The ATM Forum

Technical Committee

LAN Emulation Client Management Specification Version 2.0

AF-LANE-0093.000

October, 1998

The information in this publication is believed to be accurate as of its publication date. Such information is subject to change without notice and The ATM Forum is not responsible for any errors. The ATM Forum does not assume any responsibility to update or correct any information in this publication. Notwithstanding anything to the contrary, neither The ATM Forum nor the publisher make any representation or warranty, expressed or implied, concerning the completeness, accuracy, or applicability of any information contained in this publication. No liability of any kind shall be assumed by The ATM Forum or the publisher as a result of reliance upon any information contained in this publication.

The receipt or any use of this document or its contents does not in any way create by implication or otherwise:

- Any express or implied license or right to or under any ATM Forum member company's patent, copyright, trademark or trade secret rights which are or may be associated with the ideas, techniques, concepts or expressions contained herein; nor
- Any warranty or representation that any ATM Forum member companies will announce any product(s) and/or service(s) related thereto, or if such announcements are made, that such announced product(s) and/or service(s) embody any or all of the ideas, technologies, or concepts contained herein; nor
- Any form of relationship between any ATM Forum member companies and the recipient or user of this document.

Implementation or use of specific ATM standards or recommendations and ATM Forum specifications will be voluntary, and no company shall agree or be obliged to implement them by virtue of participation in The ATM Forum.

The ATM Forum is a non-profit international organization accelerating industry cooperation on ATM technology. The ATM Forum does not, expressly or otherwise, endorse or promote any specific products or services.

NOTE: The user's attention is called to the possibility that implementation of the ATM interoperability specification contained herein may require use of an invention covered by patent rights held by ATM Forum Member companies or others. By publication of this ATM interoperability specification, no position is taken by The ATM Forum with respect to validity of any patent claims or of any patent rights related thereto or the ability to obtain the license to use such rights. ATM Forum Member companies agree to grant licenses under the relevant patents they own on reasonable and nondiscriminatory terms and conditions to applicants desiring to obtain such a license. For additional information contact:

The ATM Forum
Worldwide Headquarters
2570 West El Camino Real, Suite 304
Mountain View, CA 94040-1313
Tel: +1-650-949-6700
Fax: +1-650-949-6705

Acknowledgement

The contributions adopted by the ATM Forum and the assistance of the editing group of the LAN Emulation Sub-Working group were essential to the completion of this specification and its predecessor.

In particular, the editor would like to recognize the following members who made significant contributions to this effort:

Andy Bierman

Bill Ellington

Norm Finn

Andre Fredette

John D. Keene

Jon Lawrence

Faye Ly

Keith McCloghrie

Jason Perreault

Andrew Smith

Matt Squire

Andrew Tylee

The assistance by these members and all who participated in the LAN Emulation Sub-Working Group is appreciated. Special thanks to everyone who hosted and attended the special editing meeting for the first version of the LEC MIB.

Thomas D. Newton, Editor

Contents

1. Introduction..1

1.1 Terminology...1

1.2 References..2

2. Management Functions...3

2.1 Configuration Management...3

2.2 Performance Management..3

2.2.1 Observing the behavior of an ATM emulated LAN...............3

2.2.2 Levels of performance management.....................................4

2.3 Fault Management...5

2.4 Security Management..5

2.5 Accounting Management...5

3. Management Framework...6

3.1 SNMP...6

3.2 RFCs..6

3.3 Requirements for LAN Emulation Hosts...7

3.3.1 Requirements for SNMP..7

3.3.2 Requirements for Other Network Management Protocols...7

3.3.3 Conclusion...7

4. MIB-II and RFC 1573 Support...8

4.1 Emulated Network Interfaces and the Interfaces table........................8

4.2 Interpretations of Interface tables for ATM emulated LANs................10

4.3 AToM MIB support and the Interfaces Stack Table.............................15

4.4 Maximum Data Frame Sizes...16

5. LAN Emulation Client Addresses..17

5.1 MAC Addresses..17

5.2 ATM Addresses..19

6. LAN Emulation Client MIB Description...20

6.1 Motivation...20

6.2 MIB Organization..20

6.3 MIB Groups..21

6.3.1 Configuration Group..21

6.3.2 Status Group...23

6.3.3 Statistics Group...25

6.3.4 Server Connections Group...25

6.3.5 ATM Addresses Group..25

6.3.6 Registered MAC Addresses Group..26

6.3.7 Registered Route Descriptors Group...26

6.3.8 LE_ARP Cache Group - MAC Addresses.....................................26

6.3.9 LE_ARP Cache Group - Route Descriptors...................................27

6.3.10 Index Mapping Group...27

6.3.11 Multicast Forward VCC Group...27

6.3.12 Proxy LE_ARP Response Group – MAC Addresses.........28

6.3.13 Proxy LE_ARP Response Group – Route Descriptors......28

6.3.14 TLV Group...29

6.4 Interaction with ifAdminStatus..29

6.5 Limitations...30

7. LAN Emulation Client MIB Text..31

1. Introduction

One of the most promising networking technologies is ATM. ATM offers high performance, the ability to carry many types of services (data, voice, video), and the ability to carry traffic over both the LAN and the WAN.

LAN Emulation provides a means of running existing applications and protocols over ATM. The LAN Emulation specification defines four types of components:

- LAN Emulation Clients - which use emulated LANs to send and receive data.
- LAN Emulation Servers - which perform control coordination functions, and assist clients in doing address resolution.
- Broadcast and Unknown Servers - which emulate the multicast functions of a shared-media LAN.
- LAN Emulation Configuration Servers - which provide auto-configuration services.

Network management is concerned with the ability to remotely monitor and control networks and networked devices.

This document defines network management for LAN Emulation Clients.

1.1 Terminology

The following acronyms and terminology are used throughout this document.

AAL	ATM Adaptation Layer
ATM	Asynchronous Transfer Mode
BUS	Broadcast and Unknown Server
ELAN	Emulated Local Area Network
LAN	Local Area Network
LANE	LAN Emulation
LE	LAN Emulation
LE_ARP	LAN Emulation Address Resolution Protocol
LEC	LAN Emulation Client
LECID	LAN Emulation Client Identifier
LECS	LAN Emulation Configuration Server
LES	LAN Emulation Server
LUNI	LAN Emulation User-Network Interface
MAC	Medium Access Control

RD	Route Descriptor
RFC	Request For Comment (Document Series)
UNI	User-Network Interface
VCC	Virtual Channel Connection
VCI	Virtual Channel Identifier
VPI	Virtual Path Identifier

1.2 References

- The ATM Forum, *LAN Emulation Over ATM Specification, Version 1.0*.
- The ATM Forum, *LAN Emulation Over ATM Version 2 — LUNI Specification.*
- *RFC 1213*, McCloghrie and Rose, *Management Information Base for Network Management of TCP/IP-based internets: MIB-II.*
- *RFC 1493*, Decker, Langille, Rijsinghani, and McCloghrie, *Definitions of Managed Objects for Bridges.*
- *RFC 1573*, McCloghrie and Kastenholtz, *Evolution of the Interfaces Group of MIB-II.*
- *RFC 1695*, Ahmed and Tesink, *Definitions of Managed Objects for ATM Management.*

2. Management Functions

Network management can be divided into the areas of configuration, performance, fault, security, and accounting management.

2.1 Configuration Management

Configuration management is concerned with setting up network elements for normal operation, finding out their current operational parameters, and keeping track of inventories and network topologies.

This network management specification addresses several common tasks, including

- Identifying all of the LAN Emulation Clients currently set up at a managed device.
- Creating and destroying LAN Emulation Clients, for hosts which support this. (This is like adding network interface cards to a machine, or removing them from a machine.)
- Forcing LAN Emulation Clients to join or leave emulated LANs. (This is like attaching a machine to a traditional LAN, or detaching it from a traditional LAN. Making a client leave an emulated LAN does not destroy the client, but simply returns it to an Initial State, where its configuration can be examined and altered.)
- Examining and changing Initial State Parameters.
- Identifying Configuration, Control, and Multicast VCCs.

It does not address identification of Data Direct VCCs, configuration of LAN Emulation PVCs, or identification of the ATM addresses at each end of a VCC. These are expected to be handled via extensions to other MIBs.

Other areas which affect LEC management are at least partially outside of the scope of the LUNI. They include configuration of LAN Emulation Configuration Servers, fault management at the LAN Emulation Server, and full auto-discovery of emulated LAN topology.

2.2 Performance Management

Performance management is concerned with the quality of service that a system delivers to its users, and the efficiency with which it delivers that service.

2.2.1 Observing the behavior of an ATM emulated LAN

One important component of performance management is the ability to observe the behavior of a network. Several factors make ATM emulated LANs harder to observe than traditional ones.

- Traffic is spread out over many virtual circuits, instead of being concentrated in one physical network segment.

- These virtual circuits may be set up and torn down frequently, presenting a moving target to would-be observers.
- The performance of each virtual circuit may be affected by factors outside the control of the ATM emulated LAN hosts. In particular, these include which switches are congested, and what actions those switches take in response to congestion (cell loss, flow control, etc.).
- In addition to data traffic, there may be a fair amount of LAN Emulation control traffic. It is desirable to monitor this control traffic separately.
- It is an explicit non-goal for ATM Forum LAN Emulation to support promiscuous listeners (hosts who want to listen to all unicast traffic). Thus, managers can't simply attach protocol analyzers to an emulated LAN to find out what is happening.

Given this, the only ways for a network management station to monitor the amount of traffic going to a specific host are

1. To enlist the help of LAN Emulation Clients (LECs) in collecting this information.
2. To collect and aggregate performance statistics about individual virtual circuits.
3. To collect performance statistics at the ATM port level, using the AToM MIB.
4. To eavesdrop on communication between two LAN Emulation hosts. A major disadvantage of this is that the network manager must decide which paths to monitor in advance.

This specification requires LAN Emulation Clients to carry out basic traffic measurements, and to make them available through SNMP MIBs, as described in the following text.

2.2.2 Levels of performance management

There are several potential levels of performance management, including

- Performance management of the MAC layer, where we view an emulated LAN as providing a connectionless packet delivery service.
- Performance management of LUNI traffic, especially LE_ARPs.
- Performance management of individual virtual circuits within an emulated LAN.
- Performance management of the ATM switch network over which an emulated LAN runs.

This specification focuses on the first two levels. While performance management of the switch network is important, it is beyond the proper scope of the LUNI. Furthermore, we recognize the philosophy that

> "The SNMP explicitly minimizes the number and complexity of management functions realized by the management agent itself." - RFC 1157
>
> "This memo strongly recommends that connection-oriented sub-layers not have a conceptual row in the ifTable for each virtual circuit. This avoids the proliferation of conceptual rows, especially those which have considerable redundant information." - RFC 1573

In particular, this specification does **not** require clients to implement elaborate protocol analysis functionality, or to maintain separate traffic counters for each virtual circuit. Vendors may, of course, choose to implement such features in addition to those specified herein.

2.3 Fault Management

Fault management is concerned with the prevention, detection, and correction of problems in an emulated LAN that are caused by the failure of network elements.

At the LAN Emulation Client level,

- The MIB-II object 'ifOperStatus' indicates whether the emulated interface is up or down.
- The RFC 1573 object 'ifLinkUpDownTrapEnable' provides an optional way to enable and disable 'linkUp'/'linkDown' traps. (Because this object's MIN-ACCESS is 'read-only', and its default value is 'disabled', an implementation is not required to implement these traps for the LAN Emulation Client layer.)
- The LEC MIB object 'lecInterfaceState' provides additional information on the state of the LAN Emulation Client.
- The LEC MIB objects 'lecLastFailureRespCode' and 'lecLastFailureState' provide information on the last Configure failure or Join failure.

At the virtual circuit level, operational status, administrative status, and AAL5 error statistics are available for Configuration, Control, and Multicast VCCs, via the LEC MIB Server VCC table and the AToM MIB.

2.4 Security Management

Security management is outside the scope of this document.

2.5 Accounting Management

Accounting management is outside the scope of this document.

3. Management Framework

This chapter defines the network management framework for LAN Emulation Clients.

3.1 SNMP

LAN Emulation Client management is defined using SNMP MIBs. SNMP is an Internet-standard network management framework whose basic concepts are as follows:

> "A network management system contains: several (potentially many) nodes, each with a processing entity, termed an agent, which has access to management instrumentation; at least one management station; and, a management protocol, used to convey management information between the agents and management stations. Operations of the protocol are carried out under an administrative framework which defines both authentication and authorization policies.
>
> Network management stations execute management applications which monitor and control network elements. Network elements are devices such as hosts, routers, terminal servers, etc., which are monitored and controlled through access to their management information.
>
> Management information is viewed as a collection of managed objects, residing in a virtual information store, termed the Management Information Base (MIB). Collections of related objects are defined in MIB modules. These modules are written using a subset of OSI's Abstract Syntax Notation One (ASN.1)." - RFC 1442 (SMI for SNMPv2)

Currently, there are two versions of the SNMP framework. This specification uses SNMPv2 as the basis for defining a LAN Emulation Client MIB, and refers to other MIBs whose published definitions use SNMPv2 conventions. However, this specification does not mandate the use of SNMPv2, as opposed to SNMPv1. RFC 1452 describes how SNMPv1 and SNMPv2 coexist, and *The Simple Times* reports that automatic translations can be obtained by mailing SNMPv2 MIBs to `mib-v2tov1@simple-times.org`.

3.2 RFCs

RFC ("Request for Comment") documents are a standard way of proposing new protocols and MIBs in the Internet community. Some of the most relevant RFCs, for our purposes, are

- RFC 1213, which defines MIB-II, the core set of managed objects for the Internet suite of protocols.
- RFCs 1441 through 1452, which define SNMPv2. These RFCs total more than 400 pages. *The Simple Times* volume 2, number 3 contains a useful road map.
- RFCs 1902 through 1902, which update the SNMPv2 specification. For the purposes of this document, the security/complexity debate reflected in RFCs 1901 and 1910 is irrelevant.
- RFC 1573, which defines the evolution of the Interfaces group of MIB-II.

3.3 Requirements for LAN Emulation Hosts

All LAN Emulation Clients SHOULD be network-manageable, either via the SNMP network management protocol or via some other network management protocol.

3.3.1 Requirements for SNMP

SNMP agents that support LAN Emulation Clients MUST implement:

- The Systems and Interfaces groups of STD 17, RFC 1213 - *MIB II*.
- RFC 1695 - *Definitions of Managed Objects for ATM Management*, according to the conformance statements defined in that RFC. This RFC is better known as the AToM MIB.
- The LAN Emulation Client MIB defined in this specification[1].

These are in addition to any other MIB objects that a particular host may be required to support. For instance, an ATM-to-Ethernet bridge should also support the Bridge MIB.

Optionally, an agent may support

- RFC 1573 - *Evolution of the Interfaces Group of MIB-II.*

In this event, the specifications in RFC 1573 supersede those in MIB-II (for example, an agent which implements RFC 1573 does not need to support 'ifInNUcastPkts' or 'ifOutNUcastPkts').

3.3.2 Requirements for Other Network Management Protocols

Although the use of SNMP is recommended, it is not strictly required. LAN Emulation Client management for other protocols SHOULD be based on the MIBs listed above - especially MIB-II, RFC 1573 and the LAN Emulation Client MIB.

3.3.3 Conclusion

We now define these requirements in more detail, starting with MIB-II and RFC 1573.

[1] A SNMP agent which only supports LANE 1.0 clients may choose to conform with Version 1.0 of this document instead. However, note that several of the changes made here and in the LEC MIB addendum benefit LANE 1.0 clients.

4. MIB-II and RFC 1573 Support

All SNMP agents which support LAN Emulation Clients must implement MIB-II, and are also encouraged to implement RFC 1573. This chapter explains why, and defines interpretations of MIB-II / RFC 1573 as they apply to LAN Emulation Clients.

4.1 Emulated Network Interfaces and the Interfaces table

The goal of LAN Emulation is to present the illusion that one or more ATM ports can be treated as one or more 802.* LAN ports[2]. Real LAN ports have entries in the MIB-II Interfaces table and the RFC 1573 Interface Extensions table. To preserve a consistent management framework, it is highly desirable for each emulated port to have entries in these tables as well.

Each SNMP agent that supports LAN Emulation Clients must support such entries. Thus, the host in Figure 1 would have at least three ifTable entries - one for the physical ATM port, and two for the emulated network interfaces corresponding to its LAN Emulation Clients.

To identify an interface as belonging to an emulated LAN, it must be tagged with one of these ifType constants:

- 'aflane8023(59)', for an emulated LAN that supports the IEEE 802.3 and Ethernet data frame formats.
- 'aflane8025(60)', for an emulated LAN that supports the IEEE 802.5 / Token Ring data frame format.

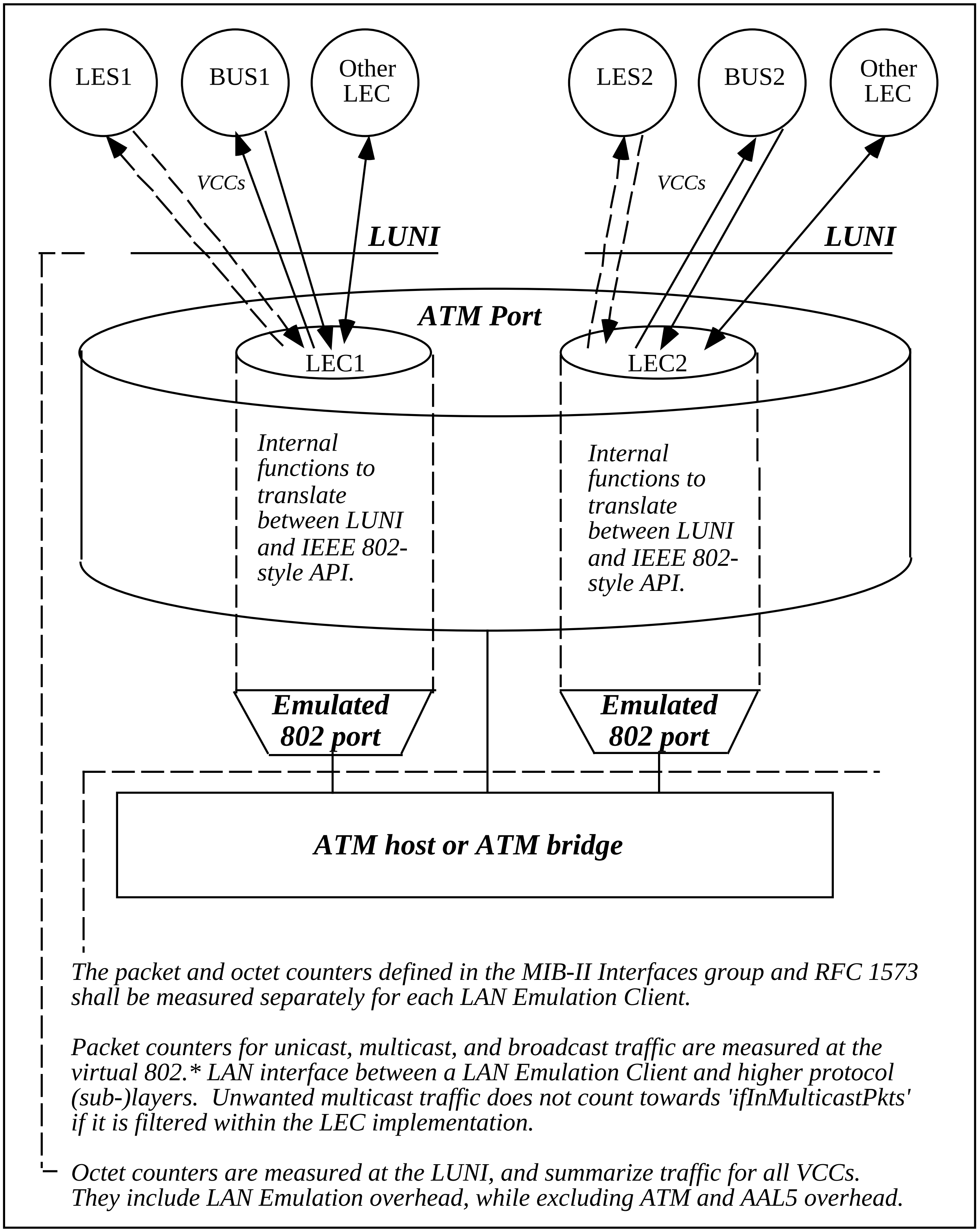
These constants let a network management application know that additional information about the interface is available via the LAN Emulation Client MIB.

Unlike most kinds of interfaces, LAN Emulation Client interfaces can be created and destroyed by network management. So that agents can control ifIndex allocation, most tables in the LEC MIB use a separate lecIndex. For convenience,

- *lecMappingIndex* converts an ifIndex to a lecIndex.
- *lecIfIndex* converts the other way.

[2] The mapping may be, but is not required to be, one-to-one. A single ATM port may support several active LAN Emulation Clients, and a single LAN Emulation Client may employ several ATM ports.

Figure 1 - Emulated Network Interfaces at a LAN Emulation Client Host


LES1
BUS1
Other LEC
LES2
BUS2
Other LEC
VCCs
VCCs
LUNI
LUNI
ATM Port
LEC1
LEC2
Internal functions to translate between LUNI and IEEE 802-style API.
Internal functions to translate between LUNI and IEEE 802-style API.
Emulated 802 port
Emulated 802 port
ATM host or ATM bridge

The packet and octet counters defined in the MIB-II Interfaces group and RFC 1573 shall be measured separately for each LAN Emulation Client.

Packet counters for unicast, multicast, and broadcast traffic are measured at the virtual 802. LAN interface between a LAN Emulation Client and higher protocol (sub-)layers. Unwanted multicast traffic does not count towards 'ifInMulticastPkts' if it is filtered within the LEC implementation.*

Octet counters are measured at the LUNI, and summarize traffic for all VCCs. They include LAN Emulation overhead, while excluding ATM and AAL5 overhead.

4.2 Interpretations of Interface tables for ATM emulated LANs

RFC 1573 "defines a portion of the Management Information Base (MIB) for use with the network management protocols in the Internet community." Specifically, this includes an updated version of the MIB-II 'ifTable' and an extension table called the 'ifXTable'.

The following tables specify interpretations for 'ifTable' and 'ifXTable' objects, as they apply to emulated LANs. Text copied verbatim from RFC 1573 is printed in the `Courier` font, to distinguish it from this document's additions and changes.

Note 1

Although 'ifSpeed' and 'ifHighSpeed' are defined to be 0, decisions as to which of RFC 1573's 64-bit octet and/or packet counters must be implemented must be based on the maximum AAL SDU bandwidth that a host theoretically offers to its LAN Emulation Clients. This is an extra requirement that the LAN Emulation Client MIB imposes above and beyond the conformance statement in RFC 1573, to ensure that LAN Emulation Clients implement the spirit as well as the letter of that MIB.

For instance, a host whose LAN Emulation Clients have access to a single 155 Mb/S SONET interface would be required to implement 64-bit octet counters and 32-bit packet counters, as specified in the compliance statement for 'ifHCPacketGroup'.

Note 2

As a result of people's real-world experience, the interpretation of 'ifInDiscards' has changed. Echo packets and unwanted, filtered multicast packets are now excluded from 'ifInDiscards' and counted in the LEC MIB. This is intended to make the behavior of a LEC interface more like that of real Ethernet and Token Ring hardware adapters, and to prevent false alarms in network management systems which assume that all discards are bad.

Object	Use for the emulated network interface layer
ifIndex	Same interpretation as RFC 1573. A unique value, greater than zero, for each interface. Note that the LAN Emulation Client MIB's 'LeConnectionInterface' textual convention depends upon the assurance that no Interfaces table entry has an 'ifIndex' of zero.
ifDescr	Same interpretation as RFC 1573. A textual string containing information about the interface. This string should include the name of the manufacturer, the product name, and the version of the interface hardware/software.
ifType	If the 'lecActualLanType' for the corresponding LAN Emulation Client is 'unspecified' or 'aflane8023', the IANA ifType constant 'aflane8023(59)'. If the 'lecActualLanType' is 'aflane8025', the IANA ifType constant 'aflane8025(60)'.
ifMtu	This read-only value is a function of the 'lecActualLanType' and 'lecActualMaxDataFrameSize' objects in the LAN Emulation Client MIB. For the purposes of calculating this value, an unspecified maximum frame size is taken to mean the smallest standard size. For a client whose 'lecActualLanType' is 'unspecified' or 'aflane8023', 'ifMtu' is equal to the maximum AAL5 SDU size for a non-multiplexed VCC minus 'X' octets. The value of 'X' is up to the implementation, and may be either 16 (Ethernet encapsulation) or 24 (LLC/SNAP encapsulation). For a client whose 'lecActualLanType' is 'aflane8025', 'ifMtu' is equal to the maximum AAL5 SDU size for a non-multiplexed VCC minus 54 octets.
ifSpeed, ifHighSpeed	Set to 0. Note that decisions as to whether to implement 64-bit octet and/or packet counters must be made on the basis of the maximum AAL5 SDU bandwidth that is theoretically available to any of the managed device's LAN Emulation Clients.
ifPhysAddress	For an active LAN Emulation client, (one of) the MAC address(es) registered for this emulated network interface, stored using the MacAddress convention defined in RFC 1443. For an inactive client which has at least one MAC address, one of those addresses. For an inactive client with no MAC addresses, the empty string. The LUNI lets clients register MAC addresses dynamically. Thus, the value of this object may change over time. However, a client should only change the value of 'ifPhysAddress' at the time that it joins an emulated LAN or deregisters the current 'ifPhysAddress'. Note that the ATM addresses of the LAN Emulation Client and LAN Emulation Server are located in the LEC MIB.

Object	Use for the emulated network interface layer
ifAdminStatus	Provides manual control over Joins and terminations. To make an inactive client join an emulated LAN, set its 'ifAdminStatus' to 'up'. The MIB-II 'ifOperStatus' and LAN Emulation Client MIB 'lecInterfaceState' will reflect the progress and success of the attempt. Once 'ifAdminStatus' has been set to 'up', it will stay 'up' until the manager changes it, or until the managed system changes it as specified in RFC 1573. It will not go 'down' simply because a Join attempt fails or because the client becomes disconnected from an emulated LAN. This makes it easy to distinguish operational problems from intentional termination, ensuring that 'ifAdminStatus' reflects the desired state of the interface. To make a client leave an emulated LAN, set its 'ifAdminStatus' to 'down'. This will cause 'ifOperStatus' to change to 'down', and 'lecInterfaceState' to change to 'initialState'. The 'testing' value is not currently supported for LE Clients.
ifOperStatus	The current operational state of the interface. In particular, the state of the MAC interface between the LAN Emulation Client and higher (sub-)layers, as opposed to the health of the client. 'ifOperStatus' is defined to be 'up' when, and only when, the 'lecInterfaceState' of the LAN Emulation Client is 'operational'. It may take on the values 'down' or 'unknown' at other times. The values 'testing' and 'dormant' are not used.
ifLastChange	Same interpretation as RFC 1573. The value of sysUpTime at the time the interface entered its current operational state. If the current state was entered prior to the last re-initialization of the local network management subsystem, then this object contains a zero value.
ifInOctets, ifOutOctets, and their 64-bit HC counterparts	The total number of PDU octets (received, transmitted) on all of the non-multiplexed VCCs or LLC-multiplexed flows associated with this emulated network interface. This includes octets from circuits (and flows) which have been torn down (or terminated) since this client joined the emulated LAN. Note that these counters are not measured at the emulated network interface between the LEC and higher (sub-)layers. Instead they are measured at the interface between the LEC and the AAL layer. Thus, all received multicast packets and LAN Emulation control packets count towards 'ifInOctets', even if the LEC implementation filters out unwanted multicast packets before they reach higher (sub-)layers.

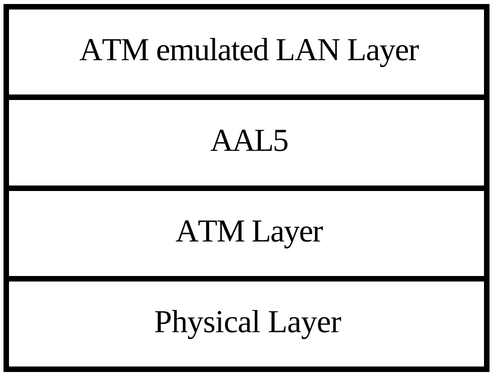
Object	Use for the emulated network interface layer
(continued)	These counts exclude ATM cell layer overhead and AAL overhead. They include LAN Emulation overhead: the LEC-ID, and the LLC header on LLC-multiplexed VCC flows.
Packet counters in general	Traffic counters: Total number of qualifying packets received or transmitted across the emulated network interface between this LAN Emulation Client and higher (sub-)layers. Error counters: Total number of errored PDUs, including control and data PDUs. All counters include packets from VCCs (or flows) torn down (or terminated) since this client joined the emulated LAN. Specific interpretations for each counter follow.
ifInUcastPkts, ifHCInUcastPkts	The number of data packets delivered by this LAN Emulation Client to a higher (sub-)layer, which were not addressed to a multicast or broadcast LAN Destination.
ifInNUcastPkts	Required in MIB-II; deprecated in RFC 1573. The sum of 'ifInMulticastPkts' and 'ifInBroadcastPkts'.
ifInDiscards	The number of inbound LAN Emulation PDU packets which were chosen to be discarded even though no errors had been detected to prevent their being deliverable to a higher-layer protocol or to the control entity in the LAN Emulation Client. One possible reason for discarding such a packet could be to free up buffer space. This count does not include discarded echo packets and filtered multicast packets.
ifInErrors	The number of inbound LAN Emulation PDU packets that contained errors preventing them from being deliverable to a higher-layer protocol or to the control entity in the LAN Emulation Client.
ifInUnknownProtos	The number of LAN Emulation PDU packets this client received via the LUNI which were discarded because of an unknown or unsupported LAN Emulation Control, IEEE 802.3, Ethernet, or IEEE 802.5 protocol.
ifOutUcastPkts, ifHCOutUcastPkts	The total number of data packets that higher-level protocols asked this LAN Emulation Client to transmit, and which were not addressed to a multicast or broadcast LAN Destination, including those that were discarded or not sent.
ifOutNUcastPkts	Required in MIB-II; deprecated in RFC 1573. The sum of 'ifOutMulticastPkts' and 'ifOutBroadcastPkts'.

Object	Use for the emulated network interface layer
ifOutDiscards	The number of outbound LAN Emulation PDU packets which were chosen to be discarded even though no errors had been detected to prevent their being transmitted. One possible reason for discarding such a packet could be to free up buffer space. This count includes all types of LAN Emulation PDUs: Ethernet, 802.3, 802.5, and Control.
ifOutErrors	The number of outbound LAN Emulation PDU packets that could not be transmitted because of errors. This count includes all types of LAN Emulation PDUs: Ethernet, 802.3, 802.5, and Control.
ifOutQLen	Deprecated in RFC 1573. The number of data packets that higher-level protocols have requested this LAN Emulation Client to transmit, and which have not (yet) been discarded or transmitted across the LUNI.
ifSpecific	Deprecated in RFC 1573. Set to OBJECT IDENTIFIER { 0 0 } if present.
ifName	Same interpretation as RFC 1573.
ifInMulticastPkts, ifHCInMulticastPkts	The number of data packets, delivered by this sub-layer to a higher (sub-)layer, which were addressed to a multicast LAN Destination. This includes both Group and Functional MAC addresses. The LANE V1.0 LUNI requires the BUS to forward all broadcast and multicast packets whose ages do not exceed the maximum hold time to all clients. The LANE V2.0 LUNI allows a LAN Emulation Service to offer selective multicasting, but clients still must be able to deal with unwanted multicast frames. LAN Emulation Clients are permitted to filter unwanted, received, multicast frames, but are not required to do so. These counters are measured at the emulated packet interface between the LEC and higher (sub-)layers. Thus, unwanted multicast frames are included in the count if, and only if, the implementation delivers them to a higher (sub-)layer.
ifInBroadcastPkts, ifHCInBroadcastPkts	The number of data packets delivered by this LAN Emulation Client to a higher (sub-)layer, which were addressed to the broadcast MAC address.

Object	**Use for the emulated network interface layer**
ifOutMulticastPkts, ifHCOutMulticastPkts	The total number of data packets that higher-level protocols asked this LAN Emulation Client to transmit, and which were addressed to a multicast LAN Destination, including those that were discarded or not sent. This includes both Group and Functional MAC addresses.
ifOutBroadcastPkts, ifHCOutBroadcastPkts	The total number of data packets that higher-level protocols asked this LAN Emulation Client to transmit, and which were addressed to the broadcast MAC address, including those that were discarded or not sent.
ifLinkUpDownTrapEnable	Default is disabled(2).
ifPromiscuousMode	Set to false(2). LE Clients do not support promiscuous mode in the sense that a network manager or user understands.
ifConnectorPresent	Set to false(2).

4.3 AToM MIB support and the Interfaces Stack Table

Hosts which implement the LAN Emulation Client MIB may also implement RFC 1573. If a client uses AAL5 encapsulation, its Interfaces Stack looks like this:



Note that

- There may be both upwards and downwards multiplexing between the emulated LAN layer and the AAL5 layer. One client may use several ATM ports, and several clients may share an ATM port - perhaps at the same time.

- On hosts with multiple ATM ports, a writeable ifStackTable may be desirable, so a manager can control the assignment of LECs to AAL5 entities and their associated ATM interfaces.
- In a sense, each LAN Emulation Client row in the Interfaces table represents two interfaces: the emulated IEEE 802.* packet interface between the LEC and higher (sub-)layers, and the VCC-oriented interface between the LEC and the AAL5 (sub-)layer. The reason why there is one Interfaces table row per LEC is that the LEC-to-AAL5 interface does not have enough interesting MIB-II-style traffic measurements to justify a separate Interfaces table entry.

4.4 Maximum Data Frame Sizes

'lecActualMaxDataFrameSize' and 'ifMtu' both reflect a client's current maximum frame size. Since 'ifMtu' is the "size of the largest network datagram that can be sent", it depends on the packet format as well. (Octets before the INFO field and octets that are part of LLC/SNAP fields don't count.) For convenience, here is a table summarizing the relationship.

lecActualMaxDataFrameSize	**ifMtu (Ethernet)**	**ifMtu (802.3)**	**ifMtu (802.5)**	**AAL5 SDU max. octets**
unspecified	1500	1492	1462	*n/a*
max1516	1500	1492	1462	1516 / 1528
max1580	1564	1556	1526	1580 / 1592
max4544	4528	4520	4490	4544 / 4556
max9234	9218	9210	9180	9234 / 9246
max18190	18174	18166	18136	18190 / 18202

Note that

- Clients whose LAN Type is unspecified should use either the Ethernet column or the 802.3 column.
- The 1580 octet size is new in LANE Version 2 and is only supported for clients whose "C29 V2 Capable" flag is set.
- The sizes shown on the left side of the AAL5 SDU column are for non-multiplexed VCCs. LANE V2 clients may use multiplexed VCCs as outlined in the LUNI specification. When doing so, they should add 12 octets to the non-multiplexed size to cover the LLC overhead.

5. LAN Emulation Client Addresses

Because they live in both the IEEE 802.* and ATM worlds, LAN Emulation Clients have both MAC and ATM addresses. Furthermore, although MIB-II models interfaces as having a single 'ifPhysAddress', a LEC may have several addresses of each type.

This chapter addresses the question of how to store and configure these addresses.

5.1 MAC Addresses

Each LAN Emulation Client has

- Zero or more unicast MAC addresses which it has registered, or wishes to register.
- Zero or more multicast MAC addresses which it wishes to receive[3].

In keeping with the principle that Interfaces table entries should model the emulated IEEE 802.* packet interface between a LEC and higher protocol (sub-)layers, these addresses are stored and configured using objects defined in MIB-II and RFC 1573. For hosts which implement MIB-II and the LEC MIB,

- 'ifPhysAddress' contains one of a client's local unicast MAC addresses. It follows the RFC 1443 MacAddress convention.
- 'lecMacAddressTable' contains all of a client's registered local unicast MAC addresses and the ATM addresses associated with them. It is a read-only table.

For hosts which implement RFC 1573 and the LEC MIB,

- Implementation of the 'ifRcvAddressGroup' is mandatory. Interpretations of its objects can be found in the table at the end of this section.
- 'ifRcvAddressTable' holds all of a client's local unicast MAC addresses, all of the multicast MAC addresses the client wishes to receive[4], and the broadcast MAC address.
- Write and create access to 'ifRcvAddressTable' are optional. A host may allow such access but restrict it to a subset of rows - for instance, to allow configuration of multicast filters for a client whose MAC address is hardwired.
- Creating, enabling, disabling, or deleting unicast MAC address rows for an operational client will cause it to generate LE_[UN]REGISTER_REQUESTs as appropriate.
- Creating or enabling multicast MAC address rows will ensure that an operational client passes all received frames with those addresses to higher (sub-)layers. Clients MAY filter unwanted frames, but are not required to do so. Unwanted multicast frames count towards 'ifIn[HC]MulticastPkts' if and only if they are "delivered" to a higher (sub-)layer.

[3] Not counting the broadcast MAC address.

[4] This may not be true for all clients. For instance, a bridge LEC which wishes to receive all multicast traffic would not create an ifRcvAddressTable containing every possible multicast MAC address.

Object	**Use for the emulated network interface layer**
ifRcvAddressAddress	One of the 48-bit MAC addresses associated with the corresponding LAN Emulation Client, stored using the MacAddress convention. Rows for unicast MAC addresses denote addresses that clients own or want to own (use as source MAC addresses). Rows for multicast and broadcast MAC addresses denote addresses higher (sub-)layers wish to receive. 'ifRcvAddressTable' must include rows for all of the unicast MAC addresses the client has successfully registered with the LES. This includes 'ifPhysAddress'. This table must also include one row for the broadcast MAC address (present by default).
ifRcvAddressStatus	**Unicast MAC Addresses** For a client whose 'lecInterfaceState' is 'busConnect' or 'operational', 'ifRcvAddressStatus' is defined to be 'active' if a unicast address is registered, 'notInService' if it is unregistered and administratively disabled, or 'notReady' otherwise (LES rejected registration, information needed to activate row is missing, etc.). For a client whose 'lecInterfaceState' is 'initialState', 'lecsConnect', or 'configure', 'ifRcvAddressStatus' is defined to be 'active' if the client is configured to register the unicast MAC address on its next Join, 'notInService' if an address is administratively disabled, or 'notReady' otherwise. For a client whose 'lecInterfaceState' is either 'join' or 'initialRegistration', a unicast address which the client will attempt to register, but which the client has not yet attempted to register, shall have the status 'active'. **Multicast MAC Addresses** An 'ifRcvAddressStatus' of 'active' means that incoming packets addressed to the specified multicast address should be delivered to higher protocol (sub-)layers. LEC implementations may, but are not required to, filter out unwanted received multicast traffic based on the multicast MAC addresses in 'ifRcvAddressTable'. Other status values mean that the row will be ignored because it has been disabled ('notInService'), or it has not been properly configured ('notReady'). The client is free to filter incoming packets for the specified multicast address, rather than delivering them to higher (sub-) layers. **Broadcast MAC Addresses** These addresses are treated in a similar fashion to multicast MAC addresses, with one major exception: a management station may not create or delete broadcast address rows. An implementation may permit management stations to set the status of a broadcast address row to 'notInService'. In this event, a client may discard incoming frames, but may not tear down the Multicast Forward VCC.
ifRcvAddressType	Same interpretation as RFC 1573.

5.2 ATM Addresses

Each LAN Emulation Client has zero or more ATM addresses - one or more addresses if it is operational. These addresses may denote different ATM ports or the same port. Several LECs may share an ATM port, provided that they use different ATM addresses.

Because we have reserved each client's 'ifPhysAddress' and 'ifRcvAddressTable' to hold MAC addresses, we cannot use MIB-II and RFC 1573 to store a LEC's ATM addresses. Neither can we use the Interfaces stack to determine them - the many-to-many relationships between LECs, LEC ATM addresses, and ATM ports would make this technique unreliable even if the use of RFC 1573 was not optional.

Thus, we place the addresses in the LAN Emulation Client MIB.

- *lecPrimaryAtmAddress* identifies each client's primary ATM address.
- *lecAtmAddressTable* identifies all of each client's ATM addresses.

6. LAN Emulation Client MIB Description

Each network-manageable host which can act as a LAN Emulation Client SHOULD implement the LAN Emulation Client MIB.

6.1 Motivation

While Interfaces table entries for each LEC are useful, they do not cover all of the things which a network manager might need to know or do. The LEC MIB lets network managers

- Examine Initial State Parameters - including the LAN Name and LES ATM address for each operational client. Given that a LES is likely to know all its current clients, identifying the LES may make it easier for a network manager to locate other clients.
- Monitor statistics for LAN Emulation control traffic and SVC failures. This can provide an indication of whether a LEC is being swamped by LE_ARPs, or is failing to receive them.
- Examine LE_ARP caches.

The LEC MIB also provides the optional abilities to

- Create and destroy LAN Emulation Clients.
- Configure a client's Initial State Parameters.
- Create and destroy LE_ARP cache entries.

6.2 MIB Organization

The LAN Emulation Client MIB is organized into a number of groups, each corresponding to a table. They include

1. Client groups - *one row per client*
    a. Configuration group
    creation, deletion, and configuration

    b. Status group
    status and operational parameters

    c. Statistics group
    counters for LAN Emulation control traffic and SVCs

    d. Server Connections group
    identifies Control and Multicast VCCs

2. ATM Addresses group - *one row per (client, ATM address)*

3. Registered LAN Destination groups - *one row per (client, LAN Destination)*

   a. MAC Addresses group

   b. Route Descriptors group

4. LE_ARP cache groups - *one row per (client, LAN Destination)*

   a. MAC Address translations

   b. Route Descriptor translations

5. Index Mapping group - *one row per client, for translating ifIndex values to lecIndex values*

6. Multicast Forward VCC group - *one row per LANE V2.0 Multicast Forward VCC*

7. Proxy LE_ARP response groups - *two sets of query objects shared by all clients*

   a. MAC Address query objects

   b. Route Descriptor query objects

8. TLV group - *contains sets of TLVs that supplement information in other tables*

6.3 MIB Groups

This section briefly describes each group and object in the LEC MIB. For more detail, consult the MIB definition and the LAN Emulation specification.

6.3.1 Configuration Group

This mandatory group contains settable configuration parameters, and consists of a table with one row for each LAN Emulation Client. Many objects correspond to Initial State Parameters in the LAN Emulation Specification, and are annotated with the appropriate *(Cxx)* label.

- *lecIndex* - Identifies the client.
- *lecRowStatus* - Used to create and destroy clients on hosts which support this.
- *lecOwner* - The entity that configured this entry and is therefore using the resources assigned to it.
- *lecConfigMode* - Controls whether this client uses the LECS to auto-configure.
- *lecConfigLanType (C2c)* - The LAN Type this client will use the next time it enters the Initial state. This MIB object does not reflect LE_{CONFIGURE, JOIN}_RESPONSEs.
- *lecConfigMaxDataFrameSize (C3c)* - The maximum data frame size this client will use the next time it enters the Initial state. This MIB object does not reflect LE_{CONFIGURE, JOIN}_RESPONSEs.
- *lecConfigLanName (C5c)* - The ELAN name this client will use the next time it enters the Initial State. This MIB object does not reflect LE_{CONFIGURE, JOIN}_RESPONSEs.

- *lecConfigLesAtmAddress (C9)* - The LAN Emulation Server which this client will use the next time that you start the client in 'manual' configuration mode. This object is ignored if the client is using the LECS.
- *lecControlTimeout (C7)* - Time-out period used for timing out most request/response control frame interactions. For LANE V2, this is a cumulative maximum.
- *lecMaxUnknownFrameCount (C10)* - Used to limit flooding to the BUS. It has been deprecated in favor of *lecV2MaxUnknownFrameCount.*
- *lecMaxUnknownFrameTime (C11)* - Used to limit flooding to the BUS.
- *lecVccTimeoutPeriod (C12)* - The length of time after which an inactive Data Direct SVC should be closed.
- *lecMaxRetryCount (C13)* - Limits the number of LE_ARP_REQUESTs that can be issued for a given data frame.
- *lecAgingTime (C17)* - The maximum time that a LE Client will maintain an entry in its LE_ARP cache in the absence of a verification of that relationship.
- *lecForwardDelayTime (C18)* - The maximum time that a LE Client will maintain an entry in its LE_ARP cache for a non-local MAC address in the absence of a verification of that relationship, so long as the Topology Change flag is true. This value SHOULD BE less than the *lecAgingTime.*
- *lecExpectedArpResponseTime (C20)* - The maximum time that the LE Client expects a LE_ARP_REQUEST/LE_ARP_RESPONSE cycle to take. This value is used for retries and verifies.
- *lecFlushTimeout (C21)* - Time limit to wait to receive a LE_FLUSH_RESPONSE after the LE_FLUSH_REQUEST has been sent before taking recovery action.
- *lecPathSwitchingDelay (C22)* - The time since sending a frame to the BUS after which the LE Client may assume that the frame has been either discarded or delivered to the recipient.
- *lecLocalSegmentID (C23)* - For an IEEE 802.5 LAN Emulation Client which is a Source Routing bridge, the segment ID of the emulated LAN. This object has been deprecated in favor of *lecConfigLocalSegmentID* and *lecActualLocalSegmentID.*
- *lecMulticastSendType (C24)* - The type of service (best-effort, variable-bit-rate, or constant-bit-rate) that the LE Client should request when establishing the Multicast Send VCC.
- *lecMulticastSendAvgRate (C25)* - The Forward and Backward Sustained Cell Rate that the LE Client should request when setting up the Multicast Send VCC, when requesting variable-bit-rate service.
- *lecMulticastSendPeakRate (C26)* - The Forward and Backward Peak Cell Rate that the LE Client should request when setting up the Multicast Send VCC, when requesting either variable-bit-rate or constant-bit-rate service.
- *lecConnectionCompleteTimer (C28)* - In Connection Establishment, the time period in which data or a READY_IND message is expected from a Calling Party.

- *lecConfigLecsAtmAddress* – The preconfigured LECS address for this client, if one exists. A network manager would probably preconfigure LECS addresses at the client level only if other, more automatic means of establishing LEC-LECS contact had failed.
- *lecInitialControlTimeout (C7i)* – The initial control timeout period for a V2 LEC.
- *lecControlTimeoutMultiplier (C7x)* – The control timeout retry multiplier (or logarithmic back-off base) for a V2 LEC.
- *lecV2MaxUnknownFrameCount (C10)* – Like *lecMaxUnknownFrameCount*, but with a new default value and an upper limit of 20 frames (to accomodate V2 LECs).
- *lecConfigLocalSegmentID (C23c)* – One of two objects that replace *lecLocalSegmentID*. This MIB object does not reflect LE_{CONFIGURE, JOIN} responses.
- *lecConfigV2Capable (C29c)* – Indicates whether a client is configured to act as a V1 LEC or a V2 LEC, in places where V1 and V2 behaviors differ. This MIB object does not reflect LE_{CONFIGURE, JOIN} responses.
- *lecConfigSelectiveMulticast (C32c)* – Indicates whether a client is configured to register the multicast addresses it wants to receive. This MIB object does not reflect LE_{CONFIGURE, JOIN} responses.
- *lecForwardDisconnectTimeout (C33)* – If a V2 LEC detects that the BUS has had no Multicast Forward VCCs set up to the LEC for this time period, the LEC must disconnect from the BUS and try to re-establish a new connection.
- *lecConfigLLCMultiplexCapable (C34c)* – Gives a network manager a way to disable LLC multiplexing on a client which supports it.
- *lecMinReconfigureDelay (C37)* – The minimum time that a LANE V2 client must wait before retrying configuration.
- *lecMaxReconfigureDelay (C38)* – The maximum time that a LANE V2 client must wait before retrying configuration.
- *lecMaxBusConnectRetries (C39)* – The maximum number of times that a LANE V2 client can try to establish the Multicast Send VCC before returning to the configuration procedure.
- *lecTokenRingExplorerExclude (C40)* – Indicates whether or not the LEC wishes to receive Token Ring explorer frames other than those directed to one of its registered MAC addresses. Normally only bridges need to receive such frames.

6.3.2 Status Group

This mandatory group contains read-only status, identification, and operational parameters, and consists of a table with one row for each LAN Emulation Client. Many objects correspond to Initial State Parameters, and are annotated with the appropriate *(Cxx)* label.

- *lecIndex* - Identifies the client.
- *lecIfIndex* - Identifies the client's row in the MIB-II / RFC 1573 Interfaces table.
- *lecPrimaryAtmAddress (C1)* - The primary ATM address of this client.

- *lecID (C14)* - For an operational client, the LEC-ID assigned by the LAN Emulation Server. This ID must be in the range X'0001' through X'FEFF'. For a client which does not yet have a valid LEC-ID, the value 0.
- *lecInterfaceState* - Describes the operational state of each client in more detail than 'ifOperStatus'.
- *lecLastFailureRespCode* - Status code from the last failed Configure or Join response.
- *lecLastFailureState* - The state this client was in when it updated *lecLastFailureRespCode.*
- *lecProtocol* - The LAN Emulation protocol which this client supports.
- *lecVersion* - The LAN Emulation protocol version which this client supports.
- *lecTopologyChange (C19)* - Indicates whether the LE Client is using the Forward Delay Time to age non-local entries in its LE-ARP cache.
- *lecConfigServerAtmAddress* - The address of the LAN Emulation Configuration Server.
- *lecConfigSource* - Indicates whether this client used the LECS, and, if so, what method it used to establish the Configuration Direct VCC.
- *lecActualLanType (C2)* - The LAN Type this client is using now.
- *lecActualMaxDataFrameSize (C3)* - The maximum data frame size this client is using now.
- *lecActualLanName (C5)* - The ELAN name this client is using now.
- *lecActualLesAtmAddress (C9)* - The LAN Emulation Server address currently in use or most recently attempted.
- *lecProxyClient (C4)* - Indicates whether this client is a proxy client.
- *lecActualLocalSegmentID (C23)* - For an IEEE 802.5 LAN Emulation Client which is a Source Routing bridge, the segment ID of the emulated LAN.
- *lecActualV2Capable (C29)* - Indicates whether the client is acting as a LANE V2 client or as a LANE V1 client.
- *lecElanID (C31)* - The ELAN-ID used in all LANE LLC-multiplexed data frames sent or received by the client.
- *lecActualSelectiveMulticast (C32)* - Indicates whether the client is set up to register its multicast addresses, so it can take advantage of Selective Multicast VCCs.
- *lecActualLLCMultiplexCapable (C34)* - Indicates whether a client can, and is allowed to, accept LLC-multiplexed Data Direct VCCs.
- *lecPreferredLesAddress (C35)* - The Preferred LAN Emulation Server's address. This is set during the Configuration phase.
- *lecStoresServiceCategories (C4)* - Indicates whether or not the client saves Service Category TLV data and makes it available through the leArpTable and leRDArpTable.

6.3.3 Statistics Group

This mandatory group consists of a table with one row for each LAN Emulation Client.

- *lecIndex* - Identifies the client.
- *lecArpRequestsOut, lecArpRequestsIn, lecArpRepliesOut, lecArpRepliesIn* - Counts of the LE_ARP requests and replies received and transmitted by this client.
- *lecControlFramesOut, lecControlFramesIn* - The total number of control frames sent and received by this client.
- *lecSvcFailures* - The number of SVCs which this client either tried and failed to establish, or rejected for protocol or security reasons.
- *lecFlowFailures* - The number of LLC-multiplexed flows which this client either tried and failed to establish, or rejected.
- *lecEchoDiscards* - The number of data frames this client has discarded because they were copies of its own traffic. Such frames are no longer counted against ifInDiscards.
- *lecFilteredMulticasts* - The number of data frames this client has discarded because they were for unwanted multicast or group addresses. Such frames are no longer counted against ifInDiscards. This counter will vary depending on whether / how well the client and server support selective multicasting.

6.3.4 Server Connections Group

This mandatory group consists of a table with one row for each LAN Emulation Client.

- *lecIndex* - Identifies the client.
- *lecConfigDirect{ Interface, Vpi, Vci }* - Identify the Configuration Direct VCC, if any.
- *lecControlDirect{ Interface, Vpi, Vci }* - Identify the Control Direct VCC, if any.
- *lecControlDistribute{ Interface, Vpi, Vci }* - Identify the Control Distribute VCC, if any.
- *lecMulticastSend{ Interface, Vpi, Vci }* - Identify the (Default) Multicast Send VCC, if any.
- *lecMulticastForward{ Interface, Vpi, Vci }* - Identify the LANE V1.0 Multicast Forward VCC, if any. For LANE V2.0, where there is often no way to identify the Default Multicast Forward VCC, these objects have the value (0, 0, 0).

6.3.5 ATM Addresses Group

This mandatory group lists all of the ATM Addresses (Initial State Parameter C1) for this host's LAN Emulation Clients. It consists of a table indexed by LE Client and ATM address.

- *lecIndex* - Identifies the client.
- *lecAtmAddress* - Identifies one of its ATM addresses.
- *lecAtmAddressStatus* - Allows managers to create and delete table rows.

- *lecAtmAddressMuxed* - Indicates whether the LEC is willing to establish non-multiplexed connections, multiplexed connections, or both at the specified ATM address.

6.3.6 Registered MAC Addresses Group

This mandatory group lists all of the Local Unicast MAC Addresses (Initial State Parameter C6) and LE Client Multicast MAC Addresses (Initial State Parameter C15) registered for this host's LAN Emulation Clients. It consists of a table indexed by LE Client and MAC address.

- *lecIndex* - Identifies the client.
- *lecMacAddress* - The registered MAC address.
- *lecMacAddressAtmBinding* - The non-multiplexed ATM address registered for this MAC address.
- *lecMacAddressMuxedAtmBinding* - The LLC-multiplexed ATM address registered for this MAC address.
- *lecMacAddressServiceCategory* - A bitmap describing the service categories that the LEC supports for this MAC address.

6.3.7 Registered Route Descriptors Group

This conditionally-mandatory group lists all of the Route Descriptors (Initial State Parameter C8) registered for this host's LAN Emulation clients. It consists of a table indexed by LE Client, IEEE 802.5 LAN ID, and bridge number.

- *lecIndex* - Identifies the client.
- *lecRouteDescrSegmentID* - The LAN ID portion of the IEEE 802.5 route descriptor.
- *lecRouteDescrBridgeNumber* - The Bridge Number portion of the route descriptor.
- *lecRouteDescrAtmBinding* - The non-multiplexed ATM address registered for this route descriptor.
- *lecRouteDescrMuxedAtmBinding* - The LLC-multiplexed ATM address registered for this route descriptor.
- *lecRouteDescrServiceCategory* - A bitmap describing the service categories that the LEC supports for this route descriptor.

6.3.8 LE_ARP Cache Group - MAC Addresses

This mandatory group provides access to a LAN Emulation Client's MAC-to-ATM ARP cache (Initial State Parameter C16). It consists of a table indexed by LE Client and MAC address.

- *lecIndex* - Identifies the client.
- *leArpMacAddress* - The MAC address for which this cache entry provides a translation. This may be a unicast, multicast, or broadcast MAC address; note that multicast LE-ARPs are illegal for LANE V1 clients.

- *leArpAtmAddress* - The non-multiplexed ATM address to which it translates.
- *leArpIsRemoteAddress* - Indicates whether *leArpMACaddress* is local or remote.
- *leArpEntryType* - Indicates how this entry was created and whether it is aged.
- *leArpRowStatus* - Lets network managers create and destroy LE_ARP cache entries.
- *leArpMuxedAtmAddress* - The LLC-multiplexed ATM address for *leArpMacAddress*.
- *leArpRowStatus* - Service Category information from the LE_ARP_RESPONSE.

6.3.9 LE_ARP Cache Group - Route Descriptors

This conditionally-mandatory group provides access to an IEEE 802.5 LAN Emulation Client's Route Descriptor-to-ATM ARP cache (Initial State Parameter C16). It consists of a table indexed by LE Client, IEEE 802.5 LAN ID, and bridge number.

- *lecIndex* - Identifies the client.
- *leRDArpSegmentID* - The LAN ID portion of the IEEE 802.5 route descriptor.
- *leRDArpBridgeNumber* - The Bridge Number portion of the IEEE 802.5 route descriptor.
- *leRDArpAtmAddress* - The non-multiplexed ATM address to which it translates.
- *leRDArpEntryType* - Indicates how this entry was created and whether it is aged.
- *leRDArpRowStatus* - Lets network managers create and destroy LE_ARP cache entries.
- *leRDArpMuxedAtmAddress* - The LLC-multiplexed ATM address to which the IEEE 802.5 route descriptor translates.
- *leRDArpServiceCategory* - Service Category information from the LE_ARP_RESPONSE.
- *leRDArpIsRemoteDescriptor* - Indicates whether the route descriptor is local or remote.

6.3.10 Index Mapping Group

This mandatory group describes the ifIndex --> lecIndex translation. It consists of a table with one row for each LE Client.

- *ifIndex* - Identifies the client's entry in the MIB-II / RFC 1573 Interfaces table.
- *lecMappingIndex* - Identifies the corresponding *lecIndex* value.

6.3.11 Multicast Forward VCC Group

This group identifies Multicast Forward VCCs belonging to LANE V2.0 clients. It is indexed by LEC ID, interface, VPI, and VCI.

- *lecMcForwardInterface* - The interface on which the Multicast Forward VCC lives.
- *lecMcForwardVpi* - The VPI of the Multicast Forward VCC.
- *lecMcForwardVci* - The VCI of the Multicast Forward VCC.

- *lecMcForwardVerification* - Indicates whether the caller's address was verified implicitly, explicitly, or not at all.

6.3.12 Proxy LE_ARP Response Group – MAC Addresses

This optional group lets a network manager find out what ATM addresses and Service Category TLVs a proxy client will use for its "C27 Remote Unicast MAC Address(es)." It differs from the LE_ARP group in that it focuses on the LE_ARP_RESPONSEs the proxy client would send to other clients, rather than on the ones it has received. To avoid the need for a new MIB table the size of a bridge's forwarding database, these objects implement a request (SET) / response (GET) model.

- *lecMacQueryLecIndex* - When written, identifies the client to be queried and triggers a query. When read, identifies the client to which results apply.
- *lecMacQueryAddress* - When written, identifies the MAC address for which information is desired and triggers a query. When read, identifies the MAC address to which results apply.
- *lecMacQueryStatus* - Indicates if the latest query succeeded, and if so, whether the address was local or remote.
- *lecMacQueryAtmAddress* - The non-multiplexed ATM address, if any, associated with *lecMacQueryAddress.*
- *lecMacQueryMuxedAtmAddress* - The LLC-multiplexed ATM address, if any, associated with *lecMacQueryAddress.*
- *lecMacQueryServiceCategory* - The Service Category information, if any, associated with *lecMacQueryAddress.*

6.3.13 Proxy LE_ARP Response Group – Route Descriptors

This optional group lets a network manager find out what ATM addresses and Service Category TLVs a proxy client will use for its "C30 Remote Route Descriptors." It differs from the LE_ARP group in that it focuses on the LE_ARP_RESPONSEs the proxy client would send to other clients, rather than on the ones it has received.

- *lecRDQueryLecIndex* - When written, identifies the client to be queried and triggers a query. When read, identifies the client to which results apply.
- *lecRDQuerySegmentID* - When written, identifies the Segment ID portion of the IEEE 802.5 route descriptor for which information is desired and triggers a query. When read, identifies the Segment ID of the IEEE 802.5 route descriptor to which results apply.
- *lecRDQueryBridgeNumber* - When written, identifies the Bridge Number portion of the IEEE 802.5 route descriptor for which information is desired and triggers a query. When read, identifies the Bridge Number of the IEEE 802.5 route descriptor to which results apply.
- *lecRDQueryStatus* - Indicates if the latest query succeeded, and if so, whether the route descriptor was local or remote.

- *lecRDQueryAtmAddress* - The non-multiplexed ATM address, if any, associated with (*lecRDQuerySegmentID, lecRDQueryBridgeNumber)*..
- *lecRDQueryMuxedAtmAddress* - The LLC-multiplexed ATM address, if any, associated with (*lecRDQuerySegmentID, lecRDQueryBridgeNumber)*..
- *lecRDQueryServiceCategory* - The Service Category information, if any, associated with (*lecRDQuerySegmentID, lecRDQueryBridgeNumber)*..

6.3.14 TLV Group

This table provides a place for an agent to store sets of TLVs that are associated with other MIB entries (such as leArpTable entries). This table is indexed by the *lecIndex*, the *lecTlvSetIndex*, the *lecTlvLanDestination*, and the *lecTlvIndex*.

- *lecTlvSetIndex* - Helps to identify a set of TLVs by indicating their purpose (e.g., LE_ARP cache TLVs, registration TLVs, etc.). See the DESCRIPTION of this MIB object for the rules which govern the use of each sub-table.
- *lecTlvLanDestination* - The LAN Destination (or other well-defined index) associated with the TLVs.
- *lecTlvIndex* - An arbitrary index value which allows multiple TLVs to be associated with a particular set, even if they have the same type.
- *lecTlvType* - The type of the TLV.
- *lecTlvValue* - The value (and, implicitly, the length) of the TLV.
- *lecTlvRowStatus* - For creating and deleting TLVs (when appropriate).

6.4 Interaction with ifAdminStatus

Suppose that we have an inactive client whose lecIndex is 1, whose lecIfIndex is 10, and whose lecConfigMode is manual. Now suppose that we send its SNMP agent this request:

```
ifAdminStatus.10 = up;
lecConfigMode.1 = automatic;
```

Assuming that the request is accepted, two things might happen.

- The agent might set the client into automatic mode before enabling it, causing the client to auto-configure. This would be the most useful response.
- The agent might also start the client up in manual configuration mode, and then tell it to auto-configure next time, surprising the user in an unpleasant way.

Recommendation

When processing a SetRequest PDU which will change a client's parameters and enable the client, an agent SHOULD update the parameters before enabling the client.

6.5 Limitations

The LEC MIB does not provide ways to

- Identify Data Direct VCCs.
- Get VCC topology information (local and far-end ATM addresses) that is not currently in the AToM MIB, or
- Configure PVCs for LAN Emulation use. (The use of PVCs for LAN Emulation is beyond the scope of the LUNI 2.0 specification.)

It is expected that some of these features will be incorporated into other ATM MIBs.

7. LAN Emulation Client MIB Text

```
--
--      MIB for configuration, performance, and fault management of ATM
--      LAN Emulation Clients.
--


        LAN-EMULATION-CLIENT-MIB DEFINITIONS ::= BEGIN


            IMPORTS

                MODULE-IDENTITY, OBJECT-TYPE,
                enterprises, Counter32, Integer32      FROM SNMPv2-SMI

                TEXTUAL-CONVENTION, MacAddress,
                RowStatus, TruthValue, DisplayString   FROM SNMPv2-TC

                MODULE-COMPLIANCE, OBJECT-GROUP        FROM SNMPv2-CONF

                InterfaceIndex                         FROM IF-MIB

                ifIndex                                FROM RFC1213-MIB

                OwnerString                            FROM RFC1271-MIB;



            leClientMIB MODULE-IDENTITY
                LAST-UPDATED "9802230000Z"
                ORGANIZATION "ATM Forum LAN Emulation Sub-Working Group"
                CONTACT-INFO
                      "
                        The ATM Forum
                        2570 West El Camino Real, Suite 304
                        Mountain View, CA  94040-1313
                        Tel: +1-650-949-6700
                        Fax: +1-650-949-6705
                        E-mail: info@atmforum.com
                        Web: http://www.atmforum.com/
                                                           "
                DESCRIPTION
                    "This module defines a portion of the management
                     information base (MIB) for managing ATM LAN Emulation
                     Client nodes.  It is meant to be used in connection with
                     the AToM MIB and MIB-II / RFC 1573 'ifTable' entries
                     for each LEC / emulated 802.x network interface."

                REVISION
                    "9505100000Z"
                DESCRIPTION
                    "The LAST-UPDATED date of the MIB in the Version 1.0 LAN
                     Emulation Client management specification."

                REVISION
                    "9802230000Z"
                DESCRIPTION
                    "This MIB module now provides extensive support for LAN
                     Emulation Version 2.0 clients, and the optional ability
```

```
          to query a proxy client (say, one attached to a bridge)
          to find out how it would respond to a LE_ARP request."

     ::= { atmfLanEmulation 1 }


-- ------------------------------------------------------------ --
--
--  If your compiler gives you trouble with the forward reference
--  to 'atmfLanEmulation', you may need to move the definition to
--  a non-standard place before the MODULE-IDENTITY statement.
--
-- ------------------------------------------------------------ --


atmForum                  OBJECT IDENTIFIER ::= { enterprises 353 }
atmForumNetworkManagement OBJECT IDENTIFIER ::= { atmForum 5 }

-- The object identifier subtree for ATM Forum LAN Emulation MIBs
atmfLanEmulation          OBJECT IDENTIFIER ::=
                                    { atmForumNetworkManagement 3 }


leClientMIBObjects  OBJECT IDENTIFIER ::= { leClientMIB 1 }


-- This MIB module consists of the following groups:
--
--  (1) LAN Emulation Client groups
--      (a) Configuration group
--      (b) Status group
--      (c) Statistics group
--      (d) Server VCC group
--
--  (2) ifIndex mapping group
--
--  (3) LEC ATM Addresses group
--
--  (4) Registered LAN Destination groups
--      (a) MAC addresses group
--      (b) Route Descriptors group
--
--  (5) LE_ARP cache groups
--      (a) MAC address cache group
--      (b) Route descriptor cache group
--
--  (6) Multicast Forward VCC group
--
--  (7) Proxy LE_ARP response groups
--      (a) MAC address query group
--      (b) Route descriptor query group
--
--  (8) TLV table group
--
-- Security management objects have been explicitly omitted from
-- this MIB as being outside the scope of the V2.0 LAN Emulation
-- specification.


AtmLaneAddress ::= TEXTUAL-CONVENTION
    STATUS     current
    DESCRIPTION
          "A twenty-octet binary string, containing a standard
           ATM Forum address -- or the zero-length string,
           indicating the absence of an address.

           For LAN Emulation purposes, the eight-octet address
           format is not used."
    SYNTAX     OCTET STRING ( SIZE( 0 | 20 ) )

VpiInteger ::= TEXTUAL-CONVENTION
    STATUS     current
    DESCRIPTION
          "An integer large enough to hold a VPI.  Objects that
           use this textual convention should contain either

             -  A value which conforms to the AtmVpIdentifier
                convention in the proposed ATM-TC-MIB, or

             -  Zero (if there is no circuit/path data).

           AtmVpIdentifier is currently defined as

                The VPI value for a VPL or VCL.  The value VPI=0
                is only allowed for a VCL.  For ATM UNIs supporting
                VPCs the VPI value ranges from 1 to 255.  For ATM
                UNIs supporting VCCs the VPI value ranges from 0
                to 255.  For ATM NNIs the VPI value ranges from 0
                to 4095.  The maximum VPI value cannot exceed the
                value allowable by atmInterfaceMaxVpiBits defined
                in ATM-MIB.

           To align the textual conventions, VpiInteger now ranges
           between 0 and 4095, instead of between 0 and 255."
    SYNTAX     INTEGER ( 0..4095 )

VciInteger ::= TEXTUAL-CONVENTION
    STATUS     current
    DESCRIPTION
         "An integer large enough to hold a VCI.  Objects that
          use this textual convention should contain either

             -  A value which conforms to the AtmVcIdentifier
                convention in the proposed ATM-TC-MIB, or

             -  Zero (if there is no circuit data).

          AtmVcIdentifier is currently defined as

               The VCI value for a VCL.  The maximum VCI value
               cannot exceed the value allowable by
               atmInterfaceMaxVciBits defined in ATM-MIB."
    SYNTAX     INTEGER ( 0..65535 )

LeConnectionInterface ::= TEXTUAL-CONVENTION
    STATUS     current
    DESCRIPTION
            "A combination flag and ifTable index, indicating
             whether a LAN Emulation circuit exists - and, if
             so, which interface is associated with it at the
             point where it connects to a LE Client.

             Objects of this type are always defined as part of
             a set that includes

                     fooInterface   LeConnectionInterface
                     fooVpi         VpiInteger
                     fooVci         VciInteger
             The interpretation of these objects is as follows.

             1.  If no connection exists, 'fooInterface' has a
                 value of 0.  Because Interfaces table entries
                 always have 'ifIndex' values larger than 0,
                 'fooInterface' reliably serves as a 'connection
                 exists' flag.

                 In this case, 'fooVpi' and 'fooVci' are
                 meaningless, and have the value 0.

             2.  If a PVC or SVC exists, 'fooInterface' is defined
                 to have the value of the MIB-II/RFC 1573 'ifIndex'
                 of the 'atm' interface associated with the VCC.
                 'fooVpi' and 'fooVci' will contain its VPI/VCI.
                 Therefore, ('fooInterface', 'fooVpi', 'fooVci')
                 give you an index into the AToM MIB which allows
                 you to examine and play with the circuit further.

              3. If a proprietary connection (such as an internal
                 data path between co-located components) exists,
                 this specification does not mandate how it should
                 be managed.  One option is to set 'fooInterface'
                 to the value of the MIB-II/RFC 1573 'ifIndex' for
                 the LAN Emulation Client.  This indicates that a
                 connection exists, but is not being managed in the
                 same fashion as an ATM VCC."
    SYNTAX   INTEGER ( 0..2147483647 )

LecState ::= TEXTUAL-CONVENTION
    STATUS     current
    DESCRIPTION
          "An official protocol state of a LAN Emulation Client."
    REFERENCE
          "ATM Forum LAN Emulation Over ATM Specification, V1.0,
           Section 2.3.1"
    SYNTAX     INTEGER {
                         initialState(1),
                         lecsConnect(2),
                         configure(3),
                         join(4),
                         initialRegistration(5),
                         busConnect(6),
                         operational(7)
                        }

LecDataFrameFormat ::= TEXTUAL-CONVENTION
    STATUS     current
    DESCRIPTION
          "A 'LAN Type' value.

           C2  LAN Type.  The type of LAN that the LE Client is,
               or wishes to become, a member of.  This MUST be one
               of Ethernet/IEEE 802.3, IEEE 802.5, or Unspecified."
    REFERENCE
          "ATM Forum LAN Emulation Over ATM Specification, V1.0,
           Section 5.1.1"
    SYNTAX      INTEGER {
                         unspecified(1),
                         aflane8023(2),
                         aflane8025(3)
                        }
```

```
LecDataFrameSize ::= TEXTUAL-CONVENTION
    STATUS     current
    DESCRIPTION
          "A 'Maximum Data Frame Size' value.

           C3  Maximum Data Frame Size.  The maximum AAL-5 SDU size
               of a data frame that the LE Client wishes to send on
               any Multicast Send VCC, or to receive on any
               Multicast Send VCC or Multicast Forward VCC.

               This parameter also specifies the maximum AAL-5 SDU
               on all of a LE Client's non-multiplexed Data Direct
               VCCs, and the maximum AAL-5 SDU - 12 (for overhead)
               on all of a LE Client's LLC-multiplexed Data Direct
               VCCs.

               This setting MUST be either 1516, 1580, 4544, 9234,
               or 18190 octets, or Unspecified.  The 1580 value is
               only available to 'V2 Capable' clients."
    REFERENCE
          "ATM Forum LAN Emulation V1 and V2 LUNI specifications,
           Section 5.1.1"
    SYNTAX      INTEGER {
                         -- LANE 1.0
                         unspecified(1),
                         max1516(2),
                         max4544(3),
                         max9234(4),
                         max18190(5),

                         -- LANE 2.0
                         max1580(6)
                        }

LeArpTableEntryType ::= TEXTUAL-CONVENTION
    STATUS     current
    DESCRIPTION
            "Indicates how a LE_ARP table entry was created and
             whether it is subject to aging.

                 other(1)
                     The LE_ARP entry does not fall into one of
                     the categories defined below.

                 learnedViaControl(2)
                     This LE_ARP entry was learned by using the
                     LE_ARP_REQUEST / LE_ARP_RESPONSE protocol,
                     or by otherwise observing relevant traffic
                     on Control VCCs.

                     Entries of this type are aged.

                 learnedViaData(3)
                     This LE_ARP entry was learned by observing
                     incoming traffic on Data VCCs.

                     Entries of this type are aged.

                 staticVolatile(4)
                     This LE_ARP entry was created by management.
                     It will not be aged.  On the other hand, it
                     will not survive a restart of the client.

                     If an agent permits LE_ARP entries of this
```

```
                        type to be created, it may require that the
                        LAN Emulation Client be in the 'operational'
                        state at the time.

                    staticNonVolatile(5)
                        This LE_ARP entry was created by management.
                        It will not be aged, and it will survive a
                        restart of the client.

                The values 'learnedViaData' and 'learnedViaControl'
                can be read, but may not be written."
        SYNTAX      INTEGER {
                            other(1),
                            learnedViaControl(2),
                            learnedViaData(3),
                            staticVolatile(4),
                            staticNonVolatile(5)
                           }


    -- ------------------------------------------------------------ --

    --
    --  LAN Emulation Client configuration group
    --

    lecConfigTable OBJECT-TYPE
        SYNTAX       SEQUENCE OF LecConfigEntry
        MAX-ACCESS   not-accessible
        STATUS       current
        DESCRIPTION
               "A table describing all of the LAN Emulation Clients
                implemented by this host.  Each LE Client has a row
                in the MIB-II/RFC 1573 Interfaces table (describing
                the emulated packet interface it presents to higher
                layers).  Each LE Client also has a row in this and
                other LEC MIB tables (describing its interface with
                other LAN Emulation components).

                This table contains configuration variables.  Three
                extension tables contain client status, performance
                statistics, and information about control/multicast
                VCCs.

                Unlike hardware ports, LE Clients can be created by
                management.  However, the RFC 1573 Interfaces table
                does not directly support row creation.  Therefore,
                creating or deleting a row in the lecConfigTable is
                defined to have the side effect of creating or
                deleting corresponding rows in

                    -  the MIB-II / RFC 1573 Interfaces table,
                    -  the 'lecMappingTable',
                    -  the 'lecStatusTable',
                    -  the 'lecServerVccTable',
                    -  the 'lecStatisticsTable', and
                    -  any other dependent tables

                New Interfaces table rows for LAN Emulation Clients
                always have 'ifAdminStatus' set to 'down'.


                A Note On Indexing:
                -------------------
                Most of the tables in this MIB are indexed in whole
                or in part by 'lecIndex' - not by 'ifIndex'.

                Why is there a separate index?

                Traditionally, ifIndex values are chosen by agents,
                and are permitted to change across restarts.  Using
                ifIndex to index lecConfigTable could complicate
                row creation and/or cause interoperability problems
                (if each agent had special restrictions on ifIndex).
                Having a separate index avoids these problems.      "
        ::= { leClientMIBObjects 1 }

    lecConfigEntry OBJECT-TYPE
        SYNTAX       LecConfigEntry
        MAX-ACCESS   not-accessible
        STATUS       current
        DESCRIPTION
               "Each table entry contains configuration information
                for one LAN Emulation Client.

                Most of the objects are derived from Initial State
                Parameters in the LAN Emulation specification."
        INDEX   { lecIndex }
        ::= { lecConfigTable 1 }

    LecConfigEntry ::=
        SEQUENCE {

            --
            --  Row indices, creation, and deletion.
            --
            lecIndex                    INTEGER,
            lecRowStatus                RowStatus,
            lecOwner                    OwnerString,

            --
            --  Key parameters for Configure / Join phase.  Note that
            --  these are not operational values - see lecStatusTable
            --  for those.
            --
            lecConfigMode               INTEGER,
            lecConfigLanType            LecDataFrameFormat,
            lecConfigMaxDataFrameSize   LecDataFrameSize,
            lecConfigLanName            DisplayString,
            lecConfigLesAtmAddress      AtmLaneAddress,

            --
            --  Other parameters.  These may be changed either by the
            --  network manager or by the LE Configuration Server.
            --
            lecControlTimeout              INTEGER,
            lecMaxUnknownFrameCount        INTEGER,
            lecMaxUnknownFrameTime         INTEGER,
            lecVccTimeoutPeriod            Integer32,
            lecMaxRetryCount               INTEGER,
            lecAgingTime                   INTEGER,
            lecForwardDelayTime            INTEGER,
            lecExpectedArpResponseTime     INTEGER,
            lecFlushTimeOut                INTEGER,
            lecPathSwitchingDelay          INTEGER,
            lecLocalSegmentID              INTEGER,
            lecMulticastSendType           INTEGER,
            lecMulticastSendAvgRate        Integer32,
            lecMulticastSendPeakRate       Integer32,
            lecConnectionCompleteTimer     INTEGER,

            --
            --  New parameter for LANE 1.0 addendum
            --

            lecConfigLecsAtmAddress        AtmLaneAddress,

            --
            --  Objects to support LANE 2.0
            --

            lecInitialControlTimeout       INTEGER,
            lecControlTimeoutMultiplier    INTEGER,
            lecV2MaxUnknownFrameCount      INTEGER,
            lecConfigLocalSegmentID        INTEGER,
            lecConfigV2Capable             TruthValue,
            lecConfigSelectiveMulticast    TruthValue,
            lecForwardDisconnectTimeout    INTEGER,
            lecConfigLLCMultiplexCapable   TruthValue,
            lecMinReconfigureDelay         INTEGER,
            lecMaxReconfigureDelay         INTEGER,
            lecMaxBusConnectRetries        INTEGER,
            lecTokenRingExplorerExclude    TruthValue

        }

    lecIndex OBJECT-TYPE
        SYNTAX      INTEGER ( 1..2147483647 )
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
                "A value which uniquely identifies this conceptual
                 row in the lecConfigTable.

                 If the conceptual row identified by this value of
                 lecIndex is recreated following an agent restart,
                 the same value of lecIndex must be used to identify
                 the recreated row.  (However, the Interfaces table
                 index associated with the client may change.)"
        ::= { lecConfigEntry 1 }

    lecRowStatus OBJECT-TYPE
        SYNTAX      RowStatus
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
                "This object lets network managers create and delete
                 LAN Emulation Clients, on systems that support this
                 optional capability.

                 It does not control Joins and Terminations; they are
                 controlled by 'ifAdminStatus'.  However, taking rows
                 out of service will have the side effect of changing
                 their 'ifAdminStatus' values to 'down', thus causing
                 any active emulated LAN connections to be terminated.

                 Within each conceptual lecConfigTable row, objects
                 which are writable may be modified regardless of the
                 value of lecRowStatus.  It is not necessary to set a
                 row's status to 'notInService' first.

                 When creating a LAN Emulation Client, it is up to the
                 management station to determine a suitable 'lecIndex'.
                 To facilitate interoperability, agents should not put
                 any restrictions on the 'lecIndex' beyond the obvious
                 ones that it be valid and unused.

                 If you create a LAN Emulation Client via this object,
                 it will initially have

                     'ifAdminStatus' = 'down'
                     'ifOperStatus' = 'down'
                     'lecInterfaceState' = 'initialState'            "
        ::= { lecConfigEntry 2 }

    lecOwner OBJECT-TYPE
        SYNTAX      OwnerString( SIZE (0..127) )
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
                "The entity that configured this entry and is therefore
                 using the resources assigned to it."
        ::= { lecConfigEntry 3 }

    lecConfigMode OBJECT-TYPE
        SYNTAX      INTEGER {
                             automatic(1),
                             manual(2)
                            }
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
                "Indicates whether this LAN Emulation Client should
                 auto-configure the next time it is (re)started.

                 In automatic(1) mode, a client uses a LAN Emulation
                 Configuration Server to learn the ATM address of its
                 LAN Emulation Server, and to obtain other parameters.

                 lecConfig{ LanType, MaxDataFrameSize, LanName } are
                 used in the Configure request.  lecConfigLesAtmAddress
                 is ignored.

                 In manual(2) mode, management tells the client the
                 ATM address of its LAN Emulation Server and the
                 values of other parameters.

                 lecConfig{ LanType, MaxDataFrameSize, LanName } are
                 used in the Join request.  lecConfigLesAtmAddress
                 tells the client which LES to call."
        REFERENCE
                "ATM Forum LAN Emulation Over ATM Specification, V1.0,
                 Sections 3.4.1.1 and 5.3"
        DEFVAL   { automatic }
        ::= { lecConfigEntry 4 }

    lecConfigLanType OBJECT-TYPE
        SYNTAX      LecDataFrameFormat
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
                "C2c  LAN Type (configured).

                 The data frame format which this client will use the
                 next time it returns to the Initial State.
```

```
             Auto-configuring clients use this parameter in their
             Configure requests.  Manually-configured clients use
             it in their Join requests.

             This MIB object will not be overwritten with the new
             value from a LE_{JOIN,CONFIGURE}_RESPONSE.  Instead,
             lecActualLanType will be."
    REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification, V1.0
             and V2.0, Section 5.1.1"
    DEFVAL   { unspecified }
    ::= { lecConfigEntry 5 }

lecConfigMaxDataFrameSize OBJECT-TYPE
    SYNTAX      LecDataFrameSize
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
            "C3c  Maximum Data Frame Size (configured).

             The maximum data frame size which this client will
             use the next time it returns to the Initial State.

             Auto-configuring clients use this parameter in their
             Configure requests.  Manually-configured clients use
             it in their Join requests.

             This MIB object will not be overwritten with the new
             value from a LE_{JOIN,CONFIGURE}_RESPONSE.  Instead,
             lecActualMaxDataFrameSize will be."
    REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification, V1.0
             and V2.0, Section 5.1.1"
    DEFVAL   { unspecified }
    ::= { lecConfigEntry 6 }

lecConfigLanName OBJECT-TYPE
    SYNTAX      DisplayString (SIZE( 0..32 ))
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
            "C5c  ELAN Name (configured).

             The ELAN Name this client will use the next time it
             returns to the Initial State.

             Auto-configuring clients use this parameter in their
             Configure requests.  Manually-configured clients use
             it in their Join requests.

             This MIB object will not be overwritten with the new
             value from a LE_{JOIN,CONFIGURE}_RESPONSE.  Instead,
             lecActualLanName will be."
    REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification, V1.0
             and V2.0, Section 5.1.1"
    ::= { lecConfigEntry 7 }

lecConfigLesAtmAddress OBJECT-TYPE
    SYNTAX      AtmLaneAddress
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
            "C9  LE Server ATM Address.
```

```
                         The LAN Emulation Server which this client will use
                         the next time it is started in manual configuration
                         mode.

                         When lecConfigMode is 'automatic', there is no need
                         to set this address, and no advantage to doing so.
                         The client will use the LECS to find a LES, putting
                         the auto-configured address in lecActualLesAtmAddress
                         while leaving lecConfigLesAtmAddress alone."
                REFERENCE
                        "ATM Forum LAN Emulation Over ATM Specification, V1.0
                         and V2.0, Section 5.1.1"
                ::= { lecConfigEntry 8 }

            lecControlTimeout OBJECT-TYPE
--
--  For LANE V1.0, this object had a default of 120.
--
                SYNTAX      INTEGER ( 10..300 )
                UNITS       "seconds"
                MAX-ACCESS  read-create
                STATUS      current
                DESCRIPTION
                        "C7  Control Time-out.  Time out period used for
                             timing out most request/response control frame
                             interactions, as specified elsewhere [in the
                             LAN Emulation specification].

                             In LANE V2.0, this value specifies the maximum
                             cumulative timeout for an exponential backoff
                             algorithm.

                         This time value is expressed in seconds."
                REFERENCE
                        "ATM Forum LAN Emulation Over ATM Specification, V1.0
                         and V2.0, Section 5.1.1"
                DEFVAL   { 30 }
                ::= { lecConfigEntry 9 }

            lecMaxUnknownFrameCount OBJECT-TYPE
                SYNTAX      INTEGER ( 1..10 )
                UNITS       "frames"
                MAX-ACCESS  read-create
                STATUS      deprecated
                DESCRIPTION
                        "This is the LANE V1.0 version of 'C10 Maximum Unknown
                         Frame Count'.  It should not be implemented except as
                         required for backwards compatibility.

                         It has been deprecated because LANE V2.0 permits this
                         parameter to take values up to 20 -- a revision which
                         doesn't seem to be allowed under rules in RFC 1902."
                REFERENCE
                        "ATM Forum LAN Emulation Over ATM Specification, V1.0,
                         Section 5.1.1"
                DEFVAL   { 1 }
                ::= { lecConfigEntry 10 }

            lecMaxUnknownFrameTime OBJECT-TYPE
                SYNTAX      INTEGER ( 1..60 )
                UNITS       "seconds"
                MAX-ACCESS  read-create
                STATUS      current
```

```
    DESCRIPTION
            "C11 Maximum Unknown Frame Time.  Within the period of
                 time defined by the Maximum Unknown Frame Time, a
                 LE Client will send no more than Maximum Unknown
                 Frame Count frames to the BUS for a given unicast
                 LAN Destination, and it must also initiate the
                 address resolution protocol to resolve that LAN
                 Destination.

                 LANE V2.0 changes this to 'will send no more than
                 Maximum Unknown Frame Count frames to the BUS ...
                 without initiating the [LE_ARP]', implying that a
                 LANE V2.0 client is not always required to send a
                 LE_ARP_REQUEST along with the unknown frames.

             This time value is expressed in seconds."
    REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification, V1.0
             and V2.0, Section 5.1.1"
    DEFVAL   { 1 }
    ::= { lecConfigEntry 11 }

lecVccTimeoutPeriod OBJECT-TYPE
    SYNTAX      Integer32
    UNITS       "seconds"
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
            "C12 VCC Time-out Period.  A LE Client SHOULD release
                 any Data Direct VCC that it has not used to
                 transmit or receive any data frames for the
                 length of the VCC Time-out Period.  This parameter
                 is only meaningful for SVC Data Direct VCCs
                 signalled with the Ethernet/IEEE 802.3 LE Data
                 Direct VCC B-LLI value or the IEEE 802.5 LE Data
                 Direct VCC B-LLI value.  It should not be used for
                 any SVC signalled with the LLC-Multiplexed B-LLI
                 value.

             This time value is expressed in seconds.  The default
             value is 20 minutes.  A value of 0 seconds means that
             the timeout period is infinite.  Negative values will
             be rejected by the agent."
    REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification, V1.0
             and V2.0, Section 5.1.1"
    DEFVAL   { 1200 }
    ::= { lecConfigEntry 12 }

lecMaxRetryCount OBJECT-TYPE
    SYNTAX      INTEGER( 0..2 )
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
            "C13 Maximum Retry Count.  A LE CLient MUST not retry
                 a LE_ARP_REQUEST for a given frame's LAN
                 destination more than Maximum Retry Count times,
                 after the first LE_ARP_REQUEST for that same
                 frame's LAN destination."
    REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification, V1.0
             and V2.0, Section 5.1.1"
    DEFVAL   { 1 }
    ::= { lecConfigEntry 13 }
```

```
lecAgingTime OBJECT-TYPE
    SYNTAX      INTEGER ( 10..300 )
    UNITS       "seconds"
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
            "C17 Aging Time.  The maximum time that a LE Client
                 will maintain an entry for a unicast LAN
                 Destination in its LE_ARP cache in the absence
                 of a verification of that relationship.

             This time value is expressed in seconds."
    REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification, V1.0
             and V2.0, Section 5.1.1"
    DEFVAL  { 300 }
    ::= { lecConfigEntry 14 }

lecForwardDelayTime OBJECT-TYPE
    SYNTAX      INTEGER ( 4..30 )
    UNITS       "seconds"
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
            "C18 Forward Delay Time.  The maximum time that a LE
                 Client will maintain an entry for a non-local MAC
                 address in its LE_ARP cache in the absence of a
                 verification of that relationship, as long as the
                 Topology Change flag C19 is true.

             lecForwardDelayTime SHOULD BE less than lecAgingTime.
             When it is not, lecAgingTime governs LE_ARP aging.

             This time value is expressed in seconds."
    REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification, V1.0
             and V2.0, Section 5.1.1"
    DEFVAL  { 15 }
    ::= { lecConfigEntry 15 }

lecExpectedArpResponseTime OBJECT-TYPE
    SYNTAX      INTEGER ( 1..30 )
    UNITS       "seconds"
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
            "C20 Expected LE_ARP Reponse Time.  The maximum time
                 that the LEC expects an LE_ARP_REQUEST/
                 LE_ARP_RESPONSE cycle to take.  Used for retries
                 and verifies.

             This time value is expressed in seconds."
    REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification, V1.0
             and V2.0, Section 5.1.1"
    DEFVAL   { 1 }
    ::= { lecConfigEntry 16 }

lecFlushTimeOut OBJECT-TYPE
    SYNTAX      INTEGER ( 1..4 )
    UNITS       "seconds"
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
            "C21 Flush Time-out.  Time limit to wait to receive a
                 LE_FLUSH_RESPONSE after the LE_FLUSH_REQUEST has
                 been sent before taking recovery action.

             This time value is expressed in seconds."
    REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification, V1.0
             and V2.0, Section 5.1.1"
    DEFVAL   { 4 }
    ::= { lecConfigEntry 17 }

lecPathSwitchingDelay OBJECT-TYPE
    SYNTAX      INTEGER ( 1..8 )
    UNITS       "seconds"
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
            "C22 Path Switching Delay.  The time since sending a
                 frame to the BUS after which the LE Client may
                 assume that the frame has been either discarded
                 or delivered to the recipient.  May be used to
                 bypass the Flush protocol.

             This time value is expressed in seconds."
    REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification, V1.0
             and V2.0, Section 5.1.1"
    DEFVAL   { 6 }
    ::= { lecConfigEntry 18 }

lecLocalSegmentID OBJECT-TYPE
    SYNTAX      INTEGER ( 0..4095 )
    MAX-ACCESS  read-create
    STATUS      deprecated
    DESCRIPTION
            "C23 Local Segment ID.  The segment ID of the emulated
                 LAN.  This is only required for IEEE 802.5 clients
                 that are Source Routing bridges.

             This object has been deprecated and replaced by a pair
             of Config/Actual objects.  It shouldn't be implemented
             except as required for backwards compatibility."
    REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification, V1.0,
             Section 5.1.1"
    ::= { lecConfigEntry 19 }

lecMulticastSendType OBJECT-TYPE
    SYNTAX      INTEGER {
                         bestEffort(1),
                         variableBitRate(2),
                         constantBitRate(3)
                        }
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
            "C24 Multicast Send VCC Type.  Signalling parameter
                 that SHOULD be used by the LE Client when
                 establishing the Multicast Send VCC.

                 This is the method to be used by the LE Client
                 when specifying traffic parameters when it sets
                 up the Multicast Send VCC for this emulated LAN."
    REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification, V1.0
             and V2.0, Section 5.1.1"
    ::= { lecConfigEntry 20 }

lecMulticastSendAvgRate OBJECT-TYPE
    SYNTAX      Integer32
    UNITS       "cells per second"
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
            "C25 Multicast Send VCC AvgRate.  Signalling parameter
                 that SHOULD be used by the LE Client when
                 establishing the Multicast Send VCC.

                 Forward and Backward Sustained Cell Rate to be
                 requested by LE Client when setting up Multicast
                 Send VCC, if using Variable bit rate codings."
    REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification, V1.0
             and V2.0, Section 5.1.1"
    ::= { lecConfigEntry 21 }

lecMulticastSendPeakRate OBJECT-TYPE
    SYNTAX      Integer32
    UNITS       "cells per second"
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
            "C26 Multicast Send VCC PeakRate.  Signalling parameter
                 that SHOULD be used by the LE Client when
                 establishing the Multicast Send VCC.

                 Forward and Backward Peak Cell Rate to be
                 requested by LE Client when setting up the
                 Multicast Send VCC when using either Variable or
                 Constant bit rate codings."
    REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification, V1.0
             and V2.0, Section 5.1.1"
    ::= { lecConfigEntry 22 }

lecConnectionCompleteTimer OBJECT-TYPE
    SYNTAX      INTEGER ( 1..10 )
    UNITS       "seconds"
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
            "C28 Connection Complete Timer.  Optional.  In
                 Connection Establishment this is the time period
                 in which data or a READY_IND message is expected
                 from a Calling Party.

             This time value is expressed in seconds."
    REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification, V1.0
             and V2.0, Section 5.1.1"
    DEFVAL   { 4 }
    ::= { lecConfigEntry 23 }

lecConfigLecsAtmAddress OBJECT-TYPE
    SYNTAX      AtmLaneAddress
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
            "LE Configuration Server ATM address.

             A manually-configured LECS address that this client
             may use in its attempts at auto-configuration."
    REFERENCE
            "LANE 1.0 addendum"
    ::= { lecConfigEntry 24 }

lecInitialControlTimeout OBJECT-TYPE
    SYNTAX      INTEGER ( 1..10 )
    UNITS       "seconds"
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
            "C7i  Initial Control Time-out.

             In LANE V2.0, control time-out consists of an initial
             time-out of C7i seconds, a retry multiplier (or
             logarithmic back-off base) of C7x, and a cumulative
             maximum time-out of C7 seconds."
    REFERENCE
            "ATM Forum LAN Emulation V2.0 LUNI specification,
             Section 5.1.1"
    DEFVAL   { 5 }
    ::= { lecConfigEntry 25 }

lecControlTimeoutMultiplier OBJECT-TYPE
    SYNTAX      INTEGER ( 2..5 )
    UNITS       "seconds"
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
            "C7x  Control Time-out Multiplier.

             In LANE V2.0, control time-out consists of an initial
             time-out of C7i seconds, a retry multiplier (or
             logarithmic back-off base) of C7x, and a cumulative
             maximum time-out of C7 seconds."
    REFERENCE
            "ATM Forum LAN Emulation V2.0 LUNI specification,
             Section 5.1.1"
    DEFVAL   { 2 }
    ::= { lecConfigEntry 26 }

lecV2MaxUnknownFrameCount OBJECT-TYPE
    SYNTAX      INTEGER ( 1..20 )
    UNITS       "frames"
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
            "C10 Maximum Unknown Frame Count.  See the description
                 of lecMaxUnknownFrameTime above.

             This object may be used with both LANE V1.0 clients &
             LANE V2.0 clients.  For LANE V1.0 clients, this value
             should be in the range 1..10, with a default of 1."
    REFERENCE
            "ATM Forum LAN Emulation V2.0 LUNI specification,
             Section 5.1.1"
    DEFVAL   { 10 }
    ::= { lecConfigEntry 27 }

lecConfigLocalSegmentID OBJECT-TYPE
```

```
    SYNTAX      INTEGER ( 0..4095 )
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
            "C23c Local Segment ID (configured).  The segment ID of
                  the emulated LAN.  This is only required for IEEE
                  802.5 clients that are Source Routing bridges.

             A setting which this client will use the next time it
             returns to the Initial State.

             This MIB object will not be overwritten with the new
             value from a LE_{JOIN,CONFIGURE}_RESPONSE.  Instead,
             lecActualLocalSegmentID will be."
    REFERENCE
            "ATM Forum LAN Emulation V2.0 LUNI specification,
             Section 5.1.1"
    ::= { lecConfigEntry 28 }

lecConfigV2Capable OBJECT-TYPE
    SYNTAX        TruthValue
    MAX-ACCESS    read-create
    STATUS        current
    DESCRIPTION
            "C29c  V2 Capable (configured).

             A setting which this client will use the next time it
             returns to the Initial State.

             Indicates whether this client is set up to behave as
             a LANE V2 client.

             This MIB object will not be overwritten with the new
             value from a LE_{JOIN,CONFIGURE}_RESPONSE.  Instead,
             lecActualV2Capable will be."
    REFERENCE
            "ATM Forum LAN Emulation V2.0 LUNI specification,
             Section 5.1.1"
    ::= { lecConfigEntry 29 }

lecConfigSelectiveMulticast OBJECT-TYPE
    SYNTAX        TruthValue
    MAX-ACCESS    read-create
    STATUS        current
    DESCRIPTION
            "C32c  Selective Multicast (configured).

             A setting which this client will use the next time it
             returns to the Initial State.

             Indicates if this client is configured to register to
             receive specific multicast destination MAC addresses.
             Must not be 'true' if lecConfigV2Capable is 'false'.

             This MIB object will not be overwritten with the new
             value from a LE_{JOIN,CONFIGURE}_RESPONSE.  Instead,
             lecActualSelectiveMulticast will be."
    REFERENCE
            "ATM Forum LAN Emulation V2.0 LUNI specification,
             Section 5.1.1"
    ::= { lecConfigEntry 30 }

lecForwardDisconnectTimeout OBJECT-TYPE
    SYNTAX        INTEGER ( 10..300 )
```

```
        UNITS         "seconds"
        MAX-ACCESS    read-create
        STATUS        current
        DESCRIPTION
                "C33  Forward Disconnect Time-out

                 If the LE Client detects that the BUS has had no
                 Multicast Forward VCCs to the LE Client for time
                 Forward Disconnect Time-out, the LE Client MUST
                 disconnect from the BUS and try to re-establish
                 a new connection."
        REFERENCE
                "ATM Forum LAN Emulation V2.0 LUNI specification,
                 Section 5.1.1"
        DEFVAL   { 60 }
        ::= { lecConfigEntry 31 }

    lecConfigLLCMultiplexCapable OBJECT-TYPE
        SYNTAX        TruthValue
        MAX-ACCESS    read-create
        STATUS        current
        DESCRIPTION
                "C34c  LLC Multiplex Capable (configured).

                 A setting which this client will use the next time it
                 returns to the Initial State.

                 Gives a network manager an optional method to disable
                 LLC multiplexing on a LANE V2 client that supports it.
                 Cannot be used to enable LLC multiplexing on products
                 that do not support it.

                 This MIB object will not be overwritten with the new
                 value from a LE_{JOIN,CONFIGURE}_RESPONSE.  Instead,
                 lecActualLLCMultiplexCapable will be."
        REFERENCE
                "ATM Forum LAN Emulation V2.0 LUNI specification,
                 Section 5.1.1"
        ::= { lecConfigEntry 32 }

    lecMinReconfigureDelay OBJECT-TYPE
        SYNTAX        INTEGER ( 1..10000 )
        UNITS         "milliseconds"
        MAX-ACCESS    read-create
        STATUS        current
        DESCRIPTION
                "C37  Minimum Reconfigure Delay.

                 The minimum time that a LE client must wait before
                 retrying reconfiguration.  This setting must be
                 less than or equal to lecMaxReconfigureDelay."
        REFERENCE
                "ATM Forum LAN Emulation V2.0 LUNI specification,
                 Section 5.1.1"
        DEFVAL   { 1 }
        ::= { lecConfigEntry 33 }

    lecMaxReconfigureDelay OBJECT-TYPE
        SYNTAX        INTEGER ( 1..10000 )
        UNITS         "milliseconds"
        MAX-ACCESS    read-create
        STATUS        current
        DESCRIPTION
                "C38  Maximum Reconfigure Delay.
```

```
             The maximum time that a LE client must wait before
             retrying reconfiguration.  This setting must be
             greater than or equal to lecMinReconfigureDelay."
    REFERENCE
            "ATM Forum LAN Emulation V2.0 LUNI specification,
             Section 5.1.1"
    DEFVAL   { 5000 }
    ::= { lecConfigEntry 34 }

lecMaxBusConnectRetries OBJECT-TYPE
    SYNTAX        INTEGER ( 0..2 )
    MAX-ACCESS    read-create
    STATUS        current
    DESCRIPTION
            "C39  Maximum BUS Connect Retries.

             The maximum number of times that a LE Client can try
             to establish the Default Multicast Send VCC before
             returning to the configuration procedure."
    REFERENCE
            "ATM Forum LAN Emulation V2.0 LUNI specification,
             Section 5.1.1"
    DEFVAL   { 1 }
    ::= { lecConfigEntry 35 }

lecTokenRingExplorerExclude OBJECT-TYPE
    SYNTAX        TruthValue
    MAX-ACCESS    read-create
    STATUS        current
    DESCRIPTION
            "C40  Token Ring Explorer Frame Exclude.

             This indicates whether or not the LE Client wishes to
             receive Token Ring explorer frames other than those
             directed to one of its registered MAC addresses.
             Normally, only bridges need to receive such explorer
             frames.  This parameter only applies to token-ring
             ELANs.  This parameter MUST NOT change without
             restarting the Configure phase."
    REFERENCE
            "ATM Forum LAN Emulation V2.0 LUNI specification,
             Section 5.1.1"
    ::= { lecConfigEntry 36 }


-- ------------------------------------------------------------- --

--
--  LAN Emulation Client status group
--

lecStatusTable OBJECT-TYPE
    SYNTAX       SEQUENCE OF LecStatusEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
            "A read-only table containing identification, status,
             and operational information about the LAN Emulation
             Clients this agent manages."
    ::= { leClientMIBObjects 2 }

lecStatusEntry OBJECT-TYPE
    SYNTAX       LecStatusEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
            "Each table entry contains information about one LAN
             Emulation Client."
    AUGMENTS  { lecConfigEntry }
    ::= { lecStatusTable 1 }

LecStatusEntry ::=
    SEQUENCE {

        --
        --  Client identification
        --
        lecIfIndex                      InterfaceIndex,
        lecPrimaryAtmAddress            AtmLaneAddress,
        lecID                           INTEGER,


        --
        --  Client status and protocol version
        --
        lecInterfaceState               LecState,
        lecLastFailureRespCode          INTEGER,
        lecLastFailureState             LecState,
        lecProtocol                     INTEGER,
        lecVersion                      INTEGER,
        lecTopologyChange               TruthValue,


        --
        --  Operational parameters and their source
        --
        lecConfigServerAtmAddress       AtmLaneAddress,
        lecConfigSource                 INTEGER,
        lecActualLanType                LecDataFrameFormat,
        lecActualMaxDataFrameSize       LecDataFrameSize,
        lecActualLanName                DisplayString,
        lecActualLesAtmAddress          AtmLaneAddress,
        lecProxyClient                  TruthValue,


        --
        --  LAN Emulation Version 2.0 additions
        --

        lecActualLocalSegmentID         INTEGER,
        lecActualV2Capable              TruthValue,
        lecElanID                       Integer32,
        lecActualSelectiveMulticast     TruthValue,
        lecActualLLCMultiplexCapable    TruthValue,
        lecPreferredLesAddress          AtmLaneAddress,
        lecStoresServiceCategories      TruthValue

    }

lecIfIndex OBJECT-TYPE
    SYNTAX      InterfaceIndex
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "This object identifies the logical interface number
             ('ifIndex') assigned to this client, and is used to
             identify corresponding rows in the Interfaces MIB.
```

```
                 Note that re-initialization of the management agent
                 may cause a client's 'lecIfIndex' to change."
        ::= { lecStatusEntry 1 }

    lecPrimaryAtmAddress OBJECT-TYPE
        SYNTAX      AtmLaneAddress
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
                "C1  LE Client's ATM Addresses.

                 The primary ATM address of this LAN Emulation Client.
                 This address is used to establish the Control Direct
                 and Multicast Send VCCs, and may also be used to set
                 up Data Direct VCCs.

                 A client may have additional ATM addresses for use
                 with Data Direct VCCs.  These addresses are readable
                 via the lecAtmAddressTable."
        REFERENCE
                "ATM Forum LAN Emulation Over ATM Specification, V1.0
                 and V2.0, Section 5.1.1"
        ::= { lecStatusEntry 2 }

    lecID OBJECT-TYPE
        SYNTAX      INTEGER( 0..65279 )
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
                "C14 LE Client Identifier.  Each LE Client requires a
                     LE Client Identifier (LECID) assigned by the LE
                     Server during the Join phase.  The LECID is placed
                     in control requests by the LE Client and MAY be
                     used for echo suppression on multicast data frames
                     sent by that LE Client.  This value MUST NOT
                     change without terminating the LE Client and
                     returning to the Initial state.  A valid LECID
                     MUST be in the range X'0001' through X'FEFF'.

                 The value of this object is only meaningful for a LEC
                 that is connected to a LES.  For a LEC which does not
                 belong to an emulated LAN, the value of this object
                 is defined to be 0."
        REFERENCE
                "ATM Forum LAN Emulation Over ATM Specification, V1.0
                 and V2.0, Section 5.1.1"
        ::= { lecStatusEntry 3 }

    lecInterfaceState OBJECT-TYPE
        SYNTAX      LecState
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
                "The current state of the LAN Emulation Client.

                 Note that 'ifOperStatus' is defined to be 'up' when,
                 and only when, 'lecInterfaceState' is 'operational'."
        REFERENCE
                "ATM Forum LAN Emulation Over ATM Specification, V1.0,
                 Section 2.3.1"
        ::= { lecStatusEntry 4 }

    lecLastFailureRespCode OBJECT-TYPE
        SYNTAX      INTEGER {
                             none(1),
                             timeout(2),
                             undefinedError(3),
                             versionNotSupported(4),
                             invalidRequestParameters(5),
                             duplicateLanDestination(6),
                             duplicateAtmAddress(7),
                             insufficientResources(8),
                             accessDenied(9),
                             invalidRequesterId(10),
                             invalidLanDestination(11),
                             invalidAtmAddress(12),
                             noConfiguration(13),
                             leConfigureError(14),
                             insufficientInformation(15),

                             --
                             --  LANE V2.0
                             --

                             tlvNotFound(16)
                            }
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
                "Status code from the last failed Configure response
                 or Join response.  Failed responses are those for
                 which the LE_CONFIGURE_RESPONSE / LE_JOIN_RESPONSE
                 frame contains a non-zero code, or fails to arrive
                 within a timeout period.

                 If none of this client's requests have failed, this
                 object has the value 'none'.

                 If the failed response contained a STATUS code that
                 is not defined in the LAN Emulation specification,
                 this object has the value 'undefinedError'.

                 The value 'timeout' is self-explanatory.

                 Other failure codes correspond to those defined in
                 the specification, although they may have different
                 numeric values."
        REFERENCE
                "ATM Forum LAN Emulation Over ATM Specification, V1.0
                 and V2.0, Section 4.2, Table 13"
        ::= { lecStatusEntry 5 }

    lecLastFailureState OBJECT-TYPE
        SYNTAX      LecState
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
                "The state this client was in when it updated the
                 'lecLastFailureRespCode'.

                 If 'lecLastFailureRespCode' is 'none', this object
                 has the value initialState(1)."
        REFERENCE
                "ATM Forum LAN Emulation Over ATM Specification, V1.0,
                 Section 2.3.1"
        ::= { lecStatusEntry 6 }
```

```
lecProtocol OBJECT-TYPE
    SYNTAX      INTEGER ( 1..255 )
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "The LAN Emulation protocol which this client supports,
             and specifies in its LE_JOIN_REQUESTs."
    REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification, V1.0,
             Section 4.2"
    ::= { lecStatusEntry 7 }

lecVersion OBJECT-TYPE
    SYNTAX      INTEGER ( 1..255 )
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "The LAN Emulation protocol version which this client
             supports, and specifies in its LE_JOIN_REQUESTs."
    REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification, V1.0,
             Section 4.2"
    ::= { lecStatusEntry 8 }

lecTopologyChange OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "C19 Topology Change.  Boolean indication that the LE
                 Client is using the Forward Delay Time C18,
                 instead of the Aging Time C17, to age non-local
                 entries in its LE_ARP cache C16.

             For a client which is not connected to the LES, this
             object is defined to have the value 'false'."
    REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification, V1.0
             and V2.0, Section 5.1.1"
    ::= { lecStatusEntry 9 }

lecConfigServerAtmAddress OBJECT-TYPE
    SYNTAX      AtmLaneAddress
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "The ATM address of the LAN Emulation Configuration
             Server (if known) or the empty string (otherwise)."
    ::= { lecStatusEntry 10 }

lecConfigSource OBJECT-TYPE
    SYNTAX      INTEGER {
                         gotAddressViaIlmi(1),
                         usedWellKnownAddress(2),
                         usedLecsPvc(3),
                         didNotUseLecs(4),
                         usedConfiguredAddress(5),
                         configInProgress(6)
                        }
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "Indicates whether this LAN Emulation Client used the
             LAN Emulation Configuration Server, and, if so, what
```

```
                method it used to establish the Configuration Direct
                VCC.

                The value 'configInProgress' indicates configuration
                is in progress, and may be used to troubleshoot LECs
                that get stuck in the configuration phase."
        REFERENCE
               "ATM Forum LAN Emulation Over ATM Specification, V1.0,
                Section 5.2; LANE 1.0 and LEC MIB 1.0 addendums"
        ::= { lecStatusEntry 11 }


    lecActualLanType OBJECT-TYPE
        SYNTAX      LecDataFrameFormat
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
               "C2  LAN Type.

                The data frame format that this LAN Emulation Client
                is using right now.  This may come from

                    *  lecConfigLanType,
                    *  the LAN Emulation Configuration Server, or
                    *  the LAN Emulation Server

                This value is related to 'ifMtu' and 'ifType'.  See
                the LEC management specification for more details."
        REFERENCE
               "ATM Forum LAN Emulation Over ATM Specification, V1.0
                and V2.0, Section 5.1.1"
        ::= { lecStatusEntry 12 }

    lecActualMaxDataFrameSize OBJECT-TYPE
        SYNTAX      LecDataFrameSize
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
               "C3  Maximum Data Frame Size.

                The maximum data frame size that this LAN Emulation
                client is using right now.  This may come from

                    *  lecConfigMaxDataFrameSize,
                    *  the LAN Emulation Configuration Server, or
                    *  the LAN Emulation Server                     "
        REFERENCE
               "ATM Forum LAN Emulation Over ATM Specification, V1.0
                and V2.0, Section 5.1.1"
        ::= { lecStatusEntry 13 }

    lecActualLanName OBJECT-TYPE
        SYNTAX      DisplayString (SIZE( 0..32 ))
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
               "C5  ELAN Name.

                The identity of the emulated LAN which this client
                last joined, or wishes to join.  This may come from

                    *  lecConfigLanName,
                    *  the LAN Emulation Configuration Server, or
                    *  the LAN Emulation Server                     "
    REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification, V1.0
             and V2.0, Section 5.1.1"
    ::= { lecStatusEntry 14 }

lecActualLesAtmAddress OBJECT-TYPE
    SYNTAX      AtmLaneAddress
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "C9  LE Server ATM Address.

             The LAN Emulation Server address currently in use or
             most recently attempted.

             If no LAN Emulation Server attachment has been tried,
             this object's value is the zero-length string."
    REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification, V1.0
             and V2.0, Section 5.1.1"
    ::= { lecStatusEntry 15 }

lecProxyClient OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "C4  Proxy.

             Indicates whether this client is acting as a proxy.

             Proxy clients are allowed to represent unregistered
             MAC addresses and route descriptors, and receive
             copies of LE_ARP_REQUEST frames for such LAN
             Destinations."
    REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification, V1.0
             and V2.0, Section 5.1.1"
    ::= { lecStatusEntry 16 }

lecActualLocalSegmentID OBJECT-TYPE
    SYNTAX      INTEGER ( 0..4095 )
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "C23 Local Segment ID.  The segment ID of the emulated
                 LAN.  This is only required for IEEE 802.5 clients
                 that are Source Routing bridges."
    REFERENCE
            "ATM Forum LAN Emulation V2.0 LUNI specification,
             Section 5.1.1"
    ::= { lecStatusEntry 17 }

lecActualV2Capable OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "C29  V2 Capable.

             If set, the LE Client must exhibit LANE v2 behavior
             where it differs in its requirements from LANE v1
             behavior.  If clear, the LE Client must exhibit LANE
             v1 behavior in those cases.  This variable is set or
```

```
                  cleared according to the presence or absence of the
                  V2 Required FLAGS bit in the LE_JOIN_RESPONSE."
         REFERENCE
                 "ATM Forum LAN Emulation V2.0 LUNI specification,
                  Section 5.1.1"
         ::= { lecStatusEntry 18 }

     lecElanID OBJECT-TYPE
         SYNTAX        Integer32
         MAX-ACCESS    read-only
         STATUS        current
         DESCRIPTION
                 "C31  ELAN-ID.

                  The ELAN-ID to be used in all LANE LLC-multiplexed
                  data frames sent or received by this client.  This
                  object takes the value 0 if the LES is a LANE V1.0
                  LES.  The initial value of this object is 0."
         REFERENCE
                 "ATM Forum LAN Emulation V2.0 LUNI specification,
                  Section 5.1.1"
         ::= { lecStatusEntry 19 }

     lecActualSelectiveMulticast OBJECT-TYPE
         SYNTAX      TruthValue
         MAX-ACCESS  read-only
         STATUS      current
         DESCRIPTION
                 "C32  Selective Multicast.

                  This indicates whether or not the LE Client will
                  register to receive specific multicast destination
                  MAC addresses.  If this flag is set, the LE Client
                  MUST issue a LE_REGISTER_REQUEST for all of the
                  multicast or group MAC adddresses in C15, except
                  for the broadcast MAC address.  If this variable is
                  clear, the LE Client MUST NOT issue a
                  LE_REGISTER_REQUEST for any multicast MAC address,
                  and can expect to receive all multicast frames from
                  the BUS."
         REFERENCE
                 "ATM Forum LAN Emulation V2.0 LUNI specification,
                  Section 5.1.1"
         ::= { lecStatusEntry 20 }

     lecActualLLCMultiplexCapable OBJECT-TYPE
         SYNTAX        TruthValue
         MAX-ACCESS    read-only
         STATUS        current
         DESCRIPTION
                 "C34  LLC Multiplex Capable.

                  If set, this variable indicates that the LE Client
                  can accept and process LLC-multiplexed Data Direct
                  VCCs.  If clear, it indicates that the LE Client
                  will refuse to accept LLC-multiplexed Data Direct
                  VCCs.  This variable MUST be clear if the LE Client
                  is connected to a LANE v1 LES."
         REFERENCE
                 "ATM Forum LAN Emulation V2.0 LUNI specification,
                  Section 5.1.1"
         ::= { lecStatusEntry 21 }

     lecPreferredLesAddress OBJECT-TYPE
```

```
    SYNTAX      AtmLaneAddress
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "C35  Preferred LES Address.

             The ATM address of the preferred LE Server.  This
             variable is set during the Configuration phase by
             copying the contents of the Preferred LES TLV
             received in the LE_CONFIGURE_RESPONSE."
    REFERENCE
            "ATM Forum LAN Emulation V2.0 LUNI specification,
             Section 5.1.1"
    ::= { lecStatusEntry 22 }

lecStoresServiceCategories OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "Indicates whether or not the client saves Service
             Category TLV data from LE_ARP_RESPONSEs and makes
             it available through leArpTable and leRDArpTable."
    REFERENCE
            "LAN Emulation Over ATM Version 2 - LUNI Specification,
             Annex A (LAN Emulation Control Frame TLVs)"
    ::= { lecStatusEntry 23 }


-- ------------------------------------------------------------ --

--
--  LAN Emulation Client - ifIndex Mapping Table
--

lecMappingTable OBJECT-TYPE
    SYNTAX       SEQUENCE OF LecMappingEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
            "A table mapping the 'ifIndex' values of 'aflane8023'
             and 'aflane8025' interfaces to the 'lecIndex' values
             of the corresponding LAN Emulation Clients."
    ::= { leClientMIBObjects 3 }

lecMappingEntry OBJECT-TYPE
    SYNTAX       LecMappingEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
            "Each row describes one ifIndex --> lecIndex mapping."
    INDEX   { ifIndex }
    ::= { lecMappingTable 1 }

LecMappingEntry ::=
    SEQUENCE {
        lecMappingIndex             INTEGER
    }

lecMappingIndex OBJECT-TYPE
    SYNTAX      INTEGER ( 1..2147483647 )
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
```

```
            "The 'lecIndex' of the client which implements the
             specified interface."
    ::= { lecMappingEntry 1 }

-- ------------------------------------------------------------- --

--
--  LAN Emulation Client - Statistics table
--

lecStatisticsTable OBJECT-TYPE
    SYNTAX       SEQUENCE OF LecStatisticsEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
            "An extension table containing traffic statistics for
             all the LAN Emulation Clients this host implements."
    ::= { leClientMIBObjects 4 }

lecStatisticsEntry OBJECT-TYPE
    SYNTAX       LecStatisticsEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
            "Each row in this table contains traffic statistics
             for one LAN Emulation client."
    AUGMENTS  { lecConfigEntry }
    ::= { lecStatisticsTable 1 }

LecStatisticsEntry ::=
    SEQUENCE {
        lecArpRequestsOut           Counter32,
        lecArpRequestsIn            Counter32,
        lecArpRepliesOut            Counter32,
        lecArpRepliesIn             Counter32,
        lecControlFramesOut         Counter32,
        lecControlFramesIn          Counter32,
        lecSvcFailures              Counter32,

        --
        --  LEC MIB Version 2.0 additions
        --

        lecFlowFailures             Counter32,
        lecEchoDiscards             Counter32,
        lecFilteredMulticasts       Counter32
    }

lecArpRequestsOut OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "The number of LE_ARP_REQUESTs sent over the LUNI by
             this LAN Emulation Client."
    ::= { lecStatisticsEntry 1 }

lecArpRequestsIn OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "The number of LE_ARP_REQUESTs received over the LUNI
             by this LAN Emulation Client.
```

```
                 Requests may arrive on the Control Direct VCC or on
                 the Control Distribute VCC, depending upon how the
                 LES is implemented and the chances it has had for
                 learning.  This counter covers both VCCs."
        ::= { lecStatisticsEntry 2 }

    lecArpRepliesOut OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
                "The number of LE_ARP_RESPONSEs sent over the LUNI by
                 this LAN Emulation Client."
        ::= { lecStatisticsEntry 3 }

    lecArpRepliesIn OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
                "The number of LE_ARP_RESPONSEs received over the LUNI
                 by this LAN Emulation Client.  This count includes
                 all such replies, whether solicited or not.

                 Replies may arrive on the Control Direct VCC or on
                 the Control Distribute VCC, depending upon how the
                 LES is implemented.  This counter covers both VCCs."
        ::= { lecStatisticsEntry 4 }

    lecControlFramesOut OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
                "The total number of control packets sent by this
                 LAN Emulation Client over the LUNI."
        ::= { lecStatisticsEntry 5 }

    lecControlFramesIn OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
                "The total number of control packets received by this
                 LAN Emulation Client over the LUNI."
        ::= { lecStatisticsEntry 6 }

    lecSvcFailures OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
                "The total number of

                     *  outgoing LAN Emulation SVCs which this client
                        tried, but failed, to open;

                     *  incoming LAN Emulation SVCs which this client
                        tried, but failed to establish; and

                     *  incoming LAN Emulation SVCs which this client
                        rejected for protocol or security reasons.
```

```
             It does not include SVC failures for LLC-multiplexed
             connections."
    ::= { lecStatisticsEntry 7 }

lecFlowFailures OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "The total number of

                 *  outgoing LLC-multiplexed LAN Emulation flows
                    which this client tried, but failed, to open;

                 *  incoming LLC-multiplexed LAN Emulation flows
                    which this client did not accept / establish.

             Only failures of which the LEC is aware and which are
             clearly LANE-related need to be counted."
    ::= { lecStatisticsEntry 8 }

lecEchoDiscards OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "The number of incoming data frames which were thrown
             away because they were echo frames (e. g., copies of
             this client's own outgoing traffic).

             Such discards are a normal part of LAN Emulation and
             not a cause for alarm.  So this counter is mainly of
             interest for helping to estimate average frame size.

             Frames that are included in this count should not be
             included in ifInDiscards, so as to avoid setting off
             false alarms."
    ::= { lecStatisticsEntry 9 }

lecFilteredMulticasts OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "The number of incoming data frames which were thrown
             away by the client because they were for a multicast
             (or group) address upper-layer software didn't want.

             If this counter indicates that a LEC is being loaded
             down with an excessive number of unwanted multicasts,
             one possible solution is to install LECs and LES/BUS
             pairs with better support for selective multicasting.

             Frames that are included in this count should not be
             included in ifInDiscards, so as to avoid setting off
             false alarms."
    ::= { lecStatisticsEntry 10 }


-- ------------------------------------------------------------ --

--
--  LAN Emulation Client - Server VCC table
--
```

```
lecServerVccTable OBJECT-TYPE
    SYNTAX       SEQUENCE OF LecServerVccEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
            "A table identifying the Control VCCs and Multicast
             VCCs for each LAN Emulation Client which this host
             implements.

             For LANE 2.0 clients, the table identifies Control
             VCCs and Default Multicast Send VCCs."
    ::= { leClientMIBObjects 5 }

lecServerVccEntry OBJECT-TYPE
    SYNTAX       LecServerVccEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
            "Each row in this table describes the Control VCCs
             and Multicast VCCs for one LEC."
    AUGMENTS  { lecConfigEntry }
    ::= { lecServerVccTable 1 }

LecServerVccEntry ::=
    SEQUENCE {

--
--  Note that the 'lec*Interface' objects double as 'connection
--  exists' flags.
--
        lecConfigDirectInterface       LeConnectionInterface,
        lecConfigDirectVpi             VpiInteger,
        lecConfigDirectVci             VciInteger,

        lecControlDirectInterface      LeConnectionInterface,
        lecControlDirectVpi            VpiInteger,
        lecControlDirectVci            VciInteger,

        lecControlDistributeInterface  LeConnectionInterface,
        lecControlDistributeVpi        VpiInteger,
        lecControlDistributeVci        VciInteger,

        lecMulticastSendInterface      LeConnectionInterface,
        lecMulticastSendVpi            VpiInteger,
        lecMulticastSendVci            VciInteger,

--
--  The Multicast Forward variables are not meaningful for LANE
--  V2.0 clients operating in V2.0 mode.
--
        lecMulticastForwardInterface   LeConnectionInterface,
        lecMulticastForwardVpi         VpiInteger,
        lecMulticastForwardVci         VciInteger
    }

lecConfigDirectInterface OBJECT-TYPE
    SYNTAX      LeConnectionInterface
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "The interface associated with the Configuration
             Direct VCC.  If no Configuration Direct VCC exists,
             this object has the value 0.  Otherwise, the objects
```

```
                  ( lecConfigDirectInterface,
                    lecConfigDirectVpi, lecConfigDirectVci )

              identify the circuit."
     REFERENCE
           "ATM Forum LAN Emulation Over ATM Specification, V1.0,
            Section 2.2.2.1.1"
     ::= { lecServerVccEntry 1 }

 lecConfigDirectVpi OBJECT-TYPE
     SYNTAX      VpiInteger
     MAX-ACCESS  read-only
     STATUS      current
     DESCRIPTION
             "If the Configuration Direct VCC exists, this object
              contains the VPI which identifies that VCC at
              the point where it connects to this LE client.

              Otherwise, this object has the value 0."
     REFERENCE
           "ATM Forum LAN Emulation Over ATM Specification, V1.0,
            Section 2.2.2.1.1"
     ::= { lecServerVccEntry 2 }

 lecConfigDirectVci OBJECT-TYPE
     SYNTAX      VciInteger
     MAX-ACCESS  read-only
     STATUS      current
     DESCRIPTION
             "If the Configuration Direct VCC exists, this object
              contains the VCI which identifies that VCC at
              the point where it connects to this LE client.

              Otherwise, this object has the value 0."
     REFERENCE
           "ATM Forum LAN Emulation Over ATM Specification, V1.0,
            Section 2.2.2.1.1"
     ::= { lecServerVccEntry 3 }

 lecControlDirectInterface OBJECT-TYPE
     SYNTAX      LeConnectionInterface
     MAX-ACCESS  read-only
     STATUS      current
     DESCRIPTION
             "The interface associated with the Control Direct VCC.
              If no Control Direct VCC exists, this object has the
              value 0.  Otherwise, the objects

                  ( lecControlDirectInterface,
                    lecControlDirectVpi, lecControlDirectVci )

              identify the circuit."
     REFERENCE
           "ATM Forum LAN Emulation Over ATM Specification, V1.0,
            Section 2.2.2.1.2"
     ::= { lecServerVccEntry 4 }

 lecControlDirectVpi OBJECT-TYPE
     SYNTAX      VpiInteger
     MAX-ACCESS  read-only
     STATUS      current
     DESCRIPTION
             "If the Control Direct VCC exists, this object
```

```
                 contains the VPI which identifies that VCC at
                 the point where it connects to this LE client.

                 Otherwise, this object has the value 0."
        REFERENCE
              "ATM Forum LAN Emulation Over ATM Specification, V1.0,
               Section 2.2.2.1.2"
        ::= { lecServerVccEntry 5 }

    lecControlDirectVci OBJECT-TYPE
        SYNTAX      VciInteger
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
                "If the Control Direct VCC exists, this object
                 contains the VCI which identifies that VCC at
                 the point where it connects to this LE client.

                 Otherwise, this object has the value 0."
        REFERENCE
              "ATM Forum LAN Emulation Over ATM Specification, V1.0,
               Section 2.2.2.1.2"
        ::= { lecServerVccEntry 6 }

    lecControlDistributeInterface OBJECT-TYPE
        SYNTAX      LeConnectionInterface
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
                "The interface associated with the Control Distribute
                 VCC.  If no Control Distribute VCC has been set up to
                 this client, this object has the value 0.  Otherwise,
                 the objects

                     ( lecControlDistributeInterface,
                       lecControlDistributeVpi,
                       lecControlDistributeVci )

                 identify the circuit."
        REFERENCE
              "ATM Forum LAN Emulation Over ATM Specification, V1.0,
               Section 2.2.2.1.3"
        ::= { lecServerVccEntry 7 }

    lecControlDistributeVpi OBJECT-TYPE
        SYNTAX      VpiInteger
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
                "If the Control Distribute VCC exists, this object
                 contains the VPI which identifies that VCC at the
                 point where it connects to this LE client.

                 Otherwise, this object has the value 0."
        REFERENCE
              "ATM Forum LAN Emulation Over ATM Specification, V1.0,
               Section 2.2.2.1.3"
        ::= { lecServerVccEntry 8 }

    lecControlDistributeVci OBJECT-TYPE
        SYNTAX      VciInteger
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
```

```
            "If the Control Distribute VCC exists, this object
             contains the VCI which identifies that VCC at the
             point where it connects to this LE client.

             Otherwise, this object contains the value 0."
    REFERENCE
          "ATM Forum LAN Emulation Over ATM Specification, V1.0,
           Section 2.2.2.1.3"
    ::= { lecServerVccEntry 9 }

lecMulticastSendInterface OBJECT-TYPE
    SYNTAX      LeConnectionInterface
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "The interface associated with the Multicast Send
             VCC.  If no Multicast Send VCC exists, this object
             has the value 0.  Otherwise, the objects

                 ( lecMulticastSendInterface,
                   lecMulticastSendVpi, lecMulticastSendVci )

             identify the circuit."
    REFERENCE
          "ATM Forum LAN Emulation Over ATM Specification, V1.0,
           Section 2.2.2.2.2"
    ::= { lecServerVccEntry 10 }

lecMulticastSendVpi OBJECT-TYPE
    SYNTAX      VpiInteger
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "If the Multicast Send VCC exists, this object
             contains the VPI which identifies that VCC at
             the point where it connects to this LE client.

             Otherwise, this object has the value 0."
    REFERENCE
          "ATM Forum LAN Emulation Over ATM Specification, V1.0,
           Section 2.2.2.2.2"
    ::= { lecServerVccEntry 11 }

lecMulticastSendVci OBJECT-TYPE
    SYNTAX      VciInteger
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "If the Multicast Send VCC exists, this object
             contains the VCI which identifies that VCC at
             the point where it connects to this LE client.

             Otherwise, this object has the value 0."
    REFERENCE
          "ATM Forum LAN Emulation Over ATM Specification, V1.0,
           Section 2.2.2.2.2"
    ::= { lecServerVccEntry 12 }

lecMulticastForwardInterface OBJECT-TYPE
    SYNTAX      LeConnectionInterface
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "For a LANE V1.0 client:
             -----------------------
             The interface associated with the Multicast Forward
             VCC.  If no Multicast Forward VCC has been set up to
             this client, this object has the value 0.  Otherwise,
             the objects

                 ( lecMulticastForwardInterface,
                   lecMulticastForwardVpi,
                   lecMulticastForwardVci )

             identify the circuit.


             For a LANE V2.0 client in LANE 2.0 mode:
             ----------------------------------------
             The value 0.  There may not be a single VCC that can
             be identified as the 'default'."
    REFERENCE
          "ATM Forum LAN Emulation Over ATM Specification, V1.0,
           Section 2.2.2.2.3"
    ::= { lecServerVccEntry 13 }

lecMulticastForwardVpi OBJECT-TYPE
    SYNTAX      VpiInteger
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "For a LANE V1.0 client:
             -----------------------
             If the Multicast Forward VCC exists, this object
             contains the VPI which identifies that VCC at the
             point where it connects to this LE client.

             Otherwise, this object has the value 0.


             For a LANE V2.0 client in LANE 2.0 mode:
             ----------------------------------------
             The value 0."
    REFERENCE
          "ATM Forum LAN Emulation Over ATM Specification, V1.0,
           Section 2.2.2.2.3"
    ::= { lecServerVccEntry 14 }

lecMulticastForwardVci OBJECT-TYPE
    SYNTAX      VciInteger
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "For a LANE V1.0 client:
             -----------------------
             If the Multicast Forward VCC exists, this object
             contains the VCI which identifies that VCC at the
             point where it connects to this LE client.

             Otherwise, this object has the value 0.


             For a LANE V2.0 client in LANE 2.0 mode:
             ----------------------------------------
             The value 0."
    REFERENCE
          "ATM Forum LAN Emulation Over ATM Specification, V1.0,
           Section 2.2.2.2.3"
    ::= { lecServerVccEntry 15 }

-- ------------------------------------------------------------- --

--
--  LAN Emulation Client - ATM Addresses table
--

lecAtmAddressTable OBJECT-TYPE
    SYNTAX       SEQUENCE OF LecAtmAddressEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
            "A table listing all of the ATM addresses belonging
             to this host's LAN Emulation Clients."
    ::= { leClientMIBObjects 6 }

lecAtmAddressEntry OBJECT-TYPE
    SYNTAX       LecAtmAddressEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
            "Each row in this table describes one of a client's
             ATM addresses."
    INDEX   { lecIndex, lecAtmAddress }
    ::= { lecAtmAddressTable 1 }

LecAtmAddressEntry ::=
    SEQUENCE {
        lecAtmAddress               AtmLaneAddress,
        lecAtmAddressStatus         RowStatus,

        --
        --  New for LANE 2.0
        --

        lecAtmAddressMuxed          INTEGER
    }

lecAtmAddress OBJECT-TYPE
    SYNTAX      AtmLaneAddress
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
            "The ATM address this row describes.  This could be
             either a primary address or a secondary address."
    ::= { lecAtmAddressEntry 1 }

lecAtmAddressStatus OBJECT-TYPE
    SYNTAX      RowStatus
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
            "Used to create and delete rows in this table.

             A management station may not change the status of a
             primary ATM address to 'notInService' or 'destroy'
             unless the client's ifAdminStatus is 'down', and the
             client's lecInterfaceState is 'initialState'.

             Secondary ATM addresses may be destroyed at any time
             permitted by the agent."
    ::= { lecAtmAddressEntry 2 }
```

```
lecAtmAddressMuxed OBJECT-TYPE
    SYNTAX      INTEGER {
                         muxed(1),
                         notMuxed(2),
                         both(3),
                         none(4)
                        }
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
            "Indicates what type(s) of connections the LEC is
             willing to establish at this ATM address."
    ::= { lecAtmAddressEntry 3 }


-- ------------------------------------------------------------- --

--
--  LAN Emulation Client Registered MAC Addresses group
--

lecMacAddressTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF LecMacAddressEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
            "This table contains entries for all of the registered
             MAC addresses belonging to this agent's LE clients.

             For all LANE clients, this includes

             C6  Local Unicast MAC Address(es).  Each LE Client has
                 zero or more local unicast MAC addresses.  In an
                 operational LE Client, every address in this
                 variable MUST have been registered with the LE
                 Server.  Two LE CLients joined to the same
                 emulated LAN MUST NOT have the same local unicast
                 MAC address.  A LE Client's MAC addresses may
                 change during normal operations.  When answering
                 a LE_ARP_REQUEST for any address in this list, the
                 'Remote Address' bit in the FLAGS field of the
                 LE_ARP_RESPONSE MUST be clear.

            For a LANE V2.0 client, this also includes <registered>
            multicast MAC addresses in C15."
    REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification, V1.0
             and V2.0, Section 5.1.1"
    ::= { leClientMIBObjects 7 }

lecMacAddressEntry OBJECT-TYPE
    SYNTAX      LecMacAddressEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
            "Each table row describes a (MAC address, ATM address)
             pair registered for a particular client."
    INDEX   { lecIndex, lecMacAddress }
    ::= { lecMacAddressTable 1 }

LecMacAddressEntry ::=
    SEQUENCE {

        lecMacAddress                   MacAddress,
        lecMacAddressAtmBinding         AtmLaneAddress,

        --
        --  New for LANE 2.0
        --

        lecMacAddressMuxedAtmBinding    AtmLaneAddress,
        lecMacAddressServiceCategory    INTEGER

    }

lecMacAddress OBJECT-TYPE
    SYNTAX      MacAddress
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
            "A unicast or multicast MAC address that this LEC has
             registered with its LES.

             Note that the LANE V1.0 specification does not allow
             V1.0 clients to register multicast addresses."
    ::= { lecMacAddressEntry 1 }

lecMacAddressAtmBinding OBJECT-TYPE
    SYNTAX      AtmLaneAddress
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "The non-multiplexed ATM address, if any, registered
             for 'lecMacAddress'."
    ::= { lecMacAddressEntry 2 }

lecMacAddressMuxedAtmBinding OBJECT-TYPE
    SYNTAX      AtmLaneAddress
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "The LLC-multiplexed ATM address, if any, registered
             for 'lecMacAddress'."
    ::= { lecMacAddressEntry 3 }

lecMacAddressServiceCategory OBJECT-TYPE
    SYNTAX      INTEGER ( 0..2147483647 )
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "A bitmap describing the service categories which the
             LEC supports for this MAC address.

             The bitmap is larger than the LANE 2.0 specification
             requires, in case room is ever needed for expansion."
    REFERENCE
            "LAN Emulation Over ATM Version 2 - LUNI Specification,
             Annex A (LAN Emulation Control Frame TLVs)"
    ::= { lecMacAddressEntry 4 }



-- ------------------------------------------------------------ --

--
--  LAN Emulation Client Registered Route Descriptors group
--
```

```
lecRouteDescrTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF LecRouteDescrEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
            "This table lists all of the Local Route Descriptors
             which are registered by this host's LAN Emulation
             Clients.

             C8  Local Route Descriptor(s).  Local Route
                 Descriptors exist only for source-routed IEEE
                 802.5 LE Clients that are Source-Route Bridges.
                 All Local Route Descriptors in any given emulated
                 LAN MUST be unique.  A LE Client MAY have zero or
                 more Local Route Descriptors and these Local Route
                 Descriptors MAY change during normal operation.
                 In an operational LE Client, every Local Route
                 Descriptor in C8 MUST have been registered with
                 the LE Server.  When answering a LE_ARP_REQUEST
                 for any address in this list, the 'Remote Address'
                 bit in the FLAGS field of the LE_ARP_RESPONSE MUST
                 be clear."
    REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification, V1.0
             and V2.0, Section 5.1.1"
    ::= { leClientMIBObjects 8 }

lecRouteDescrEntry OBJECT-TYPE
    SYNTAX      LecRouteDescrEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
            "Each table row describes a (Route Descriptor, ATM
             address) pair registered for a particular client."
    INDEX   { lecIndex, lecRouteDescrSegmentID,
                        lecRouteDescrBridgeNumber }
    ::= { lecRouteDescrTable 1 }

LecRouteDescrEntry ::=
    SEQUENCE {

        lecRouteDescrSegmentID          INTEGER,
        lecRouteDescrBridgeNumber       INTEGER,
        lecRouteDescrAtmBinding         AtmLaneAddress,

        --
        --  New for LANE 2.0
        --

        lecRouteDescrMuxedAtmBinding    AtmLaneAddress,
        lecRouteDescrServiceCategory    INTEGER

    }

lecRouteDescrSegmentID OBJECT-TYPE
    SYNTAX      INTEGER( 0..4095 )
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
            "The LAN ID portion of the IEEE 802.5 route descriptor
             associated with this conceptual row."
    ::= { lecRouteDescrEntry 1 }

lecRouteDescrBridgeNumber OBJECT-TYPE
```

```
    SYNTAX      INTEGER( 0..15 )
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
            "The Bridge Number portion of the IEEE 802.5 route
             descriptor associated with this conceptual row."
    ::= { lecRouteDescrEntry 2 }

lecRouteDescrAtmBinding OBJECT-TYPE
    SYNTAX      AtmLaneAddress
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "The non-multiplexed ATM address, if any, registered
             for the IEEE 802.5 route descriptor
             ('lecRouteDescrSegmentID',
              'lecRouteDescrBridgeNumber')."
    ::= { lecRouteDescrEntry 3 }

lecRouteDescrMuxedAtmBinding OBJECT-TYPE
    SYNTAX      AtmLaneAddress
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "The LLC-multiplexed ATM address, if any, registered
             for the IEEE 802.5 route descriptor
             ('lecRouteDescrSegmentID',
              'lecRouteDescrBridgeNumber')."
    ::= { lecRouteDescrEntry 4 }

lecRouteDescrServiceCategory OBJECT-TYPE
    SYNTAX      INTEGER ( 0..2147483647 )
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "A bitmap describing the service categories which the
             LEC supports for this route descriptor.

             The bitmap is larger than the LANE 2.0 specification
             requires, in case room is ever needed for expansion."
    REFERENCE
            "LAN Emulation Over ATM Version 2 - LUNI Specification,
             Annex A (LAN Emulation Control Frame TLVs)"
    ::= { lecRouteDescrEntry 5 }


-- ------------------------------------------------------------- --

--
--  LAN Emulation Client ARP cache group - MAC addresses
--

leArpTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF LeArpEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
            "This table provides access to an ATM LAN Emulation
             Client's MAC-to-ATM ARP cache.  It contains entries
             for unicast addresses and for the broadcast address.

             For LANE V2.0 clients whose C32 Selective Multicast
             flag is set, this table will also contain multicast
             address entries.
             C16 LE_ARP Cache.  A table of entries, each of which
                 establishes a relationship between a LAN
                 Destination external to the LE Client and the
                 ATM address to which data frames for that LAN
                 Destination will be sent."
    REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification,
             Section 5.1.1"
    ::= { leClientMIBObjects 9 }

leArpEntry OBJECT-TYPE
    SYNTAX      LeArpEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
            "An ATM LAN Emulation ARP cache entry containing
             information about the binding of one MAC address
             to one ATM address."
    INDEX   { lecIndex, leArpMacAddress }
    ::= { leArpTable 1 }

LeArpEntry ::=
    SEQUENCE {

        leArpMacAddress             MacAddress,
        leArpAtmAddress             AtmLaneAddress,
        leArpIsRemoteAddress        TruthValue,
        leArpEntryType              LeArpTableEntryType,
        leArpRowStatus              RowStatus,

        --
        --  New for LANE 2.0
        --

        leArpMuxedAtmAddress        AtmLaneAddress,
        leArpServiceCategory        INTEGER

    }

leArpMacAddress OBJECT-TYPE
    SYNTAX      MacAddress
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
            "The MAC address for which this cache entry provides
             a translation.  Since ATM LAN Emulation uses an ARP
             protocol to locate the Broadcast and Unknown Server,
             the value of this object could be the broadcast MAC
             address.  Starting with LANE Version 2.0, the value
             could also be a multicast or group MAC address.

             Unicast MAC addresses should be unique within any
             given ATM Emulated LAN.  However, there's no
             requirement that they be unique across disjoint
             emulated LANs."
    ::= { leArpEntry 1 }

leArpAtmAddress OBJECT-TYPE
    SYNTAX      AtmLaneAddress
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
            "The non-multiplexed LE Client or Broadcast/Multicast
```

```
                  Service ATM address corresponding to the MAC address
                  'leArpMacAddress'.

                  This value may be determined through the use of the
                  LE_ARP procedure, through source address learning,
                  or through other mechanisms.

                  Some agents may provide write access to this object,
                  as part of their support for 'static' LE_ARP entries.
                  The effect of attempting to write an ATM address to
                  a 'learned' row is explicitly undefined.  Agents may
                  disallow the write, accept the write and change the
                  row's type, or even accept the write as-is."
         ::= { leArpEntry 2 }

     leArpIsRemoteAddress OBJECT-TYPE
         SYNTAX      TruthValue
         MAX-ACCESS  read-only
         STATUS      current
         DESCRIPTION
                 "Indicates whether this entry is for a local or remote
                  MAC address.

                  In this context, 'local' means 'a MAC address that is
                  local to the remote client', as opposed to 'one of my
                  addresses'.

                      true(1)
                          The address is believed to be remote - or its
                          local/remote status is unknown.

                          For an entry created via the LE_ARP mechanism,
                          this corresponds to the 'Remote address' flag
                          being set in the LE_ARP_RESPONSE.

                          During Topology Change periods, remote LE_ARP
                          entries generally age out faster than others.
                          Specifically, they are subject to the Forward
                          Delay Time as well as to the Aging Time.

                      false(2)
                          The address is believed to be local - that is
                          to say, registered with the LES by the client
                          whose ATM address is leArpAtmAddress (and/or
                          leArpMuxedAtmAddress).

                          For an entry created via the LE_ARP mechanism,
                          this corresponds to the 'Remote address' flag
                          being cleared in the LE_ARP_RESPONSE."
         ::= { leArpEntry 3 }

     leArpEntryType OBJECT-TYPE
         SYNTAX      LeArpTableEntryType
         MAX-ACCESS  read-create
         STATUS      current
         DESCRIPTION
                 "Indicates how this LE_ARP table entry was created and
                  whether it is aged."
         DEFVAL   { staticVolatile }
         ::= { leArpEntry 4 }

     leArpRowStatus OBJECT-TYPE
         SYNTAX      RowStatus
         MAX-ACCESS  read-create
```

```
    STATUS      current
    DESCRIPTION
            "Lets users create and delete LE_ARP cache entries,
             on systems that support this capability.

             Normally clients will maintain their LE_ARP caches
             automatically.  However, managers may occasionally
             want to create 'static' LE_ARP cache entries.

             Rules

               1.  Before a new row can become 'active', values
                   must be supplied for the columnar objects
                   leArpAtmAddress and leArpEntryType.

                   The value for leArpAtmAddress may be a zero-
                   length address if there is an ATM address in
                   the leArpMuxedAtmAddress, and the LE_ARP row
                   belongs to a LEC running LANE 2.0 or higher.

               2.  It is not necessary to set leArpRowStatus to
                   'notInService' in order to modify a writable
                   object in the same conceptual row.

               3.  LE_ARP entries whose status is 'notReady' or
                   'notInService' will not be used to translate
                   LAN Destinations to ATM Addresses.  However,
                   clients may overwrite such entries with data
                   obtained from other sources.  For example, a
                   client which needed to transmit a data frame
                   to a given MAC address (or via a given Route
                   Descriptor) might find that its LE_ARP entry
                   for the LAN Destination is 'notInService' or
                   'notReady'.  The client might then act as if
                   the LAN Destination was unknown - generating
                   a LE_ARP_REQUEST, flooding the data frame to
                   the BUS, and creating a new, 'active' LE_ARP
                   cache entry based on the LE_ARP_RESPONSE."
    ::= { leArpEntry 5 }

leArpMuxedAtmAddress OBJECT-TYPE
    SYNTAX      AtmLaneAddress
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
            "The LLC-multiplexed LE Client or Broadcast/Multicast
             Service ATM address corresponding to the MAC address
             'leArpMacAddress'.

             This value may be determined through the use of the
             LE_ARP procedure, through source address learning,
             or through other mechanisms.

             Some agents may provide write access to this object,
             as part of their support for 'static' LE_ARP entries.
             The effect of attempting to write an ATM address to
             a 'learned' row is explicitly undefined.  Agents may
             disallow the write, accept the write and change the
             row's type, or even accept the write as-is."
    ::= { leArpEntry 6 }

leArpServiceCategory OBJECT-TYPE
    SYNTAX      INTEGER ( 0..2147483647 )
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
            "A bitmap describing the service categories which the
             LEC 'lecIndex' supports.

             This value normally comes from the optional 'Service-
             Category' TLV in LANE 2.0 LE_ARP_RESPONSEs.

             The bitmap is larger than the LANE 2.0 specification
             requires, in case room is ever needed for expansion.

             If service categories aren't applicable, aren't known,
             or the implementor does not want to provide memory to
             store them (e.g., because they would only be used for
             SNMP), this object should be set to the value 0.

             See also the object 'lecStoresServiceCategories'."
    REFERENCE
            "LAN Emulation Over ATM Version 2 - LUNI Specification,
             Annex A (LAN Emulation Control Frame TLVs)"
    ::= { leArpEntry 7 }



-- ------------------------------------------------------------- --

--
--  LAN Emulation Client ARP cache group - Route Descriptors
--

leRDArpTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF LeRDArpEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
            "This table provides access to an ATM LAN Emulation
             Client's Route Descriptor-to-ATM ARP cache.

             Implementation of this table is optional for hosts
             that do not support emulated IEEE 802.5 Token Ring
             LANs, and mandatory for hosts which do.

             C16 LE_ARP Cache.  A table of entries, each of which
                 establishes a relationship between a LAN
                 Destination external to the LE Client and the
                 ATM address to which data frames for that LAN
                 Destination will be sent."
    REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification,
             Section 5.1.1"
    ::= { leClientMIBObjects 10 }

leRDArpEntry OBJECT-TYPE
    SYNTAX      LeRDArpEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
            "An ATM LAN Emulation ARP cache entry containing
             information about the binding of one IEEE 802.5
             route descriptor to one ATM address."
    INDEX   { lecIndex, leRDArpSegmentID, leRDArpBridgeNumber }
    ::= { leRDArpTable 1 }

LeRDArpEntry ::=
    SEQUENCE {
        leRDArpSegmentID              INTEGER,
        leRDArpBridgeNumber           INTEGER,
        leRDArpAtmAddress             AtmLaneAddress,
        leRDArpEntryType              LeArpTableEntryType,
        leRDArpRowStatus              RowStatus,

        --
        --  New for LANE 2.0
        --

        leRDArpMuxedAtmAddress        AtmLaneAddress,
        leRDArpServiceCategory        INTEGER,
        leRDArpIsRemoteDescriptor     TruthValue
    }

leRDArpSegmentID OBJECT-TYPE
    SYNTAX      INTEGER ( 0..4095 )
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
            "The LAN ID portion of the IEEE 802.5 route descriptor
             associated with this ARP cache entry."
    ::= { leRDArpEntry 1 }

leRDArpBridgeNumber OBJECT-TYPE
    SYNTAX      INTEGER ( 0..15 )
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
            "The Bridge Number portion of the IEEE 802.5 route
             descriptor associated with this ARP cache entry."
    ::= { leRDArpEntry 2 }

leRDArpAtmAddress OBJECT-TYPE
    SYNTAX      AtmLaneAddress
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
            "The non-multiplexed ATM address of the LAN Emulation
             Client which is associated with the route descriptor
             ('leRDArpSegmentID', 'leRDArpBridgeNumber').

             This value may be determined through the use of the
             LE_ARP procedure, through source address learning,
             or through other mechanisms.

             Some agents may provide write access to this object,
             as part of their support for 'static' LE_ARP entries.
             The effect of attempting to write an ATM address to
             a 'learned' row is explicitly undefined.  Agents may
             disallow the write, accept the write and change the
             row's type, or even accept the write as-is."
    ::= { leRDArpEntry 3 }

leRDArpEntryType OBJECT-TYPE
    SYNTAX      LeArpTableEntryType
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
            "Indicates how this LE_ARP table entry was created and
             whether it is aged."
    DEFVAL   { staticVolatile }
    ::= { leRDArpEntry 4 }
```

```
leRDArpRowStatus OBJECT-TYPE
    SYNTAX      RowStatus
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
            "Lets users create and delete LE_ARP cache entries,
             on systems that support this capability.

             Normally clients will maintain their LE_ARP caches
             automatically.  However, managers may occasionally
             want to create 'static' LE_ARP cache entries.

             Rules

               1.  Before a new row can become 'active', values
                   must be supplied for the columnar objects
                   leRDArpAtmAddress and leRDArpEntryType.

               2.  It is not necessary to set leRDArpRowStatus to
                   'notInService' in order to modify a writable
                   object in the same conceptual row.

               3.  LE_ARP entries whose status is 'notReady' or
                   'notInService' will not be used to translate
                   LAN Destinations to ATM Addresses.  However,
                   clients may overwrite such entries with data
                   obtained from other sources.  For example, a
                   client which needed to transmit a data frame
                   to a given MAC address (or via a given Route
                   Descriptor) might find that its LE_ARP entry
                   for the LAN Destination is 'notInService' or
                   'notReady'.  The client might then act as if
                   the LAN Destination was unknown - generating
                   a LE_ARP_REQUEST, flooding the data frame to
                   the BUS, and creating a new, 'active' LE_ARP
                   cache entry based on the LE_ARP_RESPONSE."
    ::= { leRDArpEntry 5 }

leRDArpMuxedAtmAddress OBJECT-TYPE
    SYNTAX      AtmLaneAddress
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
            "The LLC-multiplexed ATM address of the LAN Emulation
             Client which is associated with the route descriptor
             ('leRDArpSegmentID', 'leRDArpBridgeNumber').

             This value may be determined through the use of the
             LE_ARP procedure, through source address learning,
             or through other mechanisms.

             Some agents may provide write access to this object,
             as part of their support for 'static' LE_ARP entries.
             The effect of attempting to write an ATM address to
             a 'learned' row is explicitly undefined.  Agents may
             disallow the write, accept the write and change the
             row's type, or even accept the write as-is."
    ::= { leRDArpEntry 6 }

leRDArpServiceCategory OBJECT-TYPE
    SYNTAX      INTEGER ( 0..2147483647 )
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
            "A bitmap describing the service categories which the
             LEC 'lecIndex'supports.

             This value normally comes from the optional 'Service-
             Category' TLV in LANE 2.0 LE_ARP_RESPONSEs.

             The bitmap is larger than the LANE 2.0 specification
             requires, in case room is ever needed for expansion.

             If service categories aren't applicable, aren't known,
             or the implementor does not want to provide memory to
             store them (e.g., because they would only be used for
             SNMP), this object should be set to the value 0.

             See also the object 'lecStoresServiceCategories'."
    REFERENCE
            "LAN Emulation Over ATM Version 2 - LUNI Specification,
             Annex A (LAN Emulation Control Frame TLVs)"
    ::= { leRDArpEntry 7 }

leRDArpIsRemoteDescriptor OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "Indicates whether this entry is for a local or remote
             route descriptor.

             In this context, 'local' means 'a route descriptor
             that is local to the remote client', as opposed to
             'one of my route descriptors'.

                 true(1)
                     The route descriptor is believed to be remote
                     - or its local/remote status is unknown.

                     For an entry created via the LE_ARP mechanism,
                     this corresponds to the 'Remote address' flag
                     being set in the LE_ARP_RESPONSE.

                 false(2)
                     The route descriptor is believed to be local
                     - that is to say, registered with the LES by
                     the client whose address is leRDArpAtmAddress
                     (and/or leRDArpMuxedATMAddress).

                     For an entry created via the LE_ARP mechanism,
                     this corresponds to the 'Remote address' flag
                     being cleared in the LE_ARP_RESPONSE."
    ::= { leRDArpEntry 8 }



-- ------------------------------------------------------------ --

--
--  LAN Emulation Client - Multicast Forward VCC table
--

lecMcForwardTable OBJECT-TYPE
    SYNTAX       SEQUENCE OF LecMcForwardEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
            "A table identifying Multicast Forward VCCs for LAN
             Emulation Version 2.0 clients."
    ::= { leClientMIBObjects 11 }

lecMcForwardEntry OBJECT-TYPE
    SYNTAX       LecMcForwardEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
            "Each row represents one Multicast Forward VCC."
    INDEX     { lecIndex,
                lecMcForwardInterface,
                lecMcForwardVpi,
                lecMcForwardVci }
    ::= { lecMcForwardTable 1 }

LecMcForwardEntry ::=
    SEQUENCE {
        lecMcForwardInterface       LeConnectionInterface,
        lecMcForwardVpi             VpiInteger,
        lecMcForwardVci             VciInteger,
        lecMcForwardVerification    INTEGER
    }

lecMcForwardInterface OBJECT-TYPE
    SYNTAX      LeConnectionInterface
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
            "The interface associated with the Multicast Forward
             VCC."
    ::= { lecMcForwardEntry 1 }

lecMcForwardVpi OBJECT-TYPE
    SYNTAX      VpiInteger
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
            "The VPI which identifies the Multicast Forward VCC
             at the point where it connects to the LANE client."
    ::= { lecMcForwardEntry 2 }

lecMcForwardVci OBJECT-TYPE
    SYNTAX      VciInteger
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
            "The VCI which identifies the Multicast Forward VCC
             at the point where it connects to the LANE client."
    ::= { lecMcForwardEntry 3 }

lecMcForwardVerification OBJECT-TYPE
    SYNTAX      INTEGER {
                         none(1),
                         implicit(2),
                         explicit(3)
                        }
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "Indicates whether the calling address for this VCC
             has been verified.
```

```
                  none      -  No verification

                  implicit  -  Calling party address was same as
                               BUS address

                  explicit  -  Client used LE_VERIFY protocol

              Also serves double-duty as a 'row exists' flag."
    ::= { lecMcForwardEntry 4 }


-- ------------------------------------------------------------ --

--
--  LAN Emulation Client - LE_ARP Response Group
--
--
--  These optional objects let a network manager find out the ATM
--  addresses associated with a proxy client's "C27 Remote Unicast
--  MAC Address(es)" and "C30 Remote Route Descriptor(s)".
--
--  They can also be used to find out the ATM addresses associated
--  with a client's "C6 Local Unicast MAC Address(es)" and "C8
--  Local Route Descriptor(s)".
--
--
--  In the interests of efficiency and of not creating a big table
--  that's the size of a bridge's forwarding database, the objects
--  operate on a request-response model:
--
--      1.  The network management system does a SET operation
--          indicating the specific MAC address or segment ID/
--          bridge number, and the specific client.
--
--      2.  If the agent's MIB handler accepts the SET request,
--          it triggers a query and updates the lecMacQuery* or
--          lecRdQuery* objects.
--
--      3.  The network management system does a GET operation
--          to read the results.  Ideally it should GET all of
--          the result objects in one operation and check that
--          the result keys are the same as the query keys, to
--          handle the case of simultaneous use by another NMS.
--

lecQueryObjects  OBJECT IDENTIFIER ::= { leClientMIBObjects 12 }




--
--  MAC address queries
--

lecMacQueryObjects  OBJECT IDENTIFIER ::= { lecQueryObjects 1 }

lecMacQueryLecIndex OBJECT-TYPE
    SYNTAX       INTEGER ( 0..2147483647 )
    MAX-ACCESS   read-write
    STATUS       current
    DESCRIPTION
            "The 'lecIndex' of the LEC whose MAC address binding
             the network management station wants to look up.
```

```
                When written:  Triggers a new query.

                When read:  Identifies the last LEC queried.  (This
                value may be 0 if there have been no queries, or if
                a LEC has been deleted.)"
         ::= { lecMacQueryObjects 1 }

     lecMacQueryAddress OBJECT-TYPE
         SYNTAX       MacAddress
         MAX-ACCESS   read-write
         STATUS       current
         DESCRIPTION
                "The MAC address whose binding the network management
                 station wants to look up.

                 When written:  Triggers a new query.  (Note that the
                 SNMP agent is free to reject multicast and broadcast
                 addresses with a 'badValue' error.)

                 When read:  Returns the last MAC address queried, or
                 all zeroes if lecMacQueryLecIndex is 0."
         ::= { lecMacQueryObjects 2 }

     lecMacQueryStatus OBJECT-TYPE
         SYNTAX       INTEGER {
                               none(1),
                               unsupported(2),
                               local(3),
                               remote(4),
                               unknown(5)

                              }
         MAX-ACCESS   read-only
         STATUS       current
         DESCRIPTION
                "The status of the last query - one of the following:

                     none(1)
                        No queries have been made; therefore, other
                        results values are invalid.

                     unsupported(2)
                        Queries are not supported for this LEC.

                     local(3)
                        The MAC address is one of the client's local
                        MAC addresses.

                     remote(4)
                        The MAC address is one of the client's remote
                        MAC addresses.

                     unknown(5)
                        The MAC address does not belong (is not known
                        to belong) to the client."
         ::= { lecMacQueryObjects 3 }

     lecMacQueryAtmAddress OBJECT-TYPE
         SYNTAX      AtmLaneAddress
         MAX-ACCESS  read-only
         STATUS      current
         DESCRIPTION
                "The non-multiplexed ATM address which this client --
                 i.e., client 'lecMacQueryLecIndex' -- would use in a
                 LE_ARP_RESPONSE for 'lecMacQueryAddress'.

                 This object is only meaningful when lecMacQueryStatus
                 is 'localMacAddress' or 'remoteMacAddress'."
         ::= { lecMacQueryObjects 4 }

     lecMacQueryMuxedAtmAddress OBJECT-TYPE
         SYNTAX      AtmLaneAddress
         MAX-ACCESS  read-only
         STATUS      current
         DESCRIPTION
                "The LLC-multiplexed ATM address which this client --
                 i.e., client 'lecMacQueryLecIndex' -- would use in a
                 LE_ARP_RESPONSE for 'lecMacQueryAddress'.

                 This object is only meaningful when lecMacQueryStatus
                 is 'localMacAddress' or 'remoteMacAddress'."
         ::= { lecMacQueryObjects 5 }

     lecMacQueryServiceCategory OBJECT-TYPE
         SYNTAX      INTEGER ( 0..2147483647 )
         MAX-ACCESS  read-only
         STATUS      current
         DESCRIPTION
                "The Service Category TLV that this client would use
                 in a LE_ARP_RESPONSE for 'lecMacQueryAddress'.

                 If this client wouldn't return a Service Category TLV
                 in the LE_ARP_RESPONSE, this object should be set to
                 zero.

                 The bitmap is larger than the LANE 2.0 specification
                 requires, in case room is ever needed for expansion.

                 This object is only meaningful when lecMacQueryStatus
                 is 'localMacAddress' or 'remoteMacAddress'."
         REFERENCE
                "LAN Emulation Over ATM Version 2 - LUNI Specification,
                 Annex A (LAN Emulation Control Frame TLVs)"
         ::= { lecMacQueryObjects 6 }




     --
     --  Route Descriptor queries
     --

     lecRDQueryObjects  OBJECT IDENTIFIER ::= { lecQueryObjects 2 }

     lecRDQueryLecIndex OBJECT-TYPE
         SYNTAX       INTEGER ( 0..2147483647 )
         MAX-ACCESS   read-write
         STATUS       current
         DESCRIPTION
                "The 'lecIndex' of the LEC whose route descriptor to
                 ATM address binding the network management station
                 wants to look up.

                 When written:  Triggers a new query.

                 When read:  Identifies the last LEC queried.  (This
                 value may be 0 if there have been no queries, or if
                 a LEC has been deleted.)"
     ::= { lecRDQueryObjects 1 }

lecRDQuerySegmentID OBJECT-TYPE
    SYNTAX       INTEGER( 0..4095 )
    MAX-ACCESS   read-write
    STATUS       current
    DESCRIPTION
            "The LAN ID portion of the IEEE 802.5 route descriptor
             whose binding the network management station wants to
             look up.

             When written:  Triggers a new query.

             When read:  Returns the LAN ID portion of the last RD
             queried, or 0 if lecRDQueryLecIndex is 0."
    ::= { lecRDQueryObjects 2 }

lecRDQueryBridgeNumber OBJECT-TYPE
    SYNTAX       INTEGER( 0..15 )
    MAX-ACCESS   read-write
    STATUS       current
    DESCRIPTION
            "The bridge number portion of the IEEE 802.5 route
             descriptor whose binding the network management
             station wants to look up.

             When written:  Triggers a new query.

             When read:  Returns the bridge number portion of the
             last RD queried, or 0 if lecRDQueryLecIndex is 0."
    ::= { lecRDQueryObjects 3 }

lecRDQueryStatus OBJECT-TYPE
    SYNTAX       INTEGER {
                          none(1),
                          unsupported(2),
                          local(3),
                          remote(4),
                          unknown(5)

                         }
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
            "The status of the last query - one of the following:

                none(1)
                    No queries have been made; therefore, other
                    results values are invalid.

                unsupported(2)
                    Queries are not supported for this LEC.

                local(3)
                    The route descriptor is one of the client's
                    local route descriptors.

                remote(4)
                    The route descriptor is one of the client's
                    remote route descriptors.

                unknown(5)
                    The route descriptor does not belong (is not
                    known to belong) to the client."
    ::= { lecRDQueryObjects 4 }

lecRDQueryAtmAddress OBJECT-TYPE
    SYNTAX      AtmLaneAddress
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "The non-multiplexed ATM address which this client --
             i.e., client 'lecRDQueryLecIndex' -- would use in a
             LE_ARP_RESPONSE for the route descriptor
             ('lecRDQuerySegmentID', 'lecRDQueryBridgeNumber').

             This object is only meaningful when lecRDQueryStatus
             is 'local' or 'remote'."
    ::= { lecRDQueryObjects 5 }

lecRDQueryMuxedAtmAddress OBJECT-TYPE
    SYNTAX      AtmLaneAddress
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "The LLC-multiplexed ATM address which this client --
             i.e., client 'lecRDQueryLecIndex' -- would use in a
             LE_ARP_RESPONSE for the route descriptor
             ('lecRDQuerySegmentID', 'lecRDQueryBridgeNumber').

             This object is only meaningful when lecRDQueryStatus
             is 'local' or 'remote'."
    ::= { lecRDQueryObjects 6 }

lecRDQueryServiceCategory OBJECT-TYPE
    SYNTAX      INTEGER ( 0..2147483647 )
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
            "The Service Category TLV that this client would use
             in a LE_ARP_RESPONSE for the route descriptor
             ('lecRDQuerySegmentID', 'lecRDQueryBridgeNumber').

             If this client wouldn't return a Service Category TLV
             in the LE_ARP_RESPONSE, this object should be set to
             zero.

             The bitmap is larger than the LANE 2.0 specification
             requires, in case room is ever needed for expansion.

             This object is only meaningful when lecRDQueryStatus
             is 'local' or 'remote'."
    REFERENCE
            "LAN Emulation Over ATM Version 2 - LUNI Specification,
             Annex A (LAN Emulation Control Frame TLVs)"
    ::= { lecRDQueryObjects 7 }



-- ------------------------------------------------------------ --

--
--  LAN Emulation Client TLVs
--

lecTlvTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF LecTlvEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
            "This table provides a place for an agent to store
             interesting TLVs associated with

                 o  Registered LAN destinations

                 o  LE_ARP entries

                 o  Query results from the LE_ARP response group

                 o  Initialization / configuration

             and any other categories the ATM Forum adds.

             Unless otherwise stated,

                 o  Entries in this table are completely optional.

                 o  An implementation may choose to keep only the
                    TLVs which are 'interesting' to its designers,
                    such as MPOA TLVs in LE_ARP_RESPONSEs.

                 o  TLVs which already have their own MIB objects
                    should not be duplicated here.

                 o  Where write / create access is allowed, it is
                    not mandatory.

             Implementors should document the specific lecTlvTable
             features of their agent."
    REFERENCE
            "LAN Emulation Over ATM Version 2 - LUNI Specification,
             Annex A (LAN Emulation Control Frame TLVs)"
    ::= { leClientMIBObjects 13 }

lecTlvEntry OBJECT-TYPE
    SYNTAX      LecTlvEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
            "Each table row describes one TLV of a related set."
    INDEX   { lecIndex, lecTlvSetIndex, lecTlvLanDestination,
                        lecTlvIndex }
    ::= { lecTlvTable 1 }

LecTlvEntry ::=
    SEQUENCE {

        lecTlvSetIndex                  INTEGER,
        lecTlvLanDestination            OCTET STRING,
        lecTlvIndex                     INTEGER,
        lecTlvType                      OCTET STRING,
        lecTlvValue                     OCTET STRING,
        lecTlvRowStatus                 RowStatus

    }

lecTlvSetIndex OBJECT-TYPE
    SYNTAX      INTEGER {
                         layer3Tlvs(1),
                         actualRegTlvs(2),
                         configRegTlvs(3),
                         leArpTlvs(4),
                         queryTlvs(5)
                        }
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
            "An index which helps to identify a set of TLVs, and to
             indicate their purpose.

             The value of this index defines the interpretation and
             use of corresponding table rows as follows:

                 layer3Tlvs(1)
                 -------------
                     This index provides access to Layer-3-Address
                     TLVs (Initial State variable C36) used during
                     the Initialization phase.

                     TLVs are stored in rows

                         ( lecIndex, 1,
                           LAN_Destination( all '00's ),
                           * )

                     These rows may be read / created / written /
                     deleted.


                 actualRegTlvs(2)
                 ----------------
                     This index provides access to TLVs which were
                     used to register LAN Destinations.

                     These TLVs are associated with entries in the
                     lecMacAddressTable and lecRouteDescrTable and
                     stored in rows

                         ( lecIndex, 2,
                           LAN_Destination( lecMacAddress ),
                           * )

                         ( lecIndex, 2,
                           LAN_Destination(
                               lecRouteDescrSegmentID,
                               lecRouteDescrBridgeNumber ),
                           * )

                     These rows may be read or deleted, but cannot
                     be created or modified.  Most implementations
                     are expected to make them read-only.

                     Note that TLVs that have their own objects in
                     the associated tables SHOULD NOT appear here.
                     LLC-Muxed-ATM-Address & Service-Category TLVs
                     are two examples.


                 configRegTlvs(3)
                 ----------------
                     This index provides a way to set registration
                     TLVs for unregistered LAN Destinations.

                     Hopefully this is not something that managers
                     will need to do a lot.
```

```
    TLVs are stored in rows

        ( lecIndex, 3,
          LAN_Destination( the_LAN_destination ),
          * )

    Note that the agent might restrict the set of
    LAN Destinations and TLVs that it allows; for
    instance, to screen out the 'V2 Capable' TLV.


leArpTlvs(4)
------------
    This index provides access to TLVs associated
    with the LE_ARP cache.

    These TLVs are associated with leArpTable and
    leRDArpTable, and stored in rows

        ( lecIndex, 4,
          LAN_Destination( leArpMacAddress ),
          * )

        ( lecIndex, 4,
          LAN_Destination(
                  leRDArpSegmentID,
                  leRDArpBridgeNumber ),
          * )

    Agents may allow write/create access to TLVs
    when the type of the associated LE_ARP entry
    is 'staticVolatile', 'staticNonVolatile', or
    'other'.

    Note that TLVs that have their own objects in
    the associated tables SHOULD NOT appear here.
    LLC-Muxed-ATM-Address & Service-Category TLVs
    are two examples.


queryTlvs(5)
------------
    This index provides access to TLVs associated
    with the last MAC Address or Route Descriptor
    query.

    TLVs are associated with lecMacQueryObjects &
    lecRDQueryObjects, and stored in rows

        ( lecIndex, 5,
          LAN_Destination( lecMacQueryAddress ),
          * )

        ( lecIndex, 5,
          LAN_Destination(
                  lecRDQuerySegmentID,
                  lecRDQueryBridgeNumber ),
          * )

    Note that TLVs that have their own objects in
    the associated tables SHOULD NOT appear here.
    LLC-Muxed-ATM-Address & Service-Category TLVs
    are two examples.
```

```
                  Additional indices may be defined by the ATM Forum as
                  needed."
         ::= { lecTlvEntry 1 }

     lecTlvLanDestination OBJECT-TYPE
         SYNTAX      OCTET STRING ( SIZE( 8 ) )
         MAX-ACCESS  not-accessible
         STATUS      current
         DESCRIPTION
                 "The MAC address or route descriptor associated with
                  the TLV (see the description of the lecTlvSetIndex),
                  encoded in the fashion defined in the LANE 2.0 LUNI
                  specification (Table 16).  To recap:

                  The first two octets indicate the LAN Destination's
                  type:  00, 01 for a MAC address; 00, 02 for a route
                  descriptor; 00, 00 for 'no LAN Destination'.

                  The next six octets contain either the MAC address,
                  or four 00 octets followed by the route descriptor,
                  or six nulls.

                  Note:
                  -----
                  In the future, the ATM Forum may extend this field
                  for use with other types of index values, provided
                  that each new type of index value has a unique tag.
                  Such extensions are NOT to be considered a 'change
                  in semantics' that requires the deprecation of the
                  index object and the table.

                  Network management system implementors are advised
                  to display values whose tags they do not recognize
                  as (formatted) hexadecimal strings."
         ::= { lecTlvEntry 2 }

     lecTlvIndex OBJECT-TYPE
         SYNTAX      INTEGER ( 1..65535 )
         MAX-ACCESS  not-accessible
         STATUS      current
         DESCRIPTION
                 "An arbitrary index which allows a set to contain
                  multiple TLVs."
         ::= { lecTlvEntry 3 }

     lecTlvType OBJECT-TYPE
         SYNTAX      OCTET STRING ( SIZE( 4 ) )
         MAX-ACCESS  read-create
         STATUS      current
         DESCRIPTION
                 "A four-octet string indicating this item's type.
                  It consists of a three-octet OUI, followed by an
                  octet identifying one of the organization's TLVs."
         ::= { lecTlvEntry 4 }

     lecTlvValue OBJECT-TYPE
         SYNTAX      OCTET STRING ( SIZE( 0..255 ) )
         MAX-ACCESS  read-create
         STATUS      current
         DESCRIPTION
                 "The value - and, implicitly, the length - of the
                  TLV."
         ::= { lecTlvEntry 5 }
```

```
    lecTlvRowStatus OBJECT-TYPE
        SYNTAX      RowStatus
        MAX-ACCESS  read-create
        STATUS      current
        DESCRIPTION
                "Provides a way for network managers to create and
                 delete rows in the TLV table."
        ::= { lecTlvEntry 6 }



-- Conformance Information

leClientMIBConformance  OBJECT IDENTIFIER ::= { leClientMIB 2 }

leClientMIBGroups       OBJECT IDENTIFIER ::=
                                          { leClientMIBConformance 1 }
leClientMIBCompliances  OBJECT IDENTIFIER ::=
                                          { leClientMIBConformance 2 }

-- Compliance Statements

leClientMIBCompliance           MODULE-COMPLIANCE
        STATUS           deprecated
        DESCRIPTION     "The original compliance statement for SNMP
                         entities which supported ATM LAN Emulation
                         Clients.

                         See leClientMIBCompliance2 for the updated
                         statement.

        For a host to conform to this MIB, it must also implement

          - RFC 1213 - MIB II.

          - Interfaces table entries for each LE Client, as per the
            LAN Emulation Client management specification.

          - The AToM MIB (RFC 1695 - Definitions of Managed Objects
            for ATM Management), according to the conformance
            statements defined in that RFC.

        Optionally, a host may implement

          - RFC 1573 - Evolution of the Interfaces Group of MIB-II.

          - The 'ifRcvAddressGroup' from RFC 1573, which provides a
            way to represent and configure interfaces with multiple
            addresses.

        See the LAN Emulation Client management specification for
        interpretations of RFC 1573 / MIB-II as they apply to LAN
        Emulation Clients."


        MODULE -- this module
            MANDATORY-GROUPS   {leClientConfigGroup,
                                leClientStatusGroup,
                                leClientMappingGroup,
                                leClientStatisticsGroup,
                                leClientServerVccGroup,
                                leClientAtmAddressesGroup,
                                leClientMacAddressesGroup,
                                leClientArpGroup}

            GROUP           leClientRouteDescriptorsGroup
            DESCRIPTION     "This group is mandatory only for hosts
                                 that support emulated 802.5 LANs."

            GROUP           leClientRDArpGroup
            DESCRIPTION     "This group is mandatory only for hosts
                                 that support emulated 802.5 LANs."


            --
            --  Minimum access definitions for objects
            --

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

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

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

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

                                 Implementations are not required to
                                 support all legal values."

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

                                 Implementations are not required to
                                 support all legal values."

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

            ::= { leClientMIBCompliances 1 }



    leClientMIBCompliance2          MODULE-COMPLIANCE
            STATUS           current
            DESCRIPTION     "The compliance statement for SNMP entities
                             that support LECs and Version 2 of the LEC
                             management specification.

            For a host to conform to this MIB, it must also implement

              - RFC 1213 - MIB II.

              - Interfaces table entries for each LE Client, as per the
                LAN Emulation Client management specification.

              - The AToM MIB (RFC 1695 - Definitions of Managed Objects
                for ATM Management), according to the conformance
                statements defined in that RFC.

            Optionally, a host may implement

              - RFC 1573 - Evolution of the Interfaces Group of MIB-II.

              - The 'ifRcvAddressGroup' from RFC 1573, which provides a
                way to represent and configure interfaces with multiple
                addresses.

            See the LAN Emulation Client management specification for
            interpretations of RFC 1573 / MIB-II as they apply to LAN
            Emulation Clients."


            MODULE -- this module

                MANDATORY-GROUPS   {leClientStatusGroup,
                                    leClientMappingGroup,
                                    leClientStatisticsGroup,
                                    leClientServerVccGroup,
                                    leClientAtmAddressesGroup,
                                    leClientMacAddressesGroup,
                                    leClientArpGroup,
                                    leClientConfigGroupV1,
                                    leClientConfigGroupV2,
                                    leClientStatusGroupV2,
                                    leClientStatisticsGroupV2,
                                    leClientAtmAddressesGroupV2,
                                    leClientMacAddressesGroupV2,
                                    leClientArpGroupV2,
                                    leClientMcForwardGroup}

                GROUP               leClientRouteDescriptorsGroup
                DESCRIPTION         "This group is mandatory only for hosts
                                     that support emulated 802.5 LANs."
```

```
GROUP               leClientRouteDescriptorsGroupV2
DESCRIPTION         "This group is mandatory only for hosts
                     that support emulated 802.5 LANs."

GROUP               leClientRDArpGroup
DESCRIPTION         "This group is mandatory only for hosts
                     that support emulated 802.5 LANs."

GROUP               leClientRDArpGroupV2
DESCRIPTION         "This group is mandatory only for hosts
                     that support emulated 802.5 LANs."

GROUP               leClientMacQueryGroup
DESCRIPTION         "This group is optional."

GROUP               leClientRDQueryGroup
DESCRIPTION         "This group is optional."

GROUP               leClientTlvGroup
DESCRIPTION         "This group is optional."


--
--  Minimum access definitions for objects
--

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

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

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

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

                     Implementations are not required to
                     support all legal values."

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

                     Implementations are not required to
                     support all legal values."

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

OBJECT              lecV2MaxUnknownFrameCount
MIN-ACCESS          read-only
DESCRIPTION         "Write access is not required."
```

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

OBJECT              lecConfigV2Capable
MIN-ACCESS          read-only
DESCRIPTION         "Write access is not required."

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

::= { leClientMIBCompliances 2 }


-- Units of Conformance


leClientConfigGroup    OBJECT-GROUP
        OBJECTS     { lecRowStatus, lecOwner,
                      lecConfigMode,
                      lecConfigLanType,
                      lecConfigMaxDataFrameSize,
                      lecConfigLanName,
                      lecConfigLesAtmAddress,
                      lecControlTimeout,
                      lecMaxUnknownFrameCount,
                      lecMaxUnknownFrameTime,
                      lecVccTimeoutPeriod,
                      lecMaxRetryCount,
                      lecAgingTime,
                      lecForwardDelayTime,
                      lecExpectedArpResponseTime,
                      lecFlushTimeOut,
                      lecPathSwitchingDelay,
                      lecLocalSegmentID,
                      lecMulticastSendType,
                      lecMulticastSendAvgRate,
                      lecMulticastSendPeakRate,
                      lecConnectionCompleteTimer }
        STATUS      deprecated
        DESCRIPTION
            "A collection of objects used for creating and
```

```
                     configuring LAN Emulation Clients.

                     This group is deprecated because some objects
                     in it have been replaced by newer ones."
                ::= { leClientMIBGroups 1 }

        leClientStatusGroup    OBJECT-GROUP
                OBJECTS     { lecIfIndex,
                              lecPrimaryAtmAddress,
                              lecID,
                              lecInterfaceState,
                              lecLastFailureRespCode,
                              lecLastFailureState,
                              lecProtocol, lecVersion,
                              lecTopologyChange,
                              lecConfigServerAtmAddress,
                              lecConfigSource,
                              lecActualLanType,
                              lecActualMaxDataFrameSize,
                              lecActualLanName,
                              lecActualLesAtmAddress,
                              lecProxyClient }
                STATUS      current
                DESCRIPTION
                    "A collection of objects describing the status
                     and operational parameters of the managed LAN
                     Emulation Clients."
                ::= { leClientMIBGroups 2 }

        leClientMappingGroup    OBJECT-GROUP
                OBJECTS     { lecMappingIndex }
                STATUS      current
                DESCRIPTION
                    "A collection of objects which map a LAN Emulation
                     Client's 'ifIndex' to its 'lecIndex'."
                ::= { leClientMIBGroups 3 }

        leClientStatisticsGroup    OBJECT-GROUP
                OBJECTS     { lecArpRequestsOut, lecArpRequestsIn,
                              lecArpRepliesOut , lecArpRepliesIn,
                              lecControlFramesOut,
                              lecControlFramesIn,
                              lecSvcFailures }
                STATUS      current
                DESCRIPTION
                    "A collection of objects which act as an extension
                     to the lecConfigTable.

                     These objects provide statistics on LAN Emulation
                     control traffic and SVC establishment."
                ::= { leClientMIBGroups 4 }

        leClientServerVccGroup    OBJECT-GROUP
                OBJECTS     { lecConfigDirectInterface,
                              lecConfigDirectVpi,
                              lecConfigDirectVci,
                              lecControlDirectInterface,
                              lecControlDirectVpi,
                              lecControlDirectVci,
                              lecControlDistributeInterface,
                              lecControlDistributeVpi,
                              lecControlDistributeVci,
                              lecMulticastSendInterface,
                              lecMulticastSendVpi,
                              lecMulticastSendVci,
                              lecMulticastForwardInterface,
                              lecMulticastForwardVpi,
                              lecMulticastForwardVci }
                STATUS      current
                DESCRIPTION
                    "A collection of objects which act as an extension
                     to the lecConfigTable.

                     These objects identify Configuration Direct,
                     Control, and Multicast VCCs for each client,
                     making it easier to locate their entries in the
                     AToM MIB."
                ::= { leClientMIBGroups 5 }

        leClientAtmAddressesGroup   OBJECT-GROUP
                OBJECTS     { lecAtmAddressStatus }
                STATUS      current
                DESCRIPTION
                    "A collection of objects which describe all of the
                     ATM addresses belonging to each client."
                ::= { leClientMIBGroups 6 }

        leClientMacAddressesGroup   OBJECT-GROUP
                OBJECTS     { lecMacAddressAtmBinding }
                STATUS      current
                DESCRIPTION
                    "A collection of objects which describe all of the
                     MAC addresses registered for each client."
                ::= { leClientMIBGroups 7 }

        leClientRouteDescriptorsGroup    OBJECT-GROUP
                OBJECTS     { lecRouteDescrAtmBinding }
                STATUS      current
                DESCRIPTION
                    "A collection of objects which describe all of the
                     Route Descriptors registered for each client."
                ::= { leClientMIBGroups 8 }

        leClientArpGroup    OBJECT-GROUP
                OBJECTS     { leArpAtmAddress,
                              leArpIsRemoteAddress,
                              leArpEntryType,
                              leArpRowStatus }
                STATUS      current
                DESCRIPTION
                    "A collection of objects which describe the MAC-
                     to-ATM address mappings that this LAN Emulation
                     Client has learned."
                ::= { leClientMIBGroups 9 }

        leClientRDArpGroup    OBJECT-GROUP
                OBJECTS     { leRDArpAtmAddress, leRDArpEntryType,
                              leRDArpRowStatus }
                STATUS      current
                DESCRIPTION
                    "A collection of objects which describe the IEEE
                     802.5 Route Descriptor-to-ATM address mappings
                     that this LAN Emulation Client has learned."
                ::= { leClientMIBGroups 10 }

        leClientConfigGroupV1    OBJECT-GROUP
                OBJECTS     { lecRowStatus, lecOwner,
                              lecConfigMode,
                              lecConfigLanType,
                              lecConfigMaxDataFrameSize,
                              lecConfigLanName,
                              lecConfigLesAtmAddress,
                              lecControlTimeout,
                              lecMaxUnknownFrameTime,
                              lecVccTimeoutPeriod,
                              lecMaxRetryCount,
                              lecAgingTime,
                              lecForwardDelayTime,
                              lecExpectedArpResponseTime,
                              lecFlushTimeOut,
                              lecPathSwitchingDelay,
                              lecMulticastSendType,
                              lecMulticastSendAvgRate,
                              lecMulticastSendPeakRate,
                              lecConnectionCompleteTimer,
                              lecConfigLecsAtmAddress,
                              lecV2MaxUnknownFrameCount,
                              lecConfigLocalSegmentID }
                STATUS      current
                DESCRIPTION
                    "A collection of objects used for creating and
                     configuring both V1 and V2 LE Clients.

                     It is the same as leClientConfigGroup, except
                     that the deprecated objects lecLocalSegmentID
                     and lecMaxUnknownFrameCount have been removed
                     and replaced by lecV2MaxUnknownFrameCount and
                     lecConfigLocalSegmentID.

                     Also, lecConfigLecsAtmAddress has been added."
                ::= { leClientMIBGroups 11 }

        leClientConfigGroupV2    OBJECT-GROUP
                OBJECTS     { lecInitialControlTimeout,
                              lecControlTimeoutMultiplier,
                              lecConfigV2Capable,
                              lecConfigSelectiveMulticast,
                              lecForwardDisconnectTimeout,
                              lecConfigLLCMultiplexCapable,
                              lecMinReconfigureDelay,
                              lecMaxReconfigureDelay,
                              lecMaxBusConnectRetries,
                              lecTokenRingExplorerExclude }
                STATUS      current
                DESCRIPTION
                    "A collection of objects which extend the LEC
                     configuration table to deal with LANE V2.0."
                ::= { leClientMIBGroups 12 }

        leClientStatusGroupV2    OBJECT-GROUP
                OBJECTS     { lecActualLocalSegmentID,
                              lecActualV2Capable,
                              lecElanID,
                              lecActualSelectiveMulticast,
                              lecActualLLCMultiplexCapable,
                              lecPreferredLesAddress,
                              lecStoresServiceCategories }
                STATUS      current
                DESCRIPTION
                    "A collection of objects which extend the LEC
                     status table to deal with LANE V2.0."
                ::= { leClientMIBGroups 13 }
```

```
leClientStatisticsGroupV2    OBJECT-GROUP
        OBJECTS     { lecFlowFailures,
                      lecEchoDiscards,
                      lecFilteredMulticasts }
        STATUS      current
        DESCRIPTION
            "A collection of objects which provide statistics
             on LLC-multiplexed flow establishment, and count
             harmless discards separately from ifInDiscards."
        ::= { leClientMIBGroups 14 }

leClientAtmAddressesGroupV2   OBJECT-GROUP
        OBJECTS     { lecAtmAddressMuxed }
        STATUS      current
        DESCRIPTION
            "An object which extends the lecAtmAddressTable to
             provide information about LANE V2.0 multiplexing."
        ::= { leClientMIBGroups 15 }

leClientMacAddressesGroupV2   OBJECT-GROUP
        OBJECTS     { lecMacAddressMuxedAtmBinding,
                      lecMacAddressServiceCategory }
        STATUS      current
        DESCRIPTION
            "Objects which extend the lecMacAddressTable to
             provide information about LANE V2.0 multiplexing
             and TLVs."
        ::= { leClientMIBGroups 16 }

leClientRouteDescriptorsGroupV2    OBJECT-GROUP
        OBJECTS     { lecRouteDescrMuxedAtmBinding,
                      lecRouteDescrServiceCategory }
        STATUS      current
        DESCRIPTION
            "An object which extends the lecRouteDescrTable to
             provide information about LANE V2.0 multiplexing."
        ::= { leClientMIBGroups 17 }

leClientArpGroupV2    OBJECT-GROUP
        OBJECTS     { leArpMuxedAtmAddress,
                      leArpServiceCategory }
        STATUS      current
        DESCRIPTION
            "A collection of objects which provide additional
             LE-ARP information for LANE Version 2.0."
        ::= { leClientMIBGroups 18 }

leClientRDArpGroupV2    OBJECT-GROUP
        OBJECTS     { leRDArpMuxedAtmAddress,
                      leRDArpServiceCategory,
                      leRDArpIsRemoteDescriptor }
        STATUS      current
        DESCRIPTION
            "A collection of objects which provide additional
             LE-ARP information for LANE Version 2.0."
        ::= { leClientMIBGroups 19 }

leClientMcForwardGroup    OBJECT-GROUP
        OBJECTS     { lecMcForwardVerification }
        STATUS      current
        DESCRIPTION
            "A collection of objects that provide information
             on Multicast Forward VCCs belonging to LANE V2.0
```

```
                 clients."
             ::= { leClientMIBGroups 20 }

     leClientMacQueryGroup    OBJECT-GROUP
             OBJECTS     { lecMacQueryLecIndex,
                           lecMacQueryAddress,
                           lecMacQueryStatus,
                           lecMacQueryAtmAddress,
                           lecMacQueryMuxedAtmAddress,
                           lecMacQueryServiceCategory }
             STATUS      current
             DESCRIPTION
                 "A collection of objects that allow a network
                  manager to find out ATM bindings for a proxy
                  client's remote MAC address(es)."
             ::= { leClientMIBGroups 21 }

     leClientRDQueryGroup    OBJECT-GROUP
             OBJECTS     { lecRDQueryLecIndex,
                           lecRDQuerySegmentID,
                           lecRDQueryBridgeNumber,
                           lecRDQueryStatus,
                           lecRDQueryAtmAddress,
                           lecRDQueryMuxedAtmAddress,
                           lecRDQueryServiceCategory }
             STATUS      current
             DESCRIPTION
                 "A collection of objects that allow a network
                  manager to find out ATM bindings for a proxy
                  client's remote route descriptor(s)."
             ::= { leClientMIBGroups 22 }

     leClientTlvGroup    OBJECT-GROUP
             OBJECTS     { lecTlvType,
                           lecTlvValue,
                           lecTlvRowStatus }
             STATUS      current
             DESCRIPTION
                 "A collection of objects that allow a network
                  manager to see 'interesting' TLV information."
             ::= { leClientMIBGroups 23 }


END
```