER {
    noLmiConfigured (1),
    lmiRev1          (2),
    ansiT1617D      (3), -- ANSI T1.617 Annex D
    ansiT1617B      (4), -- ANSI T1.617 Annex B
    itut933A        (5), -- CCITT Q933 Annex A
    ansiT1617D1994 (6)  -- ANSI T1.617a-1994 Annex D
}
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "This variable states which Data Link Connection
    Management scheme is active (and by implication, what
    DLCI it uses) on the Frame Relay interface."
REFERENCE
    "American National Standard T1.617-1991, American
    National Standard T1.617a-1994, ITU-T Recommendation
    Q.933 (03/93)."
```

```
 ::= { frDlcmiEntry 2 }
```

frDlcmiAddress OBJECT-TYPE

```
SYNTAX      INTEGER {
    q921              (1), -- 13 bit DLCI
    q922March90       (2), -- 11 bit DLCI
    q922November90    (3), -- 10 bit DLCI
    q922              (4)  -- Final Standard
}
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "This variable states which address format is in use on
    the Frame Relay interface."
 ::= { frDlcmiEntry 3 }
```


frDlcmiAddressLen OBJECT-TYPE

```
SYNTAX  INTEGER    {
        twoOctets (2),
        threeOctets (3),
        fourOctets (4)
    }
```

```
MAX-ACCESS  read-create
```

```
STATUS      current
```

DESCRIPTION

"This variable states the address length in octets. In the case of Q922 format, the length indicates the entire length of the address including the control portion."

```
::= { frDlcmiEntry 4 }
```

frDlcmiPollingInterval OBJECT-TYPE

```
SYNTAX  INTEGER (5..30)
```

```
UNITS    "seconds"
```

```
MAX-ACCESS  read-create
```

```
STATUS      current
```

DESCRIPTION

"This is the number of seconds between successive status enquiry messages."

REFERENCE

"American National Standard T1.617-1991, Section D.7
Timer T391."

```
DEFVAL { 10 }
```

```
::= { frDlcmiEntry 5 }
```

frDlcmiFullEnquiryInterval OBJECT-TYPE

```
SYNTAX  INTEGER (1..255)
```

```
MAX-ACCESS  read-create
```

```
STATUS      current
```

DESCRIPTION

"Number of status enquiry intervals that pass before issuance of a full status enquiry message."

REFERENCE

"American National Standard T1.617-1991, Section D.7
Counter N391."

```
DEFVAL { 6 }
```

```
::= { frDlcmiEntry 6 }
```


frDlcmiErrorThreshold OBJECT-TYPE

SYNTAX INTEGER (1..10)

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"This is the maximum number of unanswered Status Enquiries the equipment shall accept before declaring the interface down."

REFERENCE

"American National Standard T1.617-1991, Section D.5.1 Counter N392."

DEFVAL { 3 }

::= { frDlcmiEntry 7 }

frDlcmiMonitoredEvents OBJECT-TYPE

SYNTAX INTEGER (1..10)

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"This is the number of status polling intervals over which the error threshold is counted. For example, if within 'MonitoredEvents' number of events the station receives 'ErrorThreshold' number of errors, the interface is marked as down."

REFERENCE

"American National Standard T1.617-1991, Section D.5.2 Counter N393."

DEFVAL { 4 }

::= { frDlcmiEntry 8 }

frDlcmiMaxSupportedVCs OBJECT-TYPE

SYNTAX DLCI

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The maximum number of Virtual Circuits allowed for this interface. Usually dictated by the Frame Relay network.

In response to a SET, if a value less than zero or higher than the agent's maximal capability is configured, the agent should respond badValue"

::= { frDlcmiEntry 9 }

frDlcmiMulticast OBJECT-TYPE

```
SYNTAX  INTEGER  {
    nonBroadcast (1),
    broadcast (2)
}
```

```
MAX-ACCESS  read-create
```

```
STATUS  current
```

DESCRIPTION

"This indicates whether the Frame Relay interface is using a multicast service."

```
::= { frDlcmiEntry 10 }
```

frDlcmiStatus OBJECT-TYPE

```
SYNTAX  INTEGER  {
    running (1),      -- init complete, system running
    fault (2),       -- error threshold exceeded
    initializing (3)  -- system start up
}
```

```
MAX-ACCESS  read-only
```

```
STATUS  current
```

DESCRIPTION

"This indicates the status of the Frame Relay interface as determined by the performance of the dlcmi. If no dlcmi is running, the Frame Relay interface will stay in the running state indefinitely."

```
::= { frDlcmiEntry 11 }
```

frDlcmiRowStatus OBJECT-TYPE

```
SYNTAX  RowStatus
```

```
MAX-ACCESS  read-create
```

```
STATUS  current
```

DESCRIPTION

"SNMP Version 2 Row Status Variable."

```
::= { frDlcmiEntry 12 }
```

```
--
```



```
-- A Frame Relay service is a multiplexing service.  Data
-- Link Connection Identifiers enumerate virtual circuits
-- (permanent or dynamic) which are layered onto the underlying
-- circuit, represented by ifEntry.  Therefore, each of the entries
-- in the Standard MIB's Interface Table with an IfType of
-- Frame Relay represents a Q.922 interface.  Zero or more
-- virtual circuits are layered onto this interface and provide
-- interconnection with various remote destinations.
-- Each such virtual circuit is represented by an entry in the
-- circuit table.  The management virtual circuit (i.e. DLCI 0)
-- is a virtual circuit by this definition and will be represented
-- with an entry in the circuit table.
```

```
--  Circuit Table
```

```
-- The table describing the use of the DLCIs attached to
-- each Frame Relay Interface.
```

```
frCircuitTable OBJECT-TYPE
```

```
    SYNTAX      SEQUENCE OF FrCircuitEntry
```

```
    MAX-ACCESS  not-accessible
```

```
    STATUS      current
```

```
    DESCRIPTION
```

```
        "A table containing information about specific Data
        Link Connections (DLC) or virtual circuits."
```

```
    ::= { frameRelayDTE 2 }
```

```
frCircuitEntry OBJECT-TYPE
```

```
    SYNTAX      FrCircuitEntry
```

```
    MAX-ACCESS  not-accessible
```

```
    STATUS      current
```

```
    DESCRIPTION
```

```
        "The information regarding a single Data Link
        Connection."
```

```
    INDEX { frCircuitIfIndex, frCircuitDlci }
```

```
    ::= { frCircuitTable 1 }
```

```
FrCircuitEntry ::=
```

```
    SEQUENCE {
```

```
        frCircuitIfIndex      InterfaceIndex,
```

```
        frCircuitDlci        DLCI,
```

```
        frCircuitState      INTEGER,
```

```
        frCircuitReceivedFECNs Counter32,
```

```
        frCircuitReceivedBECNs Counter32,
```



```
frCircuitSentFrames          Counter32,
frCircuitSentOctets          Counter32,
frCircuitReceivedFrames     Counter32,
frCircuitReceivedOctets     Counter32,
frCircuitCreationTime       TimeStamp,
frCircuitLastTimeChange     TimeStamp,
frCircuitCommittedBurst     Integer32,
frCircuitExcessBurst        Integer32,
frCircuitThroughput         Integer32,
frCircuitMulticast          INTEGER,
frCircuitType                INTEGER,
frCircuitDiscards           Counter32,
frCircuitReceivedDEs        Counter32,
frCircuitSentDEs            Counter32,
frCircuitLogicalIfIndex     InterfaceIndex,
frCircuitRowStatus          RowStatus
}
```

frCircuitIfIndex OBJECT-TYPE

SYNTAX InterfaceIndex

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The ifIndex Value of the ifEntry this virtual circuit
is layered onto."

::= { frCircuitEntry 1 }

frCircuitDlci OBJECT-TYPE

SYNTAX DLCI

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The Data Link Connection Identifier for this virtual
circuit."

REFERENCE

"American National Standard T1.618-1991, [Section 3.3.6](#)"

::= { frCircuitEntry 2 }

frCircuitState OBJECT-TYPE

SYNTAX INTEGER {
invalid (1),
active (2),


```
        inactive (3)
    }
MAX-ACCESS    read-create
STATUS        current
DESCRIPTION
```

"Indicates whether the particular virtual circuit is operational. In the absence of a Data Link Connection Management Interface, virtual circuit entries (rows) may be created by setting virtual circuit state to 'active', or deleted by changing Circuit state to 'invalid'.

Whether or not the row actually disappears is left to the implementation, so this object may actually read as 'invalid' for some arbitrary length of time. It is also legal to set the state of a virtual circuit to 'inactive' to temporarily disable a given circuit.

The use of 'invalid' is deprecated in this SNMP Version 2 MIB, in favor of frCircuitRowStatus."

```
DEFVAL { active }
 ::= { frCircuitEntry 3 }
```

frCircuitReceivedFECNs OBJECT-TYPE

```
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
```

"Number of frames received from the network indicating forward congestion since the virtual circuit was created. This occurs when the remote DTE sets the FECN flag, or when a switch in the network enqueues the frame to a trunk whose transmission queue is congested."

REFERENCE

"American National Standard T1.618-1991, [Section 3.3.3](#)"

```
 ::= { frCircuitEntry 4 }
```

frCircuitReceivedBECNs OBJECT-TYPE

```
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
```


"Number of frames received from the network indicating backward congestion since the virtual circuit was created. This occurs when the remote DTE sets the BECN flag, or when a switch in the network receives the frame from a trunk whose transmission queue is congested."

REFERENCE

"American National Standard T1.618-1991, [Section 3.3.4](#)"
::= { frCircuitEntry 5 }

frCircuitSentFrames OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of frames sent from this virtual circuit since it was created."

::= { frCircuitEntry 6 }

frCircuitSentOctets OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of octets sent from this virtual circuit since it was created. Octets counted are the full frame relay header and the payload, but do not include the flag characters or CRC."

::= { frCircuitEntry 7 }

frCircuitReceivedFrames OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Number of frames received over this virtual circuit since it was created."

::= { frCircuitEntry 8 }

frCircuitReceivedOctets OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only
STATUS current
DESCRIPTION
"Number of octets received over this virtual circuit since it was created. Octets counted include the full frame relay header, but do not include the flag characters or the CRC."
::= { frCircuitEntry 9 }

frCircuitCreationTime OBJECT-TYPE

SYNTAX TimeStamp
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The value of sysUpTime when the virtual circuit was created, whether by the Data Link Connection Management Interface or by a SetRequest."
::= { frCircuitEntry 10 }

frCircuitLastTimeChange OBJECT-TYPE

SYNTAX TimeStamp
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The value of sysUpTime when last there was a change in the virtual circuit state"
::= { frCircuitEntry 11 }

frCircuitCommittedBurst OBJECT-TYPE

SYNTAX Integer32
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"This variable indicates the maximum amount of data, in bits, that the network agrees to transfer under normal conditions, during the measurement interval."
REFERENCE
"American National Standard T1.617-1991, [Section 6.5.19](#)"
DEFVAL { 0 } -- the default indicates no commitment
::= { frCircuitEntry 12 }

frCircuitExcessBurst OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"This variable indicates the maximum amount of uncommitted data bits that the network will attempt to deliver over the measurement interval.

By default, if not configured when creating the entry, the Excess Information Burst Size is set to the value of ifSpeed."

REFERENCE

"American National Standard T1.617-1991, [Section 6.5.19](#)"

```
::= { frCircuitEntry 13 }
```

frCircuitThroughput OBJECT-TYPE

SYNTAX Integer32

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"Throughput is the average number of 'Frame Relay Information Field' bits transferred per second across a user network interface in one direction, measured over the measurement interval.

If the configured committed burst rate and throughput are both non-zero, the measurement interval
 $T = \text{frCircuitCommittedBurst} / \text{frCircuitThroughput}$.

If the configured committed burst rate and throughput are both zero, the measurement interval
 $T = \text{frCircuitExcessBurst} / \text{ifSpeed}$."

REFERENCE

"American National Standard T1.617-1991, [Section 6.5.19](#)"

```
DEFVAL {0} -- the default value of Throughput is  
-- "no commitment".
```

```
::= { frCircuitEntry 14 }
```

frCircuitMulticast OBJECT-TYPE

```
SYNTAX INTEGER {  
    unicast (1),
```



```
        oneWay    (2),
        twoWay    (3),
        nWay      (4)
    }
MAX-ACCESS    read-create
STATUS        current
DESCRIPTION
    "This indicates whether this VC is used as a unicast VC
    (i.e. not multicast) or the type of multicast service
    subscribed to"
REFERENCE
    "Frame Relay PVC Multicast Service and Protocol
    Description Implementation: FRF.7 Frame Relay Forum
    Technical Committee October 21, 1994"
    DEFVAL {unicast} -- the default value of frCircuitMulticast is
                    -- "unicast" (not a multicast VC).
    ::= { frCircuitEntry 15 }
```

frCircuitType OBJECT-TYPE

```
SYNTAX        INTEGER    {
                    static (1),
                    dynamic (2)
                }
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
    "Indication of whether the VC was manually created
    (static), or dynamically created (dynamic) via the data
    link control management interface."
    ::= { frCircuitEntry 16 }
```

frCircuitDiscards OBJECT-TYPE

```
SYNTAX        Counter32
MAX-ACCESS    read-only
STATUS        current
DESCRIPTION
    "The number of inbound frames dropped because of format
    errors, or because the VC is inactive."
    ::= { frCircuitEntry 17 }
```

frCircuitReceivedDEs OBJECT-TYPE

```
SYNTAX        Counter32
```


MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Number of frames received from the network indicating that they were eligible for discard since the virtual circuit was created. This occurs when the remote DTE sets the DE flag, or when in remote DTE's switch detects that the frame was received as Excess Burst data."

REFERENCE

"American National Standard T1.618-1991, [Section 3.3.4](#)"
 ::= { frCircuitEntry 18 }

frCircuitSentDEs OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Number of frames sent to the network indicating that they were eligible for discard since the virtual circuit was created. This occurs when the local DTE sets the DE flag, indicating that during Network congestion situations those frames should be discarded in preference of other frames sent without the DE bit set."

REFERENCE

"American National Standard T1.618-1991, [Section 3.3.4](#)"
 ::= { frCircuitEntry 19 }

frCircuitLogicalIfIndex OBJECT-TYPE

SYNTAX InterfaceIndex

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"Normally the same value as frDlcmiIfIndex, but different when an implementation associates a virtual ifEntry with a DLC or set of DLCs in order to associate higher layer objects such as the ipAddrEntry with a subset of the virtual circuits on a Frame Relay interface. The type of such ifEntries is defined by the higher layer object; for example, if PPP/Frame Relay is implemented, the ifType of this ifEntry would be PPP. If it is not so defined, as would be the case with an


```
    ipAddrEntry, it should be of type Other."  
 ::= { frCircuitEntry 20 }
```

```
frCircuitRowStatus OBJECT-TYPE
```

```
    SYNTAX      RowStatus
```

```
    MAX-ACCESS  read-create
```

```
    STATUS      current
```

```
    DESCRIPTION
```

```
        " This object is used to create a new row or modify or  
        destroy an existing row in the manner described in the  
        definition of the RowStatus textual convention."
```

```
 ::= { frCircuitEntry 21 }
```

```
--
```

-- Error Table

-- The table describing errors encountered on each Frame
-- Relay Interface.

frErrTable OBJECT-TYPE

SYNTAX SEQUENCE OF FrErrEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A table containing information about Errors on the
Frame Relay interface."

::= { frameRelayDTE 3 }

frErrEntry OBJECT-TYPE

SYNTAX FrErrEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The error information for a single frame relay
interface."

INDEX { frErrIfIndex }

::= { frErrTable 1 }

FrErrEntry ::=

SEQUENCE {

frErrIfIndex InterfaceIndex,

frErrType INTEGER,

frErrData OCTET STRING,

frErrTime TimeStamp,

frErrFaults Counter32,

frErrFaultTime TimeStamp

}

frErrIfIndex OBJECT-TYPE

SYNTAX InterfaceIndex

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The ifIndex Value of the corresponding ifEntry."

::= { frErrEntry 1 }

frErrType OBJECT-TYPE

```

SYNTAX  INTEGER  {
    unknownError(1),
    receiveShort(2),
    receiveLong(3),
    illegalAddress(4),
    unknownAddress(5),
    dlcmiProtoErr(6),
    dlcmiUnknownIE(7),
    dlcmiSequenceErr(8),
    dlcmiUnknownRpt(9),
    noErrorSinceReset(10)
}

```

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The type of error that was last seen on this interface:

receiveShort: frame was not long enough to allow demultiplexing - the address field was incomplete, or for virtual circuits using Multiprotocol over Frame Relay, the protocol identifier was missing or incomplete.

receiveLong: frame exceeded maximum length configured for this interface.

illegalAddress: address field did not match configured format.

unknownAddress: frame received on a virtual circuit which was not active or administratively disabled.

dlcmiProtoErr: unspecified error occurred when attempting to interpret link maintenance frame.

dlcmiUnknownIE: link maintenance frame contained an Information Element type which is not valid for the configured link maintenance protocol.

dlcmiSequenceErr: link maintenance frame contained a sequence number number other than the expected value.

dlcmiUnknownRpt: link maintenance frame contained a Report Type Information Element whose value was not valid for the configured link maintenance protocol.

noErrorSinceReset: no errors have been detected since the last
cold start or warm start."
 ::= { frErrEntry 2 }

frErrData OBJECT-TYPE

SYNTAX OCTET STRING (SIZE(1..1600))

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"An octet string containing as much of the error packet
as possible. As a minimum, it must contain the Q.922
Address or as much as was delivered. It is desirable
to include all header and demultiplexing information."

::= { frErrEntry 3 }

frErrTime OBJECT-TYPE

SYNTAX TimeStamp

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The value of sysUpTime at which the error was
detected."

::= { frErrEntry 4 }

frErrFaults OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of times the interface has gone down since
it was initialized."

::= { frErrEntry 5 }

frErrFaultTime OBJECT-TYPE

SYNTAX TimeStamp

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The value of sysUpTime at the time when the interface
was taken down due to excessive errors. Excessive
errors is defined as the time when a DLCMI exceeds the


```
    frDlcmiErrorThreshold number of errors within
    frDlcmiMonitoredEvents. See FrDlcmiEntry for further
    details."
 ::= { frErrEntry 6 }
--
-- Frame Relay Trap Control

frameRelayTrapControl OBJECT IDENTIFIER ::= { frameRelayDTE 4 }
frameRelayTraps OBJECT IDENTIFIER ::= { frameRelayDTE 5 }

frTrapState OBJECT-TYPE
    SYNTAX INTEGER { enabled(1), disabled(2) }
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION
        "This variable indicates whether the system produces
        the frDLCIStatusChange trap."
    DEFVAL { disabled }
    ::= { frameRelayTrapControl 1 }

frTrapMaxRate OBJECT-TYPE
    SYNTAX INTEGER (0..3600000)
    MAX-ACCESS read-write
    STATUS current
    DESCRIPTION
        "This variable indicates the number of milliseconds
        that must elapse between trap emissions. If events
        occur more rapidly, the impementation may simply fail
        to trap, or may queue traps until an appropriate time."
    DEFVAL { 0 } -- no minimum elapsed period is specified
    ::= { frameRelayTrapControl 2 }

-- Data Link Connection Management Interface Related Traps

frDLCIStatusChange NOTIFICATION-TYPE
    OBJECTS { frCircuitState }
    STATUS current
    DESCRIPTION
        "This trap indicates that the indicated Virtual Circuit
        has changed state. It has either been created or
        invalidated, or has toggled between the active and
        inactive states. If, however, the reason for the state
```



```
        change is due to the DLCMI going down, per-DLCI traps
        should not be generated."
 ::= { frameRelayTraps 1 }
 -- conformance information

frConformance OBJECT IDENTIFIER ::= { frameRelayDTE 6 }

frGroups      OBJECT IDENTIFIER ::= { frConformance 1 }
frCompliances OBJECT IDENTIFIER ::= { frConformance 2 }

 -- compliance statements

frCompliance MODULE-COMPLIANCE
    STATUS current
    DESCRIPTION
        "The compliance statement "
    MODULE -- this module
    MANDATORY-GROUPS { frPortGroup, frCircuitGroup, frErrGroup }
    GROUP          frTrapGroup
    DESCRIPTION
        "This group is optional, and is used for the management
        of asynchronous notifications by Frame Relay Systems."

    OBJECT          frDlcmiRowStatus
    MIN-ACCESS      read-only
    DESCRIPTION
        "Row creation is not required for the frDlcmiTable."

    OBJECT          frCircuitRowStatus
    MIN-ACCESS      read-only
    DESCRIPTION
        "Row creation is not required for the frCircuitTable."

 ::= { frCompliances 1 }

 -- units of conformance

frPortGroup    OBJECT-GROUP
    OBJECTS {
        frDlcmiIfIndex, frDlcmiState, frDlcmiAddress ,
        frDlcmiAddressLen, frDlcmiPollingInterval,
        frDlcmiFullEnquiryInterval, frDlcmiErrorThreshold,
        frDlcmiMonitoredEvents, frDlcmiMaxSupportedVCs,
        frDlcmiMulticast, frDlcmiStatus, frDlcmiRowStatus
    }
}
```



```
STATUS current
DESCRIPTION
    "The objects necessary to control the Link Management
    Interface for a Frame Relay Interface as well as
    maintain the error statistics on this interface."
 ::= { frGroups 1 }

frCircuitGroup    OBJECT-GROUP
OBJECTS {
    frCircuitIfIndex, frCircuitDlci, frCircuitState,
    frCircuitReceivedFECNs, frCircuitReceivedBECNs,
    frCircuitSentFrames, frCircuitSentOctets,
    frCircuitReceivedFrames, frCircuitReceivedOctets,
    frCircuitCreationTime, frCircuitLastTimeChange,
    frCircuitCommittedBurst, frCircuitExcessBurst,
    frCircuitThroughput, frCircuitMulticast, frCircuitType,
    frCircuitDiscards, frCircuitReceivedDEs,
    frCircuitSentDEs, frCircuitLogicalIfIndex,
    frCircuitRowStatus
}
STATUS current
DESCRIPTION
    "The objects necessary to control the Virtual Circuits
    layered onto a Frame Relay Interface."
 ::= { frGroups 2 }

frTrapGroup      OBJECT-GROUP
OBJECTS { frTrapState, frTrapMaxRate }
STATUS current
DESCRIPTION
    "The objects necessary to control a Frame Relay
    Interface's notification messages."
 ::= { frGroups 3 }

frErrGroup       OBJECT-GROUP
OBJECTS {
    frErrIfIndex, frErrType, frErrData, frErrTime,
    frErrFaults, frErrFaultTime
}
STATUS current
DESCRIPTION
    "Objects designed to assist in debugging Frame Relay
    Interfaces."
 ::= { frGroups 4 }
```


Draft

Frame Relay DTE MIB

December 1996

END

6. Security Issues

Security issues for this MIB are entirely covered by the SNMP Security Architecture, and have not been expanded within the contents of this MIB.

7. Acknowledgments

This document was originally produced by the IP Over Large Public Data Networks (IPLPDN) Working Group, and has since been carried on in the PPP Working Group, sort of.

Special thanks to James Watt of Newbridge Networks for his fine suggestions and pastable text.

8. Authors' Addresses

Caralyn Brown
Cadia Networks, Inc.
1 Corporate Dirve
Andover, Massachusetts 01810
Telephone: +1 508 689 2400 x133
E-Mail: cbrown@cadia.com

Fred Baker
Cisco Systems
519 Lado Drive
Santa Barbara, California 93111
Telephone +1 408 526 4257
E-Mail: fred@cisco.com

9. References

- [1] SNMPv2 Working Group, Case, J., McCloghrie, K., Rose, M., and S. Waldbusser, "Structure of Management Information for Version 2 of the Simple Network Management Protocol (SNMPv2)", [RFC 1902](#), January 1996.
- [2] McCloghrie, K., and M. Rose, Editors, "Management Information Base for Network Management of TCP/IP-based internets: MIB-II", STD 17, [RFC 1213](#), Hughes LAN Systems, Performance Systems International, March 1991.
- [3] Case, J., Fedor, M., Schoffstall, M., and J. Davin. " A Simple Network Management Protocol (SNMP)", STD 15, [RFC 1157](#), SNMP Research, Performance Systems International, MIT Lab for Computer Science, May 1990.
- [4] SNMPv2 Working Group, 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.
- [5] McCloghrie, K. and F. Kastenholz, "Evolution of the Interfaces Group of MIB-II", [RFC 1573](#), Hughes LAN Systems, FTP Software, January 1994.
- [6] T. Bradley, C. Brown, A. Malis, "Multiprotocol Interconnect over Frame Relay", [RFC 1490](#), 07/26/1993.
- [7] International Telegraph and Telephone Consultative Committee, "ISDN Data Link Layer Specification for Frame Mode Bearer Services", CCITT Recommendation Q.922, 19 April 1991.
- [8] American National Standard For Telecommunications - Integrated Services Digital Network - Frame Relay Bearer Service - Architectural Framework and Service Description, ANSI T1.606-1991, 18 June 1991.
- [9] American National Standard For Telecommunications - Integrated Services Digital Network - Digital Subscriber Signalling System No. 1 - Signaling Specification for Frame Relay Bearer Service, ANSI T1.617-1991, 18 June 1991.

- [10] American National Standard For Telecommunications -
Integrated Services Digital Network - Core Aspects of
Frame Protocol for Use with Frame Relay Bearer Service,
ANSI T1.618-1991, 18 June 1991.

Table of Contents

1	Status of this Memo	1
2	Abstract	2
3	The SNMPv2 Network Management Framework	2
4	Overview	4
4.1	Frame Relay Operational Model	4
4.2	Textual Conventions	8
4.3	Structure of MIB	8
5	Definitions	9
5.1	Data Link Connection Management Interface	11
5.2	Circuit Table	16
5.3	Error Table	25
5.4	Trap Management	28
6	Security Issues	32
7	Acknowledgments	32
8	Authors' Addresses	32
9	References	33

