Network Working Group
Request for Comments: 2127
Category: Standards Track

G. Roeck, Editor
cisco Systems

March 1997

# ISDN Management Information Base using SMIv2

## Status of this Memo



## Abstract

This memo defines a portion of the Management Information Base (MIB) for use with network management protocols in the Internet community. In particular, it defines a minimal set of managed objects for SNMP-based management of ISDN terminal interfaces. ISDN interfaces are supported on a variety of equipment (for data and voice) including terminal adapters, bridges, hosts, and routers.

This document specifies a MIB module in a manner that is compliant to the SNMPv2 SMI. The set of objects is consistent with the SNMP framework and existing SNMP standards.

This document is a product of the ISDN MIB working group within the Internet Engineering Task Force. Comments are solicited and should be addressed to the working group's mailing list at isdn-mib@cisco.com and/or the author.

The current version of this document reflects changes made during the last call period and the IESG review.

## Table of Contents

## 1. The SNMPv2 Network Management Framework

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

- the SMI, described in RFC 1902 [1] - the mechanisms used for describing and naming objects for the purpose of management.
- the MIB-II, STD 17, RFC 1213 [2] - the core set of managed objects for the Internet suite of protocols.
- the protocol, STD 15, RFC 1157 [3] and/or RFC 1905 [4], - the protocol for accessing managed objects.

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

## 2. Object Definitions

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. Objects in the MIB are defined using the subset of Abstract Syntax Notation One (ASN.1)

defined in the SMI. In particular, each object type is named by an OBJECT IDENTIFIER, 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, we often use a textual string, termed the descriptor, to refer to the object type.

# 3. Overview

## 3.1. Structure of the MIB

For managing ISDN interfaces, the following information is necessary:

- Information for managing physical interfaces. In case of ISDN primary rate, this are usually T1 or E1 lines, being managed in the DS1/E1 MIB [12]. For Basic Rate lines, physical interfaces are managed by this MIB.
- Information for managing B channels.
- Information for managing signaling channels.
- Optionally, information for managing Terminal Endpoints (TE). A Terminal Endpoint is a link layer connection to a switch.
- Optionally, information for managing a list of directory numbers.

In order to manage connections over ISDN lines, the management of peer information and call history information is required as well. This information is defined in the Dial Control MIB [15].

The purpose for splitting the required information in two MIBs is to be able to use parts of this information for non-ISDN interfaces as well. In particular, the Dial Control MIB might also be used for other types of interfaces, e.g. modems or X.25 virtual connections.

Within this document, information has been structured into five groups, which are described in the following chapters.

### 3.1.1. General Description

This MIB controls all aspects of ISDN interfaces. It consists of five groups.

- The isdnMibBasicRateGroup is used to provide information regarding physical Basic Rate interfaces.
- The isdnMibBearerGroup is used to control B (bearer) channels.

It supports configuration parameters as well as statistical information related to B channels.

- The isdnMibSignalingGroup is used to control D (delta) channels. There are three tables in this group. The isdnSignalingTable and isdnSignalingStatsTable support ISDN Network Layer configuration and statistics. The isdnLapdTable supports ISDN Data Link Layer (LAPD) configuration and statistics.

- The optional isdnMibEndpointGroup can be used to specify Terminal Endpoints. It is required only if there are non-ISDN endpoints defined for a given D channel, or if additional information like Terminal Endpoint Identifier (TEI) values or Service Profile IDentifiers (SPID) is required to identify a given ISDN user.

- The optional isdnMibDirectoryGroup can be used to specify a list of directory numbers for each signaling channel. It is required only if the directory numbers to be accepted differ from the isdnSignalingCallingAddress as specified in the isdnSignalingTable.

## 3.2. Relationship to the Interfaces MIB

This section clarifies the relationship of this MIB to the Interfaces MIB [11]. Several areas of correlation are addressed in the following subsections. The implementor is referred to the Interfaces MIB document in order to understand the general intent of these areas.

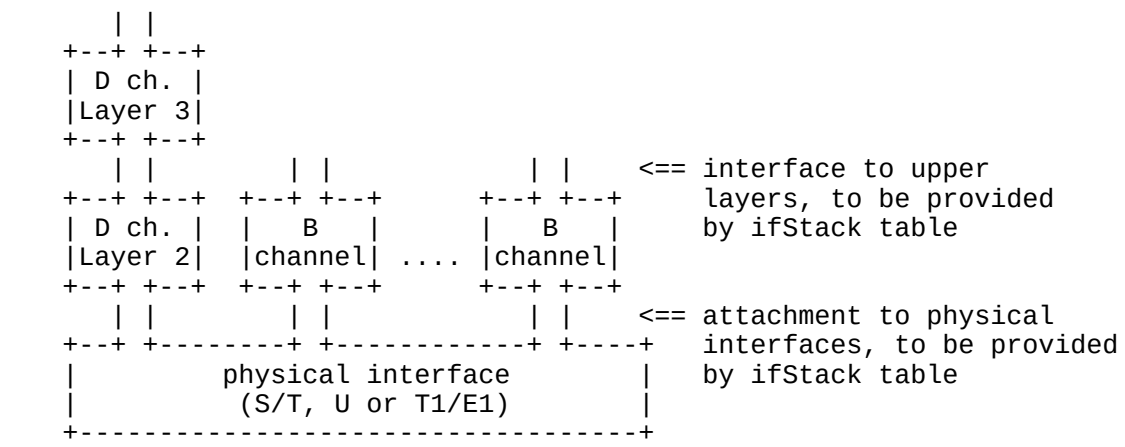
### 3.2.1. Layering Model

An ISDN interface usually consists of a D channel and a number of B channels, all of which are layered on top of a physical interface.

Furthermore, there are multiple interface layers for each D channel. There are Data Link Layer (LAPD) as well as Network Layer entities.

This is accomplished in this MIB by creating a logical interface (ifEntry) for each of the D channel entities and a logical interface (ifEntry) for each of the B channels. These are then correlated to each other and to the physical interface using the ifStack table of the Interfaces MIB [11].

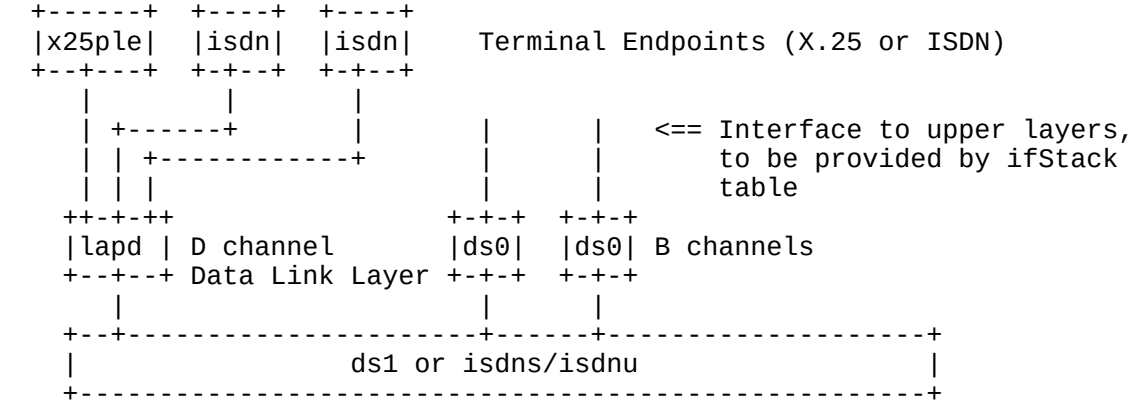
The basic model, therefore, looks something like this:




Mapping of B/D channels to physical interfaces

Each D channel can support multiple Terminal Endpoints. Terminal Endpoints can either be one or multiple ISDN signaling channels, or channels supporting X.25 based packet mode services.

To accomplish this, there can be multiple Network Layer entities on top of each ISDN Data Link Layer (LAPD) interface. The detailed model therefore looks something like this, including interface types as examples:




Detailed interface mapping

IfEntries are maintained for each D channel Network Layer entity (Terminal Endpoint), for LAPD and for each B channel.

The ifType for a Terminal Endpoint can be isdn(63) for ISDN signaling channels or x25ple(40) for X.25 based packet mode services. The ifType for D channel Data Link Layer (LAPD) interfaces is lapd(77). The ifType for B channels is ds0(81). The ifType for physical interfaces is the matching IANA ifType, usually ds1(18) for Primary Rate interfaces or isdns(75)/isdnu(76) for Basic Rate interfaces.

The ifStackTable is used to map B channels and LAPD interfaces to physical interfaces and to map D channel Network Layer interfaces (Terminal Endpoints) to LAPD.

In the example given above, the assignment of index values could for example be as follows:

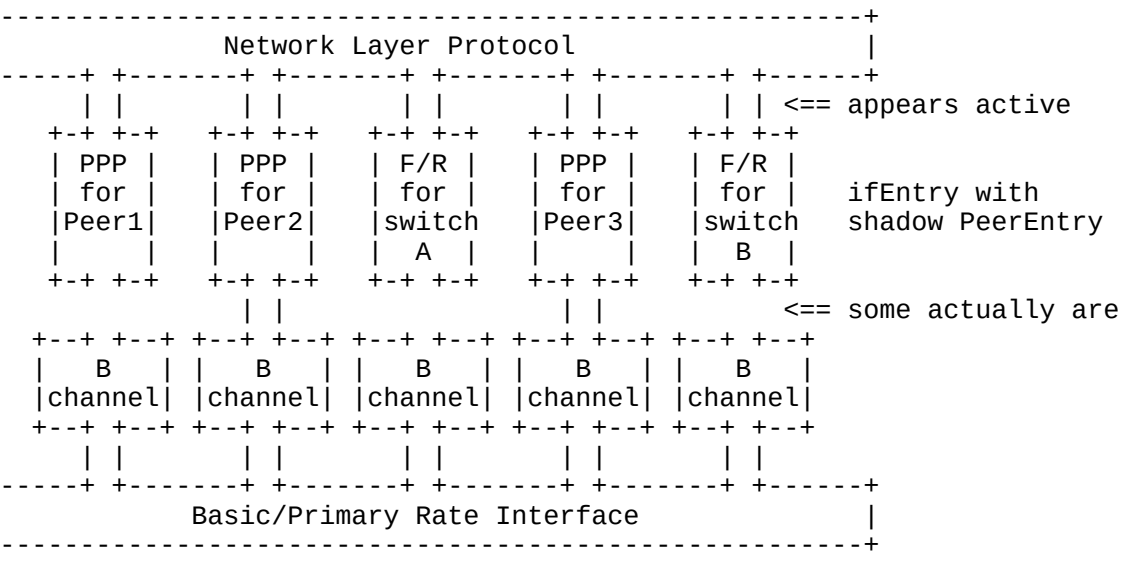
| ifIndex | ifType | ISDN MIB tables indexed by ifIndex | Description |
|---|---|---|---|
| 1 | isdns(75) | isdnBasicRateTable | Basic Rate physical interface |
| 2 | lapd(77) | isdnLapdTable | LAPD interface |
| 3 | x25ple(40) | isdnEndpointTable | X.25 Packet Layer |
| 4 | isdn(63) | isdnSignalingTable<br>isdnEndpointTable | ISDN signaling channel #1 |
| 5 | isdn(63) | isdnSignalingTable<br>isdnEndpointTable | ISDN signaling channel #2 |
| 6 | ds0(81) | isdnBearerTable | B channel #1 |
| 7 | ds0(81) | isdnBearerTable | B channel #2 |
| 8 | ppp(23) | | peer entry #1 (see below) |
| 9 | ppp(23) | | peer entry #2 (see below) |

The corresponding ifStack table entries would then be:

ifStackTable Entries

| HigherLayer | LowerLayer |
|---|---|
| 0 | 3 |
| 0 | 4 |
| 0 | 5 |
| 0 | 8 |
| 0 | 9 |
| 1 | 0 |
| 2 | 1 |
| 3 | 2 |
| 4 | 2 |
| 5 | 2 |
| 6 | 1 |
| 7 | 1 |
| 8 | 6 |
| 9 | 7 |

Mapping of B channels to upper interface layers is usually done using the Dial Control MIB. For example, mapping on top of B channels might look as follows:




Mapping of IP interfaces to Called Peers to B Channels

In this model, ifEntries are maintained for each peer. Each peer is required to have an associated ifEntry. This interface can be of any kind, e.g. PPP or LAPB.

The Dial Control MIB can be used for all types of demand-access interfaces, e.g., ISDN, modems or X.25 virtual connections.

#### 3.2.2. ifTestTable

The ifTestTable is not supported by this MIB.

#### 3.2.3. ifRcvAddressTable

The ifRcvAddressTable is not supported by this MIB.

#### 3.2.4. ifEntry

##### 3.2.4.1. ifEntry for a Basic Rate hardware interface

The ifGeneralGroup is supported for Basic Rate hardware interfaces.

| ifTable | Comments |
|---|---|
| ifIndex | Each ISDN Basic Rate hardware interface is represented by an ifEntry. |
| ifDescr | Textual port description. |
| ifType | The IANA value of isdns(75) or isdnu(76), whichever is appropriate. |
| ifSpeed | The overall bandwidth of this interface. |
| ifPhysAddress | Return an empty string. |
| ifAdminStatus | The administrative status of the ISDN interface. |
| ifOperStatus | The current operational status of this interface. The operational status is dormant(5) if the interface is in standby mode, i.e. connected to the network, but without call activity. The operational status is down(2) if the hardware has detected that there is no layer 1 connection to the switch. For other values, refer to the Interfaces MIB. |
| ifLastChange | Refer to the Interfaces MIB. |

| | |
|---|---|
| ifLinkUpDownTrapEnable | Refer to the Interfaces MIB. |
| ifConnectorPresent | Refer to the Interfaces MIB. |
| ifHighSpeed | Return zero. |
| ifName | Refer to the Interfaces MIB. |

#### 3.2.4.2. ifEntry for a B channel

The ifEntry for a B channel supports the ifGeneralGroup of the Interfaces MIB.

| ifTable | Comments |
|---|---|
| ifIndex | Each ISDN B channel is represented by an ifEntry. |
| ifDescr | Textual port description. |
| ifType | The IANA value of ds0(81). |
| ifSpeed | The bandwidth of this B channel.<br>Usually, this is the value of 56000 or 64000. |
| ifPhysAddress | Return an empty string. |
| ifAdminStatus | The administrative status of this interface. |
| ifOperStatus | The current operational status of this interface.<br>Note that dormant(5) is explicitly being used<br>as defined in the Interfaces MIB.<br>For other values, refer to the Interfaces MIB. |
| ifLastChange | Refer to the Interfaces MIB. |
| ifLinkUpDownTrapEnable | Refer to the Interfaces MIB. |
| ifConnectorPresent | Refer to the Interfaces MIB. |
| ifHighSpeed | Return zero. |
| ifName | Refer to the Interfaces MIB. |

#### 3.2.4.3. ifEntry for LAPD (D channel Data Link Layer)

The ifEntry for LAPD (D channel Data Link Layer) supports the ifGeneralGroup and the ifPacketGroup of the Interfaces MIB.

| ifTable | Comments |
|---|---|
| ifIndex | Each ISDN D channel Data Link layer is represented by an ifEntry. |
| ifDescr | Textual port description. |
| ifType | The IANA value of lapd(77). |
| ifSpeed | The bandwidth of this interface. Usually, this is the value of 16000 for basic rate interfaces or 64000 for primary rate interfaces. |
| ifPhysAddress | Return an empty string. |
| ifAdminStatus | The administrative status of this interface. |
| ifOperStatus | The current operational status of the ISDN LAPD interface. The operational status is dormant(5) if the interface is in standby mode (see Q.931 [8], Annex F, D channel backup procedures). For other values, refer to the Interfaces MIB. |
| ifLastChange | Refer to the Interfaces MIB. |
| ifLinkUpDownTrapEnable | Refer to the Interfaces MIB. |
| ifConnectorPresent | Refer to the Interfaces MIB. |
| ifHighSpeed | Return zero. |
| ifName | Refer to the Interfaces MIB. |
| ifMtu | The size of the largest frame which can be sent/received on this interface, specified in octets. Usually, this is the default value of 260 as specified in Q.921 [6], chapter 5.9.3. |

| | |
|---|---|
| ifInOctets | The total number of octets received on this interface. |
| ifInUcastPkts | The number of frames received on this interface whose address is not TEI=127. |
| ifInNUcastPkts | Deprecated. Return the number of frames received on this interface with TEI=127. |
| ifInMulticastPkts | Return zero. |
| ifInBroadcastPkts | Return the number of frames received on this interface with TEI=127. |
| ifInDiscards | The total number of received frames which have been discarded. The possible reasons are: buffer shortage. |
| ifInErrors | The number of inbound frames that contained errors preventing them from being deliverable to LAPD. |
| ifInUnknownProtos | The number of frames with known TEI, but unknown SAPI (Service Access Point Identifier, see Q.921 [6], chapter 3.3.3). |
| ifOutOctets | The total number of octets transmitted on this interface. |
| ifOutUcastPkts | The number of frames transmitted on this interface whose address is not TEI=127. |
| ifOutNUcastPkts | Deprecated. Return the number of frames transmitted on this interface with TEI=127. |
| ifOutMulticastPkts | Return zero. |
| ifOutBroadcastPkts | Return the number of frames transmitted on this interface with TEI=127. |
| ifOutDiscards | The total number of outbound frames which were discarded. Possible reasons are: buffer shortage. |
| ifOutErrors | The number of frames which could not be transmitted due to errors. |

| | |
|---|---|
| ifOutQlen | Deprecated. Return zero. |
| ifSpecific | Deprecated. Return {0 0}. |

3.2.4.4. ifEntry for a signaling channel

The ifEntry for a signaling channel supports the ifGeneralGroup and the ifPacketGroup of the Interfaces MIB.

| ifTable | Comments |
|---|---|
| ifIndex | Each ISDN signaling channel is represented by an ifEntry. |
| ifDescr | Textual port description. |
| ifType | The IANA value of isdn(63). |
| ifSpeed | The bandwidth of this signaling channel. Usually, this is the same value as for LAPD, i.e. 16000 for basic rate interfaces or 64000 for primary rate interfaces. |
| ifPhysAddress | The ISDN address assigned to this signaling channel. This is a copy of isdnSignalingCallingAddress. |
| ifAdminStatus | The administrative status of the signaling channel. |
| ifOperStatus | The current operational status of this signaling channel. The operational status is dormant(5) if the signaling channel is currently not activated. For other values, refer to the Interfaces MIB. |
| ifLastChange | Refer to the Interfaces MIB. |
| ifLinkUpDownTrapEnable | Refer to the Interfaces MIB. |
| ifConnectorPresent | Refer to the Interfaces MIB. |
| ifHighSpeed | Return zero. |
| ifName | Refer to the Interfaces MIB. |

| | |
|---|---|
| ifMtu | The size of the largest frame which can be sent/received on this signaling channel, specified in octets. Usually, this is the default value of 260 as specified in Q.921 [6], chapter 5.9.3. |
| ifInOctets | The total number of octets received on this signaling channel. |
| ifInUcastPkts | The number of frames received which are targeted to this channel. |
| ifInNUcastPkts | Deprecated. Return the number of frames received on this signaling channel with TEI=127. |
| ifInMulticastPkts | Return zero. |
| ifInBroadcastPkts | Return the number of frames received on this signaling channel with TEI=127. |
| ifInDiscards | The total number of received frames which have been discarded. The possible reasons are: buffer shortage. |
| ifInErrors | The number of inbound frames that contained errors preventing them from being deliverable to the signaling channel. |
| ifInUnknownProtos | Return zero. |
| ifOutOctets | The total number of octets transmitted on this signaling channel. |
| ifOutUcastPkts | The number of frames transmitted on this signaling channel whose address is not TEI=127. |
| ifOutNUcastPkts | Deprecated. Return the number of frames transmitted on this signaling channel with TEI=127. |
| ifOutMulticastPkts | Return zero. |
| ifOutBroadcastPkts | Return the number of frames transmitted on this signaling channel with TEI=127. |

| | |
|---|---|
| ifOutDiscards | The total number of outbound frames which were discarded. Possible reasons are: buffer shortage. |
| ifOutErrors | The number of frames which could not be transmitted due to errors. |
| ifOutQlen | Deprecated. Return zero. |
| ifSpecific | Deprecated. Return {0 0}. |

## 3.3. Relationship to other MIBs

### 3.3.1. Relationship to the DS1/E1 MIB

Implementation of the DS1/E1 MIB [12] is not required for supporting this MIB. It is however recommended to implement the DS1/E1 MIB on entities supporting Primary Rate interfaces.

### 3.3.2. Relationship to the DS0 and DS0Bundle MIBs

Implementation of the DS0 MIB [13] is optional.

Implementation of the DS0Bundle MIB [13] may be required only if hyperchannels are to be supported, depending on the multiplexing scheme used in a given implementation. See chapter 3.4.2 for details on how to implement hyperchannels.

### 3.3.3. Relationship to the Dial Control MIB

Implementation of the Dial Control MIB [15] is required.

## 3.4. ISDN interface specific information and implementation hints

### 3.4.1. ISDN leased lines

ISDN leased lines can be specified on a per-B-channel basis. To do so, the value of isdnBearerChannelType has to be set to leased(2). There is no signaling protocol support for leased line B channels, since there is no signaling protocol action for these kinds of interfaces.

If there is no signaling support available for an ISDN interface, this must be specified in the appropriate interface specific table. For Basic Rate interfaces, isdnBasicRateSignalMode of isdnBasicRateTable must be set to inactive(2). For Primary Rate interfaces, dsx1SignalMode of dsx1ConfigTable in DS1/E1 MIB [12] must be set to none(1). There are no isdnLapdTable or isdnSignalingTable entries for such interfaces.

Depending on the leased line type and the service provider, the D channel can be used for data transfer. If this is the case the D channel interface type is ds0(81) instead of lapd(77) and its usage is identical to B channel usage if there is no signaling channel available.

For a Primary Rate interface which is entirely used as a leased line, there is no ISDN specific information available or required. Such leased lines can entirely be handled by the DS1/E1 MIB.

### 3.4.2. Hyperchannels

The active switch protocol defines if hyperchannels are supported, and the actual support is implementation dependent. Hyperchannel connections will be requested by the interface user at call setup time, e.g. by the peer connection handling procedures.

In the ISDN MIB, the isdnBearerMultirate object of isdnBearerTable can be used to check if hyperchannels are being used for an active call.

If hyperchannels are being used, multiplexing between the encapsulation layer and the B channels is required, since there is one encapsulation layer interface connected to several B channel interfaces. This can be accomplished in two ways.

- The DS0Bundle MIB [13] can be used to provide the multiplexing. See the DS0Bundle MIB document for details.

- The ifStackTable can be used to provide the multiplexing. In this case, there are several ifStackTable entries with the same value of HigherLayer, and different values of LowerLayer.

It is up to the implementor to decide which multiplexing scheme to use.

Each hyperchannel call is treated as one call in the isdnSignalingStatsTable, independent of the number of B channels involved.

For a hyperchannel call, all objects in the isdnBearerTable entries related to this call (i.e., all isdnBearerTable entries associated to B channels used by the hyperchannel) have identical values. The related objects in the isdnBearerTable are:

```
isdnBearerPeerAddress
isdnBearerPeerSubAddress
isdnBearerCallOrigin
isdnBearerInfoType
isdnBearerMultirate
isdnBearerCallSetupTime
isdnBearerCallConnectTime
isdnBearerChargedUnits
```

### 3.4.3. D channel backup and NFAS trunks

D channel backup is defined in Q.931 [8], Annex F. It describes Non-Associated signaling and its use and functionality is basically identical to Non Facility Associated Signaling (NFAS) trunks.

Non Facility Accociated Signaling (NFAS) basically means that a D channel on a PRI interface is used to manage calls on other PRI trunks. This is required in North America for H11 channels, since all 24 time slots are being used for B channels.

According to Q.931, Annex F, the D channel backup feature can be provided on a subscription basis and is network dependent. The D channel backup procedure is described in detail in Q.931.

For D channel backup, the controlling isdnSignalingTable entry is layered on top of all attached LAPD interfaces. This layering is done using the ifStack table. There is only one active LAPD interface, however. Inactive LAPD interfaces have an ifOperStatus of dormant(5).

NFAS trunks are also handled using the ifStack table. In this case, a signaling channel is layered on top of a LAPD interface as well as on top of all physical interfaces which are controlled by the signaling channel, but do not supply a D channel.

### 3.4.4. X.25 based packet-mode service in B and D channels

X.25 based packet mode service over B channels can be handled using the Dial Control MIB by creating an appropriate peer entry. The peer entry ifType can then be x25(5), thus providing access to X.25 service.

X.25 based packet mode service over D channels can be handled by creating an ifEndpointTable entry with an isdnEndpointIfType of x25ple(40). The upper protocol layers can then be attached to this interface using the ifStack table.

### 3.4.5. SPID handling

Service Profile IDentifiers (SPIDs) are defined for BRI interfaces only, and being used in North America. SPIDs are required for DMS-100, NI-1 and NI-2, and are optional for 5ESS. A switch can define up to 8 SPIDs per BRI.

Each Terminal Endpoint has a SPID assigned. It is normally built from the party number (calling address for outgoing calls) with a number of digits prepended and appended. Since each network appears to be different, both the calling address and the SPID have to be stored.

The SPID identifies the particular services that have been provisioned for a terminal. If there are two B channels on a BRI, there can be two SPIDs, one for each of the two B channels. There can also be a single SPID, providing access to both B channels.

The SPID gets registered with the switch after link establishment. There is one data link for each SPID. As part of terminal registration, an EID (Endpoint IDentifier) is defined by the switch. On incoming calls, the switch may provide the EID, a called party number, or both, depending on the ISDN code implemented in the switch.

The EID has two bytes: USID (User Service IDentifier) and TID (Terminal IDentifier). These are later used by some of the software versions running on the switch side (e.g. compliant with NI-1, 5ESS custom) to broadcast SETUP messages with these included, so the correct endpoint would accept the call. Other switch software versions identify the endpoint with the Called Party Number.

In the ISDN MIB, the SPID can be entered using the isdnEndpointSpid object of isdnEndpointTable. The isdnSignalingCallingAddress, already being used to specify the calling number, cannot be used to record the SPID since the values of the SPID and the Calling Address may differ and both may be required to be present.

### 3.4.6. Closed User Groups

Closed User Groups (CUG), as defined in I.255.1 [14], are supported for circuit mode calls by ETSI (ETS 300 138) and 1TR6. In these networks, an ISDN address can have one or more Closed User Groups

assigned. If there is more than one Closed User Group assigned to a given address, one of those is the preferred Closed User Group. For such addresses, only calls from assigned Closed User Groups are accepted by the network.

Thus, Closed User Groups are a parameter for peer entries and are defined in the Dial Control MIB. A peer entry attached to a Closed User Group has to point to an ISDN interface which is attached to the Closed User Group in question.

### 3.4.7. Provision of point-to-point line topology

In the ISDN standards, there are two different meanings for the term "point-to-point".

In ISDN standards, the term point-to-point are usually used for data link connections, i.e. layer 2 connections, where each layer 2 connection from the TE to the network is a single point-to-point connection. Multiple connections of this kind may exist on one physical (layer 1) connection, however, and in case of Basic Rate interfaces there may be several TE's connected to one physical line to the network.

The second meaning of "point-to-point" refers to the line topology, i.e. to layer 1 connections. For Primary Rate interfaces, the line topology is always point-to-point. For Basic Rate interfaces, layer 1 point-to- point connections do exist in several countries, usually being used for connecting PBX systems to the network.

The second meaning (layer 1 connections) is what will be referred to as "point-to-point" connection throughout this document.

For Basic Rate interfaces, the isdnBasicRateTable object isdnBasicRateLineTopology can be used to select the line topology.

### 3.4.8. Speech and audio bearer capability information elements

The objects speech(2), audio31(6) and audio7(7), as being used in isdnBearerInfoType, refer to the Speech, 3.1 kHz Audio and old 7 kHz Audio (now Multi-use) bearer capabilities for ISDN, as defined in Q.931 [8], chapter 4.5.5, octet 3 of bearer capability information element.

These capabilities are signaling artifices that allow networks to do certain things with the call. It is up to the network to decide what to do.

The Speech Bearer Capability means that speech is being carried over the channel, as in two people talking. This would be POTS-type speech. The network may compress this, encrypt it or whatever it wants with it as long as it delivers POTS quality speech to the other end. In other words, a modem is not guaranteed to work over this connection.

The 3.1 kHz Audio capability indicates that the network carries the 3.1 kHz bandwidth across the network. This would (theoretically) allow modem signals to be carried across the network. In the US, the network automatically enters a capability of 3.1 kHz Audio on calls coming into the ISDN from a POTS network. This capability restricts the network from interfering with the data channel in a way that would corrupt the 3.1 kHz VoiceBand data.

7 kHz Audio was meant to signal the use of a higher quality audio connection (e.g., music from radio). It was changed to Multi-Use capability to allow it to be used for video-conferencing with fall back to audio.

In some cases, the Speech or 3.1 kHz Bearer Capability provides a 56 kbit/s data path through the network. Therefore, some people are setting up calls with the Speech or 3.1 kHz BC and transmitting 56 kbit/s data over the connection. This is usually to take advantage of favorable tariffs for Speech as opposed to Data.

On the incoming side, the equipment is usually configured to ignore the Bearer Capability and either answer all Speech calls as 56 kbit/s data or to use one Directory Number for real speech and another for data.

#### 3.4.9. Attaching incoming calls to router ports

In ISDN, there are several ways to identify an incoming call and to attach a router port to this call.

- The call can be identified and attached to a router port using the ISDN Calling Address, that is, the peer ISDN address. Since the peer address is defined in a Dial Control MIB configuration entry for this peer, this would be the most natural way to attach an incoming call to a router port.

  In this configuration, only a single isdnSignalingTable entry is required for each physical ISDN interface. Unfortunately, the ISDN Calling Address is not available in all countries and/or switch protocols. Therefore, other means for attaching incoming calls to router ports must be provided.

- The call can also be identified and attached to a router port using the ISDN Called Address. In this case, a distinct ISDN address or subaddress must be specified for each of the router ports. This can be accomplished in the ISDN MIB by creating a isdnSignalingTable entry for each of the router ports, and by connecting Dial Control MIB peer entries to the thereby created interface using the dialCtlPeerCfgLowerIf object of dialCtlPeerCfgTable.

  If this type of router port identification is used in an implementation, it is up to the implementor to decide if there should be distinct TEI values assigned for each of the isdnSignalingTable entries. For this reason, the isdnEndpointTable permits specifying the same TEI value in multiple entries. It is recommended to use dynamic TEI assignment whenever possible.

  The implementor should be aware that this type of configuration requires a lot of configuration work for the customer, since an entry in isdnSignalingTable must be created for each of the router ports.

- Incoming calls can also be identified and attached to router ports using a higher layer functionality, such as PPP authentication. Defining this functionality is outside the scope of this document.

### 3.4.10. Usage of isdnMibDirectoryGroup and isdnDirectoryTable

In some switch protocol or PBX implementations, the Called Number Information Element on incoming calls can differ from the Calling Number on outgoing calls. Sometimes, the Called Number can be different for incoming Local Calls, Long Distance Calls and International Calls. For Hunt Groups, the Called Number can be any of the numbers in the Hunt Group.

The isdnDirectoryTable can be used to specify all these numbers.

Entries in the isdnDirectoryTable are always connected to specific isdnSignalingTable entries. No ifEntry is created for isdnDirectoryTable entries. Therefore, the isdnDirectoryTable can not be used to attach incoming calls to router ports. For router port identification, isdnSignalingTable entries should be created instead.

## 4. Definitions

```
ISDN-MIB DEFINITIONS ::= BEGIN

IMPORTS
        MODULE-IDENTITY,
        NOTIFICATION-TYPE,
        OBJECT-TYPE,
        Counter32,
        Gauge32,
        Integer32
                FROM SNMPv2-SMI
        DisplayString,
        TruthValue,
        TimeStamp,
        RowStatus,
        TestAndIncr,
        TEXTUAL-CONVENTION
                 FROM SNMPv2-TC
        MODULE-COMPLIANCE,
        OBJECT-GROUP,
        NOTIFICATION-GROUP
                FROM SNMPv2-CONF
        ifIndex,
        InterfaceIndex
                FROM IF-MIB
        IANAifType
                FROM IANAifType-MIB
        transmission
                FROM RFC1213-MIB;

isdnMib MODULE-IDENTITY
        LAST-UPDATED    "9609231642Z" -- Sep 23, 1996
        ORGANIZATION    "IETF ISDN MIB Working Group"
        CONTACT-INFO
            "        Guenter Roeck
             Postal: cisco Systems
                     170 West Tasman Drive
                     San Jose, CA 95134
                     U.S.A.
             Phone:  +1 408 527 3143
             E-mail: groeck@cisco.com"
        DESCRIPTION
            "The MIB module to describe the
             management of ISDN interfaces."
        ::= { transmission 20 }

-- The ISDN hardware interface (BRI or PRI) is represented
```

```
-- by a media specific ifEntry.
--
-- For basic rate lines, the media specifics for the physical interface
-- is defined in the physical interface group of the ISDN MIB.
-- The ifType for physical basic rate interfaces is isdns(75)
-- or isdnu(76), whichever is appropriate.
--
-- For primary rate, the media specifics are defined in the Trunk
-- MIB and the ifType has a value of ds1(18).

-- Each signaling channel is represented by an entry
-- in the isdnSignalingTable.
-- The signaling channel has an ifType value of isdn(63).
-- Each B channel is also represented as an entry
-- in the ifTable. The B channels have an ifType value
-- of ds0(81).
-- This model is used while defining objects and tables
-- for management.
-- The ISDN MIB allows sub-layers. For example, the data transfer
-- over a B channel may take place with PPP encapsulation. While the
-- ISDN MIB describes the D and B channels, a media specific MIB
-- for PPP can be used on a layered basis. This is as per
-- the interfaces MIB.

-- Textual conventions

IsdnSignalingProtocol ::= TEXTUAL-CONVENTION
        STATUS      current
        DESCRIPTION
            "This data type is used as the syntax of the
             isdnSignalingProtocol object in the
             definition of ISDN-MIB's isdnSignalingTable.

             The definition of this textual convention with the
             addition of newly assigned values is published
             periodically by the IANA, in either the Assigned
             Numbers RFC, or some derivative of it specific to
             Internet Network Management number assignments.  (The
             latest arrangements can be obtained by contacting the
             IANA.)

             Requests for new values should be made to IANA via
             email (iana@iana.org)."
        SYNTAX      INTEGER {
             other(1),          -- none of the following
             dss1(2),           -- ITU DSS1 (formerly CCITT) Q.931
             etsi(3),           -- Europe / ETSI ETS300-102
                                -- plus supplementary services
```

```
                                -- (ETSI 300-xxx)
                                -- note that NET3, NET5 define
                                -- test procedures for ETS300-102
                                -- and have been replaced by
                                -- I-CTR 3 and I-CTR 4.
                dass2(4),       -- U.K.   / DASS2 (PRI)
                ess4(5),        -- U.S.A. / AT&T 4ESS
                ess5(6),        -- U.S.A. / AT&T 5ESS
                dms100(7),      -- U.S.A. / Northern Telecom DMS100
                dms250(8),      -- U.S.A. / Northern Telecom DMS250
                ni1(9),         -- U.S.A. / National ISDN 1 (BRI)
                ni2(10),        -- U.S.A. / National ISDN 2 (BRI, PRI)
                ni3(11),        -- U.S.A. / next one
                vn2(12),        -- France / VN2
                vn3(13),        -- France / VN3
                vn4(14),        -- France / VN4 (ETSI with changes)
                vn6(15),        -- France / VN6 (ETSI with changes)
                                -- delta document CSE P 10-21 A
                                -- test document  CSE P 10-20 A
                kdd(16),        -- Japan  / KDD
                ins64(17),      -- Japan  / NTT INS64
                ins1500(18),    -- Japan  / NTT INS1500
                itr6(19),       -- Germany/ 1TR6 (BRI, PRI)
                cornet(20),     -- Germany/ Siemens HiCom CORNET
                ts013(21),      -- Australia / TS013
                                -- (formerly TPH 1962, BRI)
                ts014(22),      -- Australia / TS014
                                -- (formerly TPH 1856, PRI)
                qsig(23),       -- Q.SIG
                swissnet2(24),  -- SwissNet-2
                swissnet3(25)   -- SwissNet-3
        }

-- Isdn Mib objects definitions

isdnMibObjects OBJECT IDENTIFIER ::= { isdnMib 1 }

-- ISDN physical interface group

-- This group describes physical basic rate interfaces.

isdnBasicRateGroup      OBJECT IDENTIFIER ::= { isdnMibObjects 1 }

isdnBasicRateTable OBJECT-TYPE
        SYNTAX      SEQUENCE OF IsdnBasicRateEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
```

```
            "Table containing configuration and operational
             parameters for all physical Basic Rate
             interfaces on this managed device."
        ::= { isdnBasicRateGroup 1 }

isdnBasicRateEntry OBJECT-TYPE
        SYNTAX      IsdnBasicRateEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "An entry in the ISDN Basic Rate Table."
        INDEX { ifIndex }
        ::= { isdnBasicRateTable 1 }

IsdnBasicRateEntry ::= SEQUENCE {
            isdnBasicRateIfType         INTEGER,
            isdnBasicRateLineTopology   INTEGER,
            isdnBasicRateIfMode         INTEGER,
            isdnBasicRateSignalMode     INTEGER
        }

isdnBasicRateIfType OBJECT-TYPE
        SYNTAX      INTEGER {
            isdns(75),
            isdnu(76)
        }
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
            "The physical interface type. For 'S/T' interfaces,
             also called 'Four-wire Basic Access Interface',
             the value of this object is isdns(75).
             For 'U' interfaces, also called 'Two-wire Basic
             Access Interface', the value of this object is
             isdnu(76)."
        ::= { isdnBasicRateEntry 1 }

isdnBasicRateLineTopology OBJECT-TYPE
        SYNTAX      INTEGER {
            pointToPoint(1),
            pointToMultipoint(2)
        }
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
            "The line topology to be used for this interface.
             Note that setting isdnBasicRateIfType to isdns(75)
             does not necessarily mean a line topology of
```

```
             point-to-multipoint."
        ::= { isdnBasicRateEntry 2 }

isdnBasicRateIfMode OBJECT-TYPE
        SYNTAX      INTEGER {
            te(1),
            nt(2)
        }
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
            "The physical interface mode. For TE mode, the value
             of this object is te(1). For NT mode, the value
             of this object is nt(2)."
        ::= { isdnBasicRateEntry 3 }

isdnBasicRateSignalMode OBJECT-TYPE
        SYNTAX      INTEGER {
            active(1),
            inactive(2)
        }
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
            "The signaling channel operational mode for this interface.
             If active(1) there is a signaling channel on this
             interface. If inactive(2) a signaling channel is
             not available."
        ::= { isdnBasicRateEntry 4 }

-- The B channel (bearer channel) group

-- Note that disconnects can explicitely be handled using the
-- ifStack table. If a connection is to be disconnected,
-- the according ifStack entry has to be removed.
-- More specifically, the ifStackTable entry which binds the high-layer
-- ifTable entry (and related dialCtlNbrCfgTable entry) to the
-- B channel ifTable entry (and related isdnBearerTable entry)
-- during an active call has to be removed.

isdnBearerGroup OBJECT IDENTIFIER ::= { isdnMibObjects 2 }

isdnBearerTable OBJECT-TYPE
        SYNTAX      SEQUENCE OF IsdnBearerEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "This table defines port specific operational, statistics
```

```
             and active call data for ISDN B channels. Each entry
             in this table describes one B (bearer) channel."
        ::= { isdnBearerGroup 1 }

isdnBearerEntry OBJECT-TYPE
        SYNTAX      IsdnBearerEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "Operational and statistics information relating to
             one port. A port is a single B channel."
        INDEX { ifIndex }
        ::= { isdnBearerTable 1 }

IsdnBearerEntry ::=
        SEQUENCE {
            isdnBearerChannelType           INTEGER,
            isdnBearerOperStatus            INTEGER,
            isdnBearerChannelNumber         INTEGER,
            isdnBearerPeerAddress           DisplayString,
            isdnBearerPeerSubAddress        DisplayString,
            isdnBearerCallOrigin            INTEGER,
            isdnBearerInfoType              INTEGER,
            isdnBearerMultirate             TruthValue,
            isdnBearerCallSetupTime         TimeStamp,
            isdnBearerCallConnectTime       TimeStamp,
            isdnBearerChargedUnits          Gauge32
        }

isdnBearerChannelType           OBJECT-TYPE
        SYNTAX INTEGER {
            dialup(1),
            leased(2)
        }
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
            "The B channel type. If the B channel is connected
             to a dialup line, this object has a value of
             dialup(1). In this case, it is controlled by
             an associated signaling channel. If the B channel
             is connected to a leased line, this object has
             a value of leased(2). For leased line B channels, there
             is no signaling channel control available."
        ::= { isdnBearerEntry 1 }

isdnBearerOperStatus            OBJECT-TYPE
        SYNTAX INTEGER {
```

```
            idle(1),
            connecting(2),
            connected(3),
            active(4)
        }
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The current call control state for this port.
             idle(1):       The B channel is idle.
                            No call or call attempt is going on.
             connecting(2): A connection attempt (outgoing call)
                            is being made on this interface.
             connected(3):  An incoming call is in the process
                            of validation.
             active(4):     A call is active on this interface."
        ::= { isdnBearerEntry 2 }

isdnBearerChannelNumber         OBJECT-TYPE
        SYNTAX INTEGER (1..30)
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The identifier being used by a signaling protocol
             to identify this B channel, also referred to as
             B channel number. If the Agent also supports the DS0 MIB,
             the values of isdnBearerChannelNumber and dsx0Ds0Number
             must be identical for a given B channel."
        ::= { isdnBearerEntry 3 }

isdnBearerPeerAddress           OBJECT-TYPE
        SYNTAX   DisplayString
        MAX-ACCESS   read-only
        STATUS   current
        DESCRIPTION
            "The ISDN address the current or last call is or was
             connected to.

             In some cases, the format of this information can not
             be predicted, since it largely depends on the type
             of switch or PBX the device is connected to. Therefore,
             the detailed format of this information is not
             specified and is implementation dependent.

             If possible, the agent should supply this information
             using the E.164 format. In this case, the number must
             start with '+'. Otherwise, IA5 number digits must be used.
```

```
             If the peer ISDN address is not available,
             this object has a length of zero."
        REFERENCE
            "ITU-T E.164, Q.931 chapter 4.5.10"
        ::= { isdnBearerEntry 4 }

isdnBearerPeerSubAddress        OBJECT-TYPE
        SYNTAX   DisplayString
        MAX-ACCESS   read-only
        STATUS   current
        DESCRIPTION
            "The ISDN subaddress the current or last call is or was
             connected to.

             The subaddress is an user supplied string of up to 20
             IA5 characters and is transmitted transparently through
             the network.

             If the peer subaddress is not available, this object
             has a length of zero."
        REFERENCE
             "ITU-T I.330, Q.931 chapter 4.5.11"
        ::= { isdnBearerEntry 5 }

isdnBearerCallOrigin            OBJECT-TYPE
          SYNTAX   INTEGER {
            unknown(1),
            originate(2),
            answer(3),
            callback(4)
        }
        MAX-ACCESS   read-only
        STATUS   current
        DESCRIPTION
            "The call origin for the current or last call. If since
             system startup there was no call on this interface,
             this object has a value of unknown(1)."
        ::= { isdnBearerEntry 6 }

isdnBearerInfoType              OBJECT-TYPE
        SYNTAX      INTEGER {
            unknown(1),
            speech(2),
            unrestrictedDigital(3),       -- as defined in Q.931
            unrestrictedDigital56(4),     -- with 56k rate adaption
            restrictedDigital(5),
            audio31(6),                   -- 3.1 kHz audio
            audio7(7),                    -- 7 kHz audio
```

```
            video(8),
            packetSwitched(9)
        }
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The Information Transfer Capability for the current
             or last call.

             speech(2) refers to a non-data connection, whereas
             audio31(6) and audio7(7) refer to data mode connections.

             Note that Q.931, chapter 4.5.5, originally defined
             audio7(7) as '7 kHz audio' and now defines it as
             'Unrestricted digital information with tones/
             announcements'.

             If since system startup there has been no call on this
             interface, this object has a value of unknown(1)."
        REFERENCE
            "Q.931 [8], chapter 4.5.5, octet 3 of bearer capability
             information element, combined with the User Rate
             (as defined in octets 5 and 5a to 5d), if rate adaption
             is being used."
        ::= { isdnBearerEntry 7 }

isdnBearerMultirate             OBJECT-TYPE
        SYNTAX      TruthValue
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "This flag indicates if the current or last call used
             multirate. The actual information transfer rate,
             in detail specified in octet 4.1 (rate multiplier),
             is the sum of all B channel ifSpeed values for
             the hyperchannel.

             If since system startup there was no call on this
             interface, this object has a value of false(2)."
        REFERENCE
            "Q.931 [8], chapter 4.5.5."
        ::= { isdnBearerEntry 8 }

isdnBearerCallSetupTime         OBJECT-TYPE
        SYNTAX      TimeStamp
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
```

```
            "The value of sysUpTime when the ISDN setup message for
             the current or last call was sent or received. If since
             system startup there has been no call on this interface,
             this object has a value of zero."
        ::= { isdnBearerEntry 9 }

isdnBearerCallConnectTime       OBJECT-TYPE
        SYNTAX      TimeStamp
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The value of sysUpTime when the ISDN connect message for
             the current or last call was sent or received. If since
             system startup there has been no call on this interface,
             this object has a value of zero."
        ::= { isdnBearerEntry 10 }

isdnBearerChargedUnits          OBJECT-TYPE
        SYNTAX      Gauge32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of charged units for the current or last
             connection. For incoming calls or if charging information
             is not supplied by the switch, the value of this object
             is zero."
        ::= { isdnBearerEntry 11 }

-- ISDN signaling group

isdnSignalingGroup OBJECT IDENTIFIER ::= { isdnMibObjects 3 }

-- signaling channel configuration table
-- There is one entry in this table for each Terminal Endpoint
-- (link layer connection to the switch).
-- Usually, there is one endpoint per D channel. In some
-- cases, however, there can be multiple endpoints.
-- Thus, entries in this table can be created and deleted.
-- This also means the creation of an associated ifEntry.
--
-- D channel backup and NFAS trunks are handled using the
-- ifStack table.
-- In case of D channel backup, there are multiple
-- Data Link Layer (LAPD) interfaces. Only one interface is
-- active; all others are dormant(5).
-- In case of NFAS trunks, one lower interface is the
-- LAPD interface, while the other lower interfaces are physical
-- interfaces.
```

```
-- If directory number and calling address differ from each other
-- or multiple directory numbers are being used,
-- the isdnDirectoryTable has to be used to enter such
-- directory numbers.

isdnSignalingGetIndex OBJECT-TYPE
        SYNTAX      TestAndIncr
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
            "The recommended procedure for selecting a new index for
             isdnSignalingTable row creation is to GET the value of
             this object, and then to SET the object with the same
             value. If the SET operation succeeds, the manager can use
             this value as an index to create a new row in this table."
        REFERENCE
            "RFC1903, TestAndIncr textual convention."
        ::= { isdnSignalingGroup 1 }

isdnSignalingTable OBJECT-TYPE
        SYNTAX      SEQUENCE OF IsdnSignalingEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "ISDN signaling table containing configuration and
             operational parameters for all ISDN signaling
             channels on this managed device."
        ::= { isdnSignalingGroup 2 }

isdnSignalingEntry OBJECT-TYPE
        SYNTAX      IsdnSignalingEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "An entry in the ISDN Signaling Table. To create a new
             entry, only isdnSignalingProtocol needs to be specified
             before isdnSignalingStatus can become active(1)."
        INDEX { isdnSignalingIndex }
        ::= { isdnSignalingTable 1 }

IsdnSignalingEntry ::= SEQUENCE {
            isdnSignalingIndex          INTEGER,
            isdnSignalingIfIndex        InterfaceIndex,
            isdnSignalingProtocol       IsdnSignalingProtocol,
            isdnSignalingCallingAddress DisplayString,
            isdnSignalingSubAddress     DisplayString,
            isdnSignalingBchannelCount  Integer32,
            isdnSignalingInfoTrapEnable INTEGER,
```

```
            isdnSignalingStatus         RowStatus
        }

isdnSignalingIndex OBJECT-TYPE
        SYNTAX      INTEGER (1..2147483647)
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "The index value which uniquely identifies an entry
             in the isdnSignalingTable."
        ::= { isdnSignalingEntry 1 }

isdnSignalingIfIndex OBJECT-TYPE
        SYNTAX      InterfaceIndex
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The ifIndex value of the interface associated with this
             signaling channel."
        ::= { isdnSignalingEntry 2 }

isdnSignalingProtocol OBJECT-TYPE
        SYNTAX      IsdnSignalingProtocol
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
            "The particular protocol type supported by the
             switch providing access to the ISDN network
             to which this signaling channel is connected."
        ::= { isdnSignalingEntry 3 }

isdnSignalingCallingAddress OBJECT-TYPE
        SYNTAX      DisplayString
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
            "The ISDN Address to be assigned to this signaling
             channel. More specifically, this is the 'Calling Address
             information element' as being passed to the switch
             in outgoing call setup messages.

             It can be an EAZ (1TR6), a calling number (DSS1, ETSI)
             or any other number necessary to identify a signaling
             interface. If there is no such number defined or required,
             this is a zero length string. It is represented in
             DisplayString form.

             Incoming calls can also be identified by this number.
```

```
             If the Directory Number, i.e. the Called Number in
             incoming calls, is different to this number, the
             isdnDirectoryTable has to be used to specify all
             possible Directory Numbers.

             The format of this information largely depends on the type
             of switch or PBX the device is connected to. Therefore,
             the detailed format of this information is not
             specified and is implementation dependent.

             If possible, the agent should implement this information
             using the E.164 number format. In this case, the number
             must start with '+'. Otherwise, IA5 number digits must
             be used."
         REFERENCE
             "ITU-T E.164, Q.931 chapter 4.5.10"
         DEFVAL { "" }
         ::= { isdnSignalingEntry 4 }

isdnSignalingSubAddress OBJECT-TYPE
         SYNTAX      DisplayString
         MAX-ACCESS  read-create
         STATUS      current
         DESCRIPTION
             "Supplementary information to the ISDN address assigned
              to this signaling channel. Usually, this is the
              subaddress as defined in Q.931.
              If there is no such number defined or required, this is
              a zero length string.
              The subaddress is used for incoming calls as well as
              for outgoing calls.
              The subaddress is an user supplied string of up to 20
              IA5 characters and is transmitted transparently through
              the network."
         REFERENCE
             "ITU-T I.330, Q.931 chapter 4.5.11"
         DEFVAL { "" }
         ::= { isdnSignalingEntry 5 }

isdnSignalingBchannelCount OBJECT-TYPE
         SYNTAX      Integer32 (1..65535)
         MAX-ACCESS  read-create
         STATUS      current
         DESCRIPTION
             "The total number of B channels (bearer channels)
              managed by this signaling channel. The default value
              of this object depends on the physical interface type
              and is either 2 for Basic Rate interfaces or
```

```
             24 (30) for Primary Rate interfaces."
        ::= { isdnSignalingEntry 6 }

isdnSignalingInfoTrapEnable    OBJECT-TYPE
        SYNTAX      INTEGER {
            enabled(1),
            disabled(2)
        }
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
            "Indicates whether isdnMibCallInformation traps
             should be generated for calls on this signaling
             channel."
        DEFVAL      { disabled }
        ::= { isdnSignalingEntry 7 }

isdnSignalingStatus OBJECT-TYPE
        SYNTAX      RowStatus
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
            "This object is used to create and delete rows in the
             isdnSignalingTable."
        ::= { isdnSignalingEntry 8 }

-- Signaling channel statistics table
-- There is one entry for each signaling connection
-- in this table.
-- Note that the ifEntry also has some statistics information.

isdnSignalingStatsTable OBJECT-TYPE
        SYNTAX      SEQUENCE OF IsdnSignalingStatsEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "ISDN signaling table containing statistics
             information for all ISDN signaling channels
             on this managed device.
             Only statistical information which is not already being
             counted in the ifTable is being defined in this table."
        ::= { isdnSignalingGroup 3 }

isdnSignalingStatsEntry OBJECT-TYPE
        SYNTAX      IsdnSignalingStatsEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
```

```
            "An entry in the ISDN Signaling statistics Table."
        AUGMENTS { isdnSignalingEntry }
        ::= { isdnSignalingStatsTable 1 }

IsdnSignalingStatsEntry ::= SEQUENCE {
            isdnSigStatsInCalls         Counter32,
            isdnSigStatsInConnected     Counter32,
            isdnSigStatsOutCalls        Counter32,
            isdnSigStatsOutConnected    Counter32,
            isdnSigStatsChargedUnits    Counter32
        }

isdnSigStatsInCalls OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of incoming calls on this interface."
        ::= { isdnSignalingStatsEntry 1 }

isdnSigStatsInConnected OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of incoming calls on this interface
             which were actually connected."
        ::= { isdnSignalingStatsEntry 2 }

isdnSigStatsOutCalls OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of outgoing calls on this interface."
        ::= { isdnSignalingStatsEntry 3 }

isdnSigStatsOutConnected OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of outgoing calls on this interface
             which were actually connected."
        ::= { isdnSignalingStatsEntry 4 }

isdnSigStatsChargedUnits OBJECT-TYPE
        SYNTAX      Counter32
```

```
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of charging units on this interface since
             system startup.
             Only the charging units applying to the local interface,
             i.e. for originated calls or for calls with 'Reverse
             charging' being active, are counted here."
        ::= { isdnSignalingStatsEntry 5 }

--
-- The LAPD table

isdnLapdTable OBJECT-TYPE
        SYNTAX      SEQUENCE OF IsdnLapdEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "Table containing configuration and statistics
             information for all LAPD (D channel Data Link)
             interfaces on this managed device.
             Only statistical information which is not already being
             counted in the ifTable is being defined in this table."
        ::= { isdnSignalingGroup 4 }

isdnLapdEntry OBJECT-TYPE
        SYNTAX      IsdnLapdEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "An entry in the LAPD Table."
        INDEX { ifIndex }
        ::= { isdnLapdTable 1 }

IsdnLapdEntry ::= SEQUENCE {
            isdnLapdPrimaryChannel  TruthValue,
            isdnLapdOperStatus      INTEGER,
            isdnLapdPeerSabme       Counter32,
            isdnLapdRecvdFrmr       Counter32
        }

isdnLapdPrimaryChannel OBJECT-TYPE
        SYNTAX      TruthValue
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
            "If set to true(1), this D channel is the designated
             primary D channel if D channel backup is active.
```

```
             There must be exactly one primary D channel
             configured. If D channel backup is not used, this
             object has a value of true(1)."
        REFERENCE
            "Q.931 [8], Annex F, D channel backup procedures."
        ::= { isdnLapdEntry 1 }

isdnLapdOperStatus OBJECT-TYPE
        SYNTAX      INTEGER {
            inactive(1),
            l1Active(2),
            l2Active(3)
        }
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The operational status of this interface:

             inactive  all layers are inactive
             l1Active  layer 1 is activated,
                       layer 2 datalink not established
             l2Active  layer 1 is activated,
                       layer 2 datalink established."
        ::= { isdnLapdEntry 2 }

isdnLapdPeerSabme OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of peer SABME frames received on this
             interface. This is the number of peer-initiated
             new connections on this interface."
        ::= { isdnLapdEntry 3 }

isdnLapdRecvdFrmr OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The number of LAPD FRMR response frames received.
             This is the number of framing errors on this
             interface."
        ::= { isdnLapdEntry 4 }

--
-- Optional groups follow here.
```

```
-- The Terminal Endpoint group and table

-- This table is required only if TEI values or SPID numbers
-- have to be entered.
-- The ifIndex values for this table are identical to those of
-- the isdnSignalingChannel table.

isdnEndpointGroup OBJECT IDENTIFIER ::= { isdnMibObjects 4 }

isdnEndpointGetIndex OBJECT-TYPE
        SYNTAX      TestAndIncr
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
            "The recommended procedure for selecting a new index for
             isdnEndpointTable row creation is to GET the value of
             this object, and then to SET the object with the same
             value. If the SET operation succeeds, the manager can use
             this value as an index to create a new row in this table."
        REFERENCE
            "RFC1903, TestAndIncr textual convention."
        ::= { isdnEndpointGroup 1 }

isdnEndpointTable OBJECT-TYPE
        SYNTAX      SEQUENCE OF IsdnEndpointEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "Table containing configuration for Terminal
             Endpoints."
        ::= { isdnEndpointGroup 2 }

isdnEndpointEntry OBJECT-TYPE
        SYNTAX      IsdnEndpointEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "An entry in the Terminal Endpoint Table. The value
             of isdnEndpointIfType must be supplied for a row
             in this table to become active."
        INDEX { isdnEndpointIndex }
        ::= { isdnEndpointTable 1 }

IsdnEndpointEntry ::= SEQUENCE {
            isdnEndpointIndex       INTEGER,
            isdnEndpointIfIndex     InterfaceIndex,
            isdnEndpointIfType      IANAifType,
            isdnEndpointTeiType     INTEGER,
```

```
            isdnEndpointTeiValue    INTEGER,
            isdnEndpointSpid        DisplayString,
            isdnEndpointStatus      RowStatus
        }

isdnEndpointIndex OBJECT-TYPE
        SYNTAX      INTEGER (1..2147483647)
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "The index value which uniquely identifies an entry
             in the isdnEndpointTable."
        ::= { isdnEndpointEntry 1 }

isdnEndpointIfIndex OBJECT-TYPE
        SYNTAX      InterfaceIndex
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
            "The ifIndex value of the interface associated with this
             Terminal Endpoint."
        ::= { isdnEndpointEntry 2 }

isdnEndpointIfType OBJECT-TYPE
        SYNTAX      IANAifType
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
            "The interface type for this Terminal Endpoint.
             Interface types of x25ple(40) and isdn(63) are allowed.
             The interface type is identical to the value of
             ifType in the associated ifEntry."
        ::= { isdnEndpointEntry 3 }

isdnEndpointTeiType OBJECT-TYPE
        SYNTAX      INTEGER {
            dynamic(1),
            static(2)
        }
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
            "The type of TEI (Terminal Endpoint Identifier)
             used for this Terminal Endpoint. In case of dynamic(1),
             the TEI value is selected by the switch. In
             case of static(2), a valid TEI value has to be
             entered in the isdnEndpointTeiValue object.
             The default value for this object depends on the
```

```
             interface type as well as the Terminal Endpoint type.
             On Primary Rate interfaces the default value is
             static(2). On Basic Rate interfaces the default value
             is dynamic(1) for isdn(63) Terminal Endpoints and
             static(2) for x25ple(40) Terminal Endpoints."
        ::= { isdnEndpointEntry 4 }

isdnEndpointTeiValue OBJECT-TYPE
        SYNTAX      INTEGER ( 0..255 )
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
            "The TEI (Terminal Endpoint Identifier) value
             for this Terminal Endpoint. If isdnEndpointTeiType
             is set to static(2), valid numbers are 0..63,
             while otherwise the value is set internally.
             The default value of this object is 0 for static
             TEI assignment.
             The default value for dynamic TEI assignment is also
             0 as long as no TEI has been assigned. After TEI
             assignment, the assigned TEI value is returned."
        ::= { isdnEndpointEntry 5 }

isdnEndpointSpid OBJECT-TYPE
        SYNTAX      DisplayString
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
            "The Service profile IDentifier (SPID) information
             for this Terminal Endpoint.

             The SPID is composed of 9-20 numeric characters.

             This information has to be defined in addition to
             the local number for some switch protocol types,
             e.g. Bellcore NI-1 and NI-2.

             If this object is not required, it is a
             zero length string."
        REFERENCE
            "Bellcore SR-NWT-001953, Generic Guidelines for ISDN
             Terminal Equipment on Basic Access Interfaces,
             Chapter 8.5.1."
        DEFVAL { "" }
        ::= { isdnEndpointEntry 6 }

isdnEndpointStatus OBJECT-TYPE
        SYNTAX      RowStatus
```

```
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
            "This object is used to create and delete rows in the
             isdnEndpointTable."
        ::= { isdnEndpointEntry 7 }

--
-- The Directory Number group
--

isdnDirectoryGroup OBJECT IDENTIFIER ::= { isdnMibObjects 5 }

isdnDirectoryTable OBJECT-TYPE
        SYNTAX      SEQUENCE OF IsdnDirectoryEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "Table containing Directory Numbers."
        ::= { isdnDirectoryGroup 1 }

isdnDirectoryEntry OBJECT-TYPE
        SYNTAX      IsdnDirectoryEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "An entry in the Directory Number Table. All objects
             in an entry must be set for a new row to become active."
        INDEX { isdnDirectoryIndex }
        ::= { isdnDirectoryTable 1 }

IsdnDirectoryEntry ::= SEQUENCE {
            isdnDirectoryIndex      INTEGER,
            isdnDirectoryNumber     DisplayString,
            isdnDirectorySigIndex   INTEGER,
            isdnDirectoryStatus     RowStatus
        }

isdnDirectoryIndex OBJECT-TYPE
        SYNTAX      INTEGER ( 1..'7fffffff'h )
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "The index value which uniquely identifies an entry
             in the isdnDirectoryTable."
        ::= { isdnDirectoryEntry 1 }

isdnDirectoryNumber OBJECT-TYPE
```

```
        SYNTAX      DisplayString
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
            "A Directory Number. Directory Numbers are used
             to identify incoming calls on the signaling
             channel given in isdnDirectorySigIndex.

             The format of this information largely depends on the type
             of switch or PBX the device is connected to. Therefore,
             the detailed format of this information is not
             specified and is implementation dependent.

             If possible, the agent should implement this information
             using the E.164 number format. In this case, the number
             must start with '+'. Otherwise, IA5 number digits must
             be used."
        REFERENCE
            "ITU-T E.164, Q.931 chapter 4.5.10"
        ::= { isdnDirectoryEntry 2 }

isdnDirectorySigIndex OBJECT-TYPE
        SYNTAX      INTEGER (1..2147483647)
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
            "An index pointing to an ISDN signaling channel.
             Incoming calls are accepted on this
             signaling channel if the isdnDirectoryNumber is
             presented as Called Number in the SETUP message."
        ::= { isdnDirectoryEntry 3 }

isdnDirectoryStatus OBJECT-TYPE
        SYNTAX      RowStatus
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
            "This object is used to create and delete rows in the
             isdnDirectoryTable."
        ::= { isdnDirectoryEntry 4 }

-- Traps

isdnMibTrapPrefix OBJECT IDENTIFIER ::= { isdnMib 2 }
isdnMibTraps OBJECT IDENTIFIER ::= { isdnMibTrapPrefix 0 }

isdnMibCallInformation NOTIFICATION-TYPE
        OBJECTS {
```

```
            ifIndex,                    -- isdnBearerTable ifIndex
            isdnBearerOperStatus,
            isdnBearerPeerAddress,
            isdnBearerPeerSubAddress,
            isdnBearerCallSetupTime,
            isdnBearerInfoType,
            isdnBearerCallOrigin
        }
        STATUS      current
        DESCRIPTION
            "This trap/inform is sent to the manager under the
             following condidions:
             - on incoming calls for each call which is rejected for
               policy reasons (e.g. unknown neighbor or access
               violation)
             - on outgoing calls whenever a call attempt is determined
               to have ultimately failed. In the event that call retry
               is active, then this will be after all retry attempts
               have failed.
             - whenever a call connects. In this case, the object
               isdnBearerCallConnectTime should be included in the
               trap.

             Only one such trap is sent in between successful or
             unsuccessful call attempts from or to a single neighbor;
             subsequent call attempts result in no trap.

             If the Dial Control MIB objects dialCtlNbrCfgId and
             dialCtlNbrCfgIndex are known by the entity generating
             this trap, both objects should be included in the trap
             as well. The receipt of this trap with no dial neighbor
             information indicates that the manager must poll the
             callHistoryTable of the Dial Control MIB to see what
             changed."
        ::= { isdnMibTraps 1 }

--
-- conformance information
--

isdnMibConformance OBJECT IDENTIFIER ::= { isdnMib 2 }
isdnMibCompliances OBJECT IDENTIFIER ::= { isdnMibConformance 1 }
isdnMibGroups      OBJECT IDENTIFIER ::= { isdnMibConformance 2 }

-- compliance statements

isdnMibCompliance MODULE-COMPLIANCE
        STATUS      current
```

```
        DESCRIPTION
            "The compliance statement for entities which implement
             the ISDN MIB."
        MODULE       -- this module

-- unconditionally mandatory groups
        MANDATORY-GROUPS {
            isdnMibSignalingGroup,
            isdnMibBearerGroup,
            isdnMibNotificationsGroup
        }

-- conditionally mandatory group
        GROUP       isdnMibBasicRateGroup
        DESCRIPTION
            "The isdnMibBasicRateGroup is mandatory for entities
             supporting ISDN Basic Rate interfaces."

-- optional groups
        GROUP       isdnMibEndpointGroup
        DESCRIPTION
            "Implementation of this group is optional for all systems
             that attach to ISDN interfaces."

        GROUP       isdnMibDirectoryGroup
        DESCRIPTION
            "Implementation of this group is optional for all systems
             that attach to ISDN interfaces."

        OBJECT      isdnBasicRateIfType
        MIN-ACCESS  read-only
        DESCRIPTION
            "It is conformant to implement this object as read-only."

        OBJECT      isdnBasicRateLineTopology
        MIN-ACCESS  read-only
        DESCRIPTION
            "It is conformant to implement this object as read-only."

        OBJECT      isdnBasicRateIfMode
        MIN-ACCESS  read-only
        DESCRIPTION
            "It is conformant to implement this object as read-only."

        OBJECT      isdnBasicRateSignalMode
        MIN-ACCESS  read-only
        DESCRIPTION
            "It is conformant to implement this object as read-only."
```

```
        ::= { isdnMibCompliances 1 }

-- units of conformance

isdnMibBasicRateGroup OBJECT-GROUP
        OBJECTS {
            isdnBasicRateIfType,
            isdnBasicRateLineTopology,
            isdnBasicRateIfMode,
            isdnBasicRateSignalMode
        }
        STATUS      current
        DESCRIPTION
            "A collection of objects required for ISDN Basic Rate
             physical interface configuration and statistics."
        ::= { isdnMibGroups 1 }

isdnMibBearerGroup OBJECT-GROUP
        OBJECTS {
            isdnBearerChannelType,
            isdnBearerOperStatus,
            isdnBearerChannelNumber,
            isdnBearerPeerAddress,
            isdnBearerPeerSubAddress,
            isdnBearerCallOrigin,
            isdnBearerInfoType,
            isdnBearerMultirate,
            isdnBearerCallSetupTime,
            isdnBearerCallConnectTime,
            isdnBearerChargedUnits
        }
        STATUS      current
        DESCRIPTION
            "A collection of objects required for ISDN Bearer channel
             control and statistics."
        ::= { isdnMibGroups 2 }

isdnMibSignalingGroup OBJECT-GROUP
        OBJECTS {
            isdnSignalingGetIndex,
            isdnSignalingIfIndex,
            isdnSignalingProtocol,
            isdnSignalingCallingAddress,
            isdnSignalingSubAddress,
            isdnSignalingBchannelCount,
            isdnSignalingInfoTrapEnable,
            isdnSignalingStatus,
            isdnSigStatsInCalls,
```

```
            isdnSigStatsInConnected,
            isdnSigStatsOutCalls,
            isdnSigStatsOutConnected,
            isdnSigStatsChargedUnits,
            isdnLapdPrimaryChannel,
            isdnLapdOperStatus,
            isdnLapdPeerSabme,
            isdnLapdRecvdFrmr
        }
        STATUS      current
        DESCRIPTION
            "A collection of objects required for ISDN D channel
             configuration and statistics."
        ::= { isdnMibGroups 3 }

isdnMibEndpointGroup OBJECT-GROUP
        OBJECTS {
            isdnEndpointGetIndex,
            isdnEndpointIfIndex,
            isdnEndpointIfType,
            isdnEndpointTeiType,
            isdnEndpointTeiValue,
            isdnEndpointSpid,
            isdnEndpointStatus
        }
        STATUS      current
        DESCRIPTION
            "A collection of objects describing Terminal Endpoints."
        ::= { isdnMibGroups 4 }

isdnMibDirectoryGroup OBJECT-GROUP
        OBJECTS {
            isdnDirectoryNumber,
            isdnDirectorySigIndex,
            isdnDirectoryStatus
        }
        STATUS      current
        DESCRIPTION
            "A collection of objects describing directory numbers."
        ::= { isdnMibGroups 5 }

isdnMibNotificationsGroup NOTIFICATION-GROUP
    NOTIFICATIONS { isdnMibCallInformation }
    STATUS        current
    DESCRIPTION
            "The notifications which a ISDN MIB entity is
             required to implement."
    ::= { isdnMibGroups 6 }
```

```
END
```

## 5. Acknowledgments

This document was produced by the ISDN MIB Working Group. Special thanks is due to the following persons:

Ed Alcoff
Fred Baker
Scott Bradner
Bibek A. Das
Maria Greene
Ken Grigg
Stefan Hochuli
Jeffrey T. Johnson
Glenn Kime
Oliver Korfmacher
Kedar Madineni
Bill Miskovetz
Mike O'Dowd
David M. Piscitello
Lisa A. Phifer
Randy Roberts
Hascall H. Sharp
John Shriver
Robert Snyder
Bob Stewart
Ron Stoughton
James Watt

## 6. 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] ITU-T Recommendation "Digital subscriber Signaling System No. 1 (DSS 1) - ISDN User-Network Interface Data Link Layer - General Aspects Rec. Q.920.

[6] ITU-T Recommendation "Digital subscriber Signaling System No. 1 (DSS 1) - ISDN User-Network Interface - Data Link Layer Specification Rec. Q.921.

[7] ITU-T Recommendation "Digital subscriber Signaling System No. 1 (DSS 1) - ISDN Data Link Layer Specification for Frame Mode Bearer Services (LAPF) Rec. Q.922.

[8] ITU-T Recommendation "Digital subscriber Signaling System No. 1 (DSS 1) - ISDN user-network interface layer 3 specification for basic call control", Rec. Q.931(I.451), March 1993.

[9] ITU-T Recommendation "Generic procedures for the control of ISDN supplementary services ISDN user-network interface layer 3 specification", Rec. Q.932(I.452).

[10] ITU-T Recommendation "Digital subscriber Signaling System No. 1 (DSS 1) - Signaling specification for frame-mode basic call control", Rec. Q.933.

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

[12] Fowler, D., "Definitions of Managed Objects for the DS1/E1/DS2/E2 Interface Types", Work in Progress.

[13] Fowler, D., "Definitions of Managed Objects for the DS0 and DS0Bundle Interface Types", Work in Progress.

[14] ITU-T Recommendation "Integrated Services Digital Network (ISDN) General Structure and Service Capabilities - Closed User Group", Rec. I.255.1.

[15] Roeck, G., "Dial Control Management Information Base", RFC 2128, March 1997.

## 7. Security Considerations

Security issues are not discussed in this memo.

## 8. Author's Address

Guenter Roeck
cisco Systems
170 West Tasman Drive
San Jose, CA 95134
U.S.A.

Phone: +1 408 527 3143
EMail: groeck@cisco.com