The ATM Forum
Technical Committee

LAN Emulation Servers Management Specification 1.0

af-lane-0057.000

March 1996

LAN Emulation Servers Management Specification 1.0

af-lane-0057.000

March 1996

The ATM Forum
Worldwide Headquarters
2570 West El Camino Real
Suite 304
Mountain View, CA 94040
Tel: +1-415-949-6700
Fax: +1-415-949-6705

Acknowledgement

The contributions adopted by the ATM Forum and the assistance by the editing group of the LAN Emulation Sub-working Group, who contributed text and reviewed the contributions were essential to this specification.

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

Chris Young

Ian Griffiths

Prakash Desai

The assistance by these members and all who participated in the LAN Emulation Sub-working Group, especially chairman Keith McCloghrie is appreciated.

Faye Ly, Editor

1. Introduction 3
1.1 Abbreviations 3
1.2 References 4
2. Management Functions 5
2.1 Configuration Management 5
2.2 Performance Management 5
2.3 Fault Management 5
2.4 Security Management 5
2.5 Accounting Management 5
3. Management Framework 6
3.1 The SNMP Network Management Framework 6
3.2 Requirements for LAN Emulation Servers 6
3.2.1 Requirements for SNMP 6
3.3 MIB Conventions 6
4. MIB-II and other MIB support 8
4.1 MIB II support 8
4.2 AToMMIB Support 8
5. Emulated LAN MIB Description 9
5.1 Management model 9
5.2 MIB Groups 9
5.2.1 ELAN Administration Group 10
5.2.2 ELAN Configuration Group 10
5.2.3 LECS Configuration Group 12
5.2.4 LECS Statistics Group 13
This table lists all counters associated with the LECS this agent maintains. 13
5.2.5 LECS Fault Management Group 13
6. LAN Emulation Server MIB Description 15
6.1 Management model 15
6.2 MIB Organisation 15
6.3 MIB Groups 15
6.3.1 LES Configuration Group 15
6.3.2 LES Statistics Group 16
6.3.3 LES-LEC Statistics Group 16
6.3.4 LES Fault Management Group 16
7. Broadcast and Unknown Server MIB description 18
7.1 Management model 18
7.2 MIB Organisation 18
7.3 MIB Groups 18
7.3.1 BUS Configuration Group 18
7.3.2 BUS Statistics Group 18
7.4.2 BUS Fault Management Group 19
8. LAN Emulation Servers MIB Definitions 20

1. Introduction

ATM LAN Emulation emulates services of existing LANs across an ATM network. It provides a MAC layer service, specifically Ethernet/IEEE 802.3 and IEEE 802.5, over an ATM network. LAN emulation is described in the ATM Forum document "LAN Emulation Over ATM Specification - Version 1.0" [1]. This document describes how to manage some of the components in the LAN Emulation.

The LAN Emulation specification 1.0 defines four components: LEC (LAN Emulation Client), LES (LAN Emulation Server), BUS (Broadcast and Unknown Server) and LECS (LAN Emulation Configuration Server). LES, BUS and LECS provide various services to enable communication between LECs, and are referred to in this document collectively as 'LAN Emulation servers'.

This document defines the management information for LES, BUS and LECS. The management information of the LEC is defined in [6]. This specification include three MIB modules. The ELAN MIB provides ELAN configuration and LECS management information. The LECS management group is mandatory if and only if the LECS is implemented. The LAN Emulation LES MIB covers LES management information, and BUS MIB provides BUS management information.

1.1 Abbreviations

The following acronyms 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
LECS	LAN Emulation Configuration Server
LES	LAN Emulation Server
LNNI	LAN Emulation Network-Network Interface
LUNI	LAN Emulation User-Network Interface
MAC	Media 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

[1] The ATM Forum, LAN Emulation Over ATM Specification, Version 1.0.

[2] RFC 1213, McCloghrie and Rose, Management Information Base for Network Management of TCP/IP-based internets: MIB-II.

[3] RFC 1573, McCloghrie and KastenHoltz, Evolution of the Interfaces Group of MIB-II.

[4] RRC 1695, Ahmed and Tesink, Definitions of Managed Objects for ATM Management.

[5] Internet Drafts, Ly, Noto, Smith, Tesink, Definitions of Supplemental Managed Objects for ATM Management. (draft-ietf-atommib-atm2-01.txt)

[6] The ATM Forum. LAN Emulation Client Management: Version 1.0 Specification.

[7] RFC 1442, Case, McCloghrie, Rose and Waldbusser, Structure of Management Information for version 2 of the Simple Network Management Protocol (SNMPv2).

[8] RFC 1445, Galvin and McCloghrie, Administrative Model for version 2 of the Simple Network Management Protocol (SNMPv2).

[9] RFC 1448, Case, McCloghrie, Rose and Waldbusser, Protocol Operations for version 2 of the Simpel Network Management Protocol (SNMPv2).

[10] RFC 1443, Case, McCloghrie, Rose and Waldbusser, Textual Conventions for version 2 of the Simple Network Management Protocol (SNMPv2).

2. Management Functions

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

2.1 Configuration Management

The MIBs enable network managers to:

- create and destroy Emulated LANs (ELANs)
- assign clients to and delete them from an ELAN
- control the parameters of the LECS, LES and BUS
- monitor ELAN topology, e.g. which LECs are joined to a LES
- identify the VCCs being used by the LES or BUS

No information is provided about VCCs beyond describing which are being used. Management of VCC configuration is specified in [4] and [5]. Management issues relating to the LANE server-to-server (LNNI) protocol are outside the scope of this specification because at time of writing the Forum work on this topic is not complete. It is however a design goal to make the MIB modules easy to extend to accommodate the future LNNI support, and it is anticipated that this will be done in a future revision of these MIBs once the LNNI protocols have been published by the ATM Forum.

2.2 Performance Management

Network managers can obtain statistics for ELAN server components. These are collected for every server and also for each server-client pairing. While LES- and BUS-related statistics are located in the LES MIB and BUS MIB respectively, LECS statistics (along with all other LECS management) are to be found in the optional LECS part of the ELAN MIB.

2.3 Fault Management

The MIBs contain Facilities for detection of problems in an ELAN that are caused by the failures detected by service components. Information is provided as follows:

- operational status is available for each service component (indicates server is ‘up’ or ‘down’)
- error logs are maintained for each service component (see section 5 for details)

2.4 Security Management

Security management is outside the scope of this version of the LAN Emulation Server Management specification.

2.5 Accounting Management

Accounting management is outside the scope of this version of the LAN Emulation Server Management specification.

3. Management Framework

The network management framework for LAN Emulation Servers is defined using SNMP MIBs. SNMP is an Internet-standard network management framework of which there are currently more than one versions. The MIB modules specified in this document are both compliant to the SNMPv2, and semantically identical to the peer SNMPv1 definitions. This specification does not mandate the use of SNMPv2. 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@dbc.mtview.ca.us`.

3.1 The SNMP Network Management Framework

The SNMP Network Management Framework are defined in the RFCs Since the SNMP is still evolving, this document only lists a few RFCs that are related.. They are:

- RFC 1442 [7] which defines the SMI, the mechanisms used for describing and naming objects for the purpose of management
- STD 17, RFC 1213 [2] defines MIB-II, the core set of managed objects for the Internet suite of protocols
- RFC 1445 [8] which defines the administrative and other architectural aspects of the framework
- RFC 1448 [9] which defines the protocol used for network access to managed objects
- RFC 1443 [10] which defines the textual conventions used by network management information definitions

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

3.2 Requirements for LAN Emulation Servers

The LAN Emulation Server and Broadcast and Unknown Server should be network manageable either via the SNMP network management protocol or via some other network management protocol. The management of the LAN Emulation Configuration Server is required only for the system that implements it.

3.2.1 Requirements for SNMP

SNMP agents that support LAN Emulation Configuration Servers, LAN Emulation Servers and Broadcast and Unknown Servers MUST implement:

- Three MIB modules defined in this specification: ELAN MIB, LES MIB and BUS MIB

3.3 MIB Conventions

The following convention is used in the MIB definitions in this specification: When a table is defined to allow row creation and deletion or to be read-creatable, the RowStatus object is required in the ‘create’ or ‘destroy’ operation. This operation is performed on the row indicated by the instance given in the SNMP request.

RowStatus is a textual convention defined in RFC 1443 [10]. It is defined with 6 enumeration:

- active
- notInService
- notReady
- createAndGo
- createAndWait
- destroy

This specification uses RowStatus in every read-creatable table.

4. MIB-II and other MIB support

All SNMP agents supporting LAN Emulation Servers may implement MIB II, and are also encouraged to implement RFC 1695. This chapter explains how ELAN MIB, LES MIB and BUS MIB are related to each of these MIB modules.

4.1 MIB II support

LAN Emulation Servers present no interfaces as defined in MIB II. They are however related to one or more ATM interfaces in many ways though:

- LECS can be listening to one or more ATM interfaces with a well-known ATM address as defined in the LANE Specification.
- LES can be receiving/sending control or ATM Le-ARP traffic via one or more ATM interfaces.
- BUS can be receiving/forwarding traffic on one or more ATM interfaces.

The ATM interface is specified throughout the MIB modules where Virtual Circuits in use by servers are defined. It is also specified in every server configuration table where the server's ATM addresses are defined. The latter is provided to enable network managers to determine which ATM switch the servers derived their ATM addresses from.

4.2 AToMMIB Support

LAN Emulation Servers use Virtual Circuits (Switched or Permanent) to communicate with the Clients. Full information on the VCCs is found in the AToMMIB. LAN Emulation Servers MIB provide the indexing for these VCCs so network managers can obtain the attributes of a server VCC from the AToMMIB.

PVCs (Permanent Virtual Circuits) are created by the network manager in the AToMMIB. LAN Server MIBs provide writable objects so the network manager can use the PVCs created in the AToMMIB. (i.e. the lesVccTable and busVccTable)..

5. Emulated LAN MIB Description

5.1 Management model

This specification defines three MIB modules: ELAN MIB, LES MIB and BUS MIB. The ELAN MIB provides network manager with the means to change the configuration of ELANs. The LES and BUS MIBs allow reading back of the current status of the ELANs. This chapter describes ELAN MIB.

It is important to understand that the ELAN MIB only manages a repository of static information. To create an ELAN which a client can join will actually require three stages using these MIBs:

- create a new ELAN in the ELAN MIB
- create a LES for that ELAN using the LES MIB
- create a BUS for that ELAN using the BUS MIB

These operations must be performed in these separate MIBs because there is no guarantee that any of the components in question will be collocated. Indeed while a LES and BUS may be likely to be managed by a single agent, it is highly likely that the user of information in the ELAN MIB (e.g. an LECS) will not be collocated with the LES or BUS. The LANE specification currently provides no standard method for centralised creation of LE Servers, hence the three operations required to create an ELAN.

The ELAN MIB simply deals with information that will be required to enable a client to join an ELAN. This information can be divided into two types: information required to decide which ELAN a client should join and information which a client will need to join the ELAN.

The assignment of clients to ELANs is controlled using ‘policies’. A policy is a rule based on information available on a client (such as ATM address, ELAN type etc.) linking that client to an ELAN. These are described in detail later in this chapter.

Typically the information in this MIB would be used by an LECS, the former in deciding which ELAN, if any, a client should be assigned to on the basis of its CONFIG_REQUEST, and the latter in deciding what to put in a CONFIG_RESPONSE. It is possible for an ELAN to exist without an LECS however, with a client obtaining this information by some other means. While such means are outside the LANE specification and therefore outside the scope of this document, it was the intention not to preclude such implementations. This is why the LECS portion of the ELAN MIB is a separate group, allowing the ELAN MIB to exist without requiring an LECS.

Systems which implements the LECS must implement the LECS group , to allow the LECS to be monitored and controlled. TLV information also resides in this part of the MIB since this is specific to the LECS configuration mechanism.

While the topology of ELANs can be controlled using the ELAN MIB, by determining which ELANs and LESs clients will use, this MIB does not reflect the current topology. To determine the current topology of an ELAN (which clients are attached to which LESs and BUSs) the LES MIB [2] and BUS MIB [3] (and the LEC MIB [1] if detailed information on the client is required) must be used.

The design goal is to decouple the configured and actual ELAN topology to achieve maximum flexibility in LEC assignment and also future LNNI (LAN Emulation Server to Server Protocol) support.

5.2 MIB Groups

This section briefly describes each group and object in the ELAN MIB. For more detail consult the MIB definition and the LANE Emulation specification [1.

5.2.1 ELAN Administration Group

This group provides a registry for the LEC assignment policy types. There are six LEC assignment policies defined in this group:

- byAtmAddr - assign a LEC to an ELAN by it's ATM address
- byMacAddr - assign a LEC to an ELAN by it's MAC address
- byRouteDescriptor - assign a LEC to an ELAN by it's Route Descriptor
- byLanType - assign a LEC to an ELAN by it's LAN type, i.e. ieee802.3 or ieee802.5
- byPktSize - assign a LEC to an ELAN by it's DataFrameSize
- byElanName - assign a LEC to an ELAN by its ELAN name

This table represents the set of assignment policy types which may be supported by the standard implementations of MIBs. These are based on the information available in a typical CONFIG REQUEST.

Vendors wishing to create new policies should **not** extend this table, but instead should define similar tables or objects in their own MIBs. Policy types are defined as OIDs (AutonomousType) to allow vendors to implement their own policies without needing centralised administration of policy types. This table may be extended in the future revisions of the ELAN MIB.

5.2.2 ELAN Configuration Group

This group provides configuration information for Emulated LANs. An ELAN is constructed or destroyed in this group for configuration purposes only; an ELAN can be created and many LECs can be assigned to this ELAN (by policies), but this MIB would not reflect whether or not LECs have actually joined the ELAN. The network manager has to poll the individual LECs or the LES/BUS to determine the actual ELAN topology.

For example, consider an ELAN 'red' in the ELAN Configuration table, where the LEC with the ATM address 'xxx...redx' is assigned to the 'red' ELAN in the LEC assignment by ATM address table. In other words, the LEC 'xxx...redx' is configured to ELAN 'red' in the ELAN MIB. The LES with the ATM address 'yyy...redy' is one of the servers which serves the ELAN 'red' as defined in the ELAN MIB's LES table.

In order to determine whether the LEC 'xxx...redx' has actually joined the ELAN 'red', or joined the LES 'yyy...redy', the network manager must either read the LEC-to-LES information in the LES MIB from the agent of LES 'yyy...redy' or talk to the agent of the LEC 'xxx...redx' to obtain the LES information from the LEC MIB. In either case, the network manager can match the information from the LES 'yyy...redy' or LEC 'xxx...redx' to the LEC assignment table by ATM address or the LES table.

In the case where assignment policies are more broad, (e.g. a policy could assign all clients with LanType 'iee802.5' to the ELAN 'Token-Ring', or address masks might be used so that all clients with a certain ATM address prefix are directed to a particular ELAN) there will be no complete list of clients in the ELAN MIB. In this case a network manager would have to use the LES MIB to find a real list of clients.

In conclusion, the ELAN group provides statically configured information and the LES and LEC MIBs provide the dynamic status information for the Emulated LAN.

5.2.2.1 ELAN Configuration Table

elanConfTable lists all Emulated LANs on which this agent maintains information; the elanConfIndex object, an arbitrary integer used as an index in this table, is used elsewhere in the ELAN MIB to identify a particular ELAN. Emulated LANs should be created or destroyed from this table in conjunction with operations to create or destroy the related service components as described in section 5.1. This table can be used by the entity which assigns a LAN Emulation clients to ELANs. This might be a LAN Emulation

Configuration Server (LECS), or in an environment where there is no LECS, the information might be used directly by a system to configure LAN Emulation Clients (LECs).

A network manager can construct the ELAN configuration table as well as the LEC assignment table locally. And then configures the LEC according to the configuration through the LEC MIB defined in [6].

The table contains the ELAN's name, LAN type, maximum frame size, and also has a TLV selector to enable it to choose a set of TLV-defined properties (set to zero if the LECS group is not supported).

5.2.2.2 ELAN LES table

This table lists all LES associated with ELANs managed by the agent. Only the ATM address of the LES is held, and because the LANE specification permits multiple LES addresses per ELAN, there may be more than one LES per ELAN. All LEC assignment tables are indexed by both the ELAN (elanConfIndex) and LES index (elanLesIndex) to allow the selection of a specific LES address for any client within an ELAN.

5.2.2.3 ELAN Policy Table

Throughout this section, all terms are used and they are: 'accept' meaning the policy accepts the client (the client has the matching information of the correct type), 'rejected' meaning the policy rejects the client (the client didn't have the information or the information doesn't match), and 'failed' meaning the client was not assigned to any ELAN as the results of the execution of all the policies.

This table describes policies currently in use for assigning LECs to particular ELANs and LESs. These policies will be carried out by whatever entity is using this information, typically an LECS but not necessarily.

When a LECS is supported, the LECS assigns a client to an ELAN based on the information given in the CONFIG request. Information in the CONFIG request is checked against the various policy tables in the ELAN MIB.

There can be multiple policies used in determining the LEC assignment at one time. These policies are executed in order of their priorities, with the lowest number getting the highest priority. (A policy with priority of 1 would be executed first.) Conflicts between policies may occur but the LECS behavior is implementation dependent.

If there exist multiple policies of the same priority, they are executed at the same time, and all policies must succeed.

Policies are grouped with the elanPolicySelectorIndex. This allows disjoint policy sets to be created which allows individual LECSs (or other users of this data) to use different policies.

Users of the information in the policy and assignment tables should use the following procedure when attempting to assign clients:

- Find the first policy with the highest priority level with the appropriate PolicySelectorIndex.
- Check to see if the client matches any assignments for this policy type. For the standard policies this will involve looking up in an assignment table (e.g. for elanPolicyType of byAtmAddr, the elanLecAtmAddrTable must be consulted). If the client passes, there will be one or more ELAN/LES index pairs. If the client fails to find any appropriate ELAN/LES pair with this policy, it has failed, and must restart from the next priority down. If there are no policies with a lower priority than the current one, then the client is not assigned to any ELAN and has been rejected.
- If there are more policies with the same priority level as the one which just passed, these must also be evaluated as described above.
- If a client has passed all the policies with a particular priority level, there will have been at least one ELAN/LES pair generated per priority. If no pair is common to every policy (i.e. the client passed

every policy, but the policies failed to agree on any ELAN/LES pair) then the client has failed for this priority level, and behaviour is as if the client had failed against any one of the policies at this priority. If a single ELAN/LES pair can be identified as common to the results of all the policies, then this is taken as the final result. If there are multiple matches and the client could be assigned validly to more than one ELAN/LES, any may be chosen, the particular outcome being implementation defined.

An example configuration might consist of four ELANs: TRed, TBlue, ERed and EBlue. Clients with an ATM address prefix of rrrrrrrrrrrrrr would be assigned to the Red ELANs, TRed if they are Token-ring clients and ERed if they are Ethernet clients. Otherwise they get assigned to TBlue or EBlue again according to LAN type. This would require three ElanPolicyEntries, at two priority levels. There would be two entries at priority level one: a policy of type byAtmAddr and a policy of type byLanType. There would be one entry at priority level two: byLanType. The assignment by ATM address table would contain two entries, both with an ATM address beginning with rrrrrrrrrrrrrr and an address mask with only the relevant prefix bits set, one pointing to TRed and an appropriate LES index, the other pointing to ERed with an appropriate LES index. As for the byLanType**,** the object elanConfLanType in the elanConfTable is used. If the client's requesting LAN type is Ethernet, then the first elanConfEntry that has the elanConfLanType set to aflane8023(2) and it also matches the ATM prefix is the ELAN this client is assigned to.

If there is no data available on the client for a policy, then that policy would fail. The client is rejected by that policy.

The agent should refuse to create any policy it does not support.

5.2.2.4 LEC assignment table by ATM address

This table is indexed by the ELAN index which points to the ELAN this LEC belongs to, the particular LES within that ELAN it will be sent to, and the ATM address plus mask of the LEC. The ATM mask allows the network manager to specify portions of an ATM address. A client would be matched by an assignment entry if bits in its address corresponding to non-zero bits in the mask match the address in the table.

5.2.2.5 LEC assignment table by MAC address

This table is indexed by the ELAN index, LES index within the ELAN, and the MAC address (with mask) of the LEC. The MAC address is compared with the client's when executing byMacAddress policies.

5.2.2.6 LEC assignment table by Route Descriptor

This table is indexed by the ELAN index, the LES within the ELAN, and the Route Descriptor of the LEC, which consists of a segment id and a bridge number. The route descriptor is compared with the client's when executing byRouteDescriptor policies.

5.2.3 LECS Configuration Group

This enables network managers to configure and monitor LECSs. It also provides configuration information on TLV (type, length and value) entries to be used in CONFIG responses..

5.2.3.1 LECS Configuration table

This table contains the configuration and status information for all LECS managed by this agent. This table is used to create, delete or configure a LECS. Critical objects such as the lecsAtmIfIndex, lecsAtmAddrSpec, and lecsAtmAddrMask cannot be modified unless the lecsAdminStatus is set to down(2). That is, the network manager has to bring the LECS down to in order to change a LECS's ATM address or ATM interface value. These changes can be brought into effect by setting the lecsAdminStatus to up(1).

The object lecsAtmIfIndex provides the number of the ATM interface on which the LECS is listening for CONFIGURE requests. This value must match an existing value in the ifTable. This object is set to zero when the ATM interface is not specified or there is more than one ATM interface used by the LECS.

A network manager may instruct the LECS to attempt to register and listen on a particular address. This is achieved through use of the lecsAtmAddrSpec and lecsAtmAddrMask objects. The default values for these objects will cause a LECS to use the with a well-known ATM address (reference to LANE 1.0 specification section 5.2.1.2). Clearly if more than one LECS is created on any interface (or on multiple interfaces which would share the domain of the well known address) at least one of them should use some other address.

To enable different LECSs to enact different sets of policies, a policy selector index is set for each LECS.

5.2.3.2 TLV table

This table is used to configure TLVs (Type, Length, and Value) for each ELAN. The table is indexed by a selector index, the TLV tag and an index. The selector allows multiple TLVs to be grouped together, so different ELANs can use different sets of TLVs. The index allows multiple TLVs with the same tag but different values

Network manager can create TLVs and assign them to an ELAN by specifying corresponding selector indices in both the TLV table and the elanConfTable.

5.2.3.3 VCC table

This table contains the LE CONFIG VCCs from all LECs to the LECS. This table can be used by the network manager as the central place for tracing VCCs.

5.2.4 LECS Statistics Group

In addition to the LECS fault management group, the LECS statistics group also provides per LECS error counters.

5.3.4.1 LECS statistics table

This table lists all counters associated with the LECS this agent maintains.

5.2.5 LECS Fault Management Group

This group provides fault information for LECSs. The network manager can enable or disable the error logging capability of a LECS. The enabled LECS will log the error events until the maximum number error log entry is reached.

The logged events are saved in the LECS Error Log table.

5.2.5.41 LECS Error Log Control Table

This table is used to control error logging capability on a LECS. The network manager can enable/disable error logging of a particular LECS managed by the agent. It can also reset the error log of a LECS. The LECS must clear all the error log entries upon such a request.

The object lecsErrCtlMaxEntries describes the maximum number of errors a LECS can log. If this object is one, the LECS can only save the last error event. It is a read only object. The object lecsErrCtlLastEntry gives a pointer to the last error log saved by a LECS in the lecsErrLogTable. It can be used by the network manager to read the most recent entry.

5.2.5.2 LECS Error Log Table

This table contains all the error logs maintained by the LECSs managed by the agent. Each entry describes when the error occurred, the nature of the error and the ATM address of the client whose CONFIG request resulted in the error.

The lecsErrLogIndex ranges from 2 to 32 minus 1 down to 1. It is assigned consecutively in the descending order. The network manager can easily retrieve the most recent N entries by using the get-next on the value of this object. The entries after 1 are discarded. For example, if the lecsErrCtlMaxEntries is 5, and the LECS has received 2 to 32 minus 1 minus 100 entries so far. The values of the lecsErrLogIndex for this LECS in the lecsErrLogTable are 100, 99, 98, 97 and 96 with the entry 96 being the most recent error occurred.

6. LAN Emulation Server MIB Description

This MIB provides LES management information. The implementation of this MIB is mandatory for agents managing LESs.

6.1 Management model

The LES management model is designed to be compliant with the LAN Emulation 1.0 specification. It was also intended to facilitate the accommodation of future LAN Emulation Server to server protocol management extensions.

This MIB may be used to determine the distribution of the LECs among LESs, and to create, configure and monitor LESs.

6.2 MIB Organisation

The MIB is divided into four groups: the LES Configuration group which provides configuration and topology information; the LES Statistics Group which provides various counters for each LES; the LES-LEC Statistics group which provides per LES/LEC pair counters; the LES Fault management group which logs LES error information.

6.3 MIB Groups

6.3.1 LES Configuration Group

6.3.1.1 LES Configuration table

This table lists all LAN Emulation Servers managed by this agent.

The object lesAtmAddrSpec specifies an ATM address that, with the ATM address mask, determines a portion of the ATM address that the LES on the designated ATM interface will use to derive the actual ATM address from the network or ILMI. The derived ATM address is specified in the object lesAtmAddrActual, which is used to receive ATM ARP requests.

The object lesElanName gives the name of the Emulated LAN this LES is providing service for. This object may be used to identify the ELAN the LES is in.

The object lesLanType describes the type of the ATM Emulated LAN this LES is providing service to. If the LEC has specified a LAN type in its JOIN request that does not match with the value of this object, the LES will reject the request. Only 802.3 and 802.5 LAN types are supported.

6.3.1.2 LES VCC Table

This table lists all the Control Distribute VCCs used by the LES to distribute control traffic to the participating LECs. The Control Distribute VCC can either be point-to-point or point-to-multipoint calls. This table is read only if SVCs are used and writable if PVCs are used.

Each entry indicates the ATM interface number, VPI value and VCI value, which comprise the index to the atmVclTable in the AToMMIB.

6.3.1.3 LES ARP table by MAC address

This table provides access to an ATM LAN Emulation Server's MAC-to-ATM address table.. It contains entries for unicast addresses and the broadcast address.

An entry for the broadcast MAC address will have the ATM address of a BUS. When the entry is for a unicast MAC address the corresponding ATM address will be for a LEC.

6.3.1.4 LES ARP table by Route Descriptor

This table provides access to an ATM LAN Emulation Server's RouteDescriptor-to-ATM ARP cache. The entries in this table are set-up by the agent or network manager depending on the entry type.

The Route Descriptors are presented as Segment Id (ring number) and Bridge number.

6.3.1.5 LES-LEC topology table

This table lists all LAN Emulation clients serviced by LESs specified in the lesConfTable. This table can be used to retrieve the topology of an ELAN, i.e. the LES to LECs mapping information.

An entry in this table is created by the agent when a LEC registers successfully with the LES. The lesLecCtlDirectVpi and lesLecCtlDirectVci objects can be modified by the network manager if PVC is used.

6.3.2 LES Statistics Group

This table contains all counters the LESs maintain. This table augments the lesConfTable. It provides performance and fault counters on a per LES basis.

The table lists all error types listed in the Table 13 in "ATM Forum LAN Emulation Over ATM Specification, V1.0".

6.3.3 LES-LEC Statistics Group

This table contains all LE-ARP request related counters and error counts on a per LEC-LES pair basis.

6.3.4 LES Fault Management Group

This group provides fault management information for managing a LES. The network manager can enable/disable the error logging capability of a LES. The enabled LES will log the error event until the maximum number error log entry is reached.

The logged events are saved in the LES Error Log table.

6.3.4.1 LES Error Log Control Table

This table is used to control the error logging capability of a LES. The network manager may enable or disable error logging on a particular LES managed by the agent. It can also reset the error log of a LES.

The object lesErrCtlMaxEntries describes the maximum number of errors a LES can log. If this object is one, the LES will only save the last error event. The object lesErrCtlLastEntry gives a pointer to the last error log saved by a LES in the lesErrLogTable.

6.3.4.2 LEC Error Log Table

This table contains all the error logs maintained by the LESs managed by the same agent. Each entry describes the error occurred, the nature of the error and the ATM address of the client whose request that resulted in the error.

The lesErrLogIndex ranges from 2 to 32 minus 1 to 1. It is assigned consecutively in the descending order. The network manager can easily retrieve the most recent N entries by using the get-next on the value of this object. The entries after 1 are discarded. For example, if the lesErrCtlMaxEntries is 5, and the LES has received 100 entries so far. The values of the lesErrLogIndex for this LES in the lesErrLogTable are 100, 99, 98, 97 and 96 with the entry 100 being the most recent error occurred.

7. Broadcast and Unknown Server MIB description

7.1 Management model

This MIB enables network managers to create, destroy, configure and determine the current status of BUSs and topology of the portions of ELANs being served by BUSs.

7.2 MIB Organisation

The MIB is devided into two groups: BUS Configuration group which provides BUS topology information. And BUS Fault Management group which provides BUS statistics and BUS-LEC statistics.

7.3 MIB Groups

7.3.1 BUS Configuration Group

This group includes the object busNextId, BUS Configuration table, BUS VCC table and BUS-LEC topology table.

The object busNextId provides the network manager the next available index used to create a BUS.

7.3.1.1 BUS Configuration Table

This table lists all LAN Emulation Broadcast and Unknown Servers (BUS) this agent manages. The BUS handles data sent by an LE Client to the broadcast MAC address, all multicast traffic, and initial unicast frames which are sent by a LEC before the appropriate data direct target ATM address has been resolved. There can be multiple BUSs per ELAN, but a BUS can service only one ELAN.

7.3.1.2 BUS VCC Table

This table lists all the Multicast Forward VCCs used by the BUS to forward multicast traffic to the participating LECs. Multicast Forward VCCs can either be point-to-point or point-to-multipoint. This table is read only if SVCs are used and writable if PVCs are used.

7.3.1.3 BUS-LEC Topology Table

This table lists the actual LECs being serviced by the BUS. It can be used to determine the mapping between BUSs and LECs.

This table provides information for Multicast send VCCs between BUS and LE clients. Objects busLecMcastSendAtmIfIndex, busLecMcastSendVpi and busLecMcstSendVci can only be modified if PVC is used.

7.3.2 BUS Statistics Group

7.3.2.1 BUS Statistics Table

This table contains all counters maintained by BUSs. This table augments the busConfTable.

7.3.2.2 BUS-LEC Statistics Table

This table contains all LEC counters the BUS maintains.

7.4.2 BUS Fault Management Group

This group provides fault management information for managing a BUS. The network manager can enable/disable the error logging capability of a BUS. The enabled BUS will log the error event until the maximum number error log entry is reached.

The logged events are saved in the BUS Error Log table.

7.4.2.1 BUS Error Log Control Table

This table is used to control the error logging capability of a BUS. The network manager may enable or disable error logging on a particular BUS managed by the agent. It can also reset the error log of a BUS.

The object busErrCtlMaxEntries describes the maximum number of errors a BUS can log. If this object is one, the BUS will only save the last error event. The object busErrCtlLastEntry gives a pointer to the last error log saved by a BUS in the busErrLogTable.

7.4.2.2 LEC Error Log Table

This table contains all the error logs maintained by the BUSs managed by the same agent. Each entry describes the error occurred, the nature of the error and the ATM address of the client whose frame cause the error.

The busErrLogIndex ranges from 2 to 32 minus 1 to 1. It is assigned consecutively in the descending order. The network manager can easily retrieve the most recent N entries by using the get-next on the value of this object. The entries after 1 are discarded. For example, if the busErrCtlMaxEntries is 5, and the BUS has received 100 entries so far. The values of the busErrLogIndex for this BUS in the busErrLogTable are 100, 99, 98, 97 and 96 with the entry 100 being the most recent error occurred.

8. LAN Emulation Servers MIB Definitions

```
--
--      MIB for managing configuration data for ATM Emulated LANs
--

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

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

                TEXTUAL-CONVENTION, DisplayString,
                RowStatus, MacAddress,
                AutonomousType, TimeStamp              FROM SNMPv2-TC

                MODULE-COMPLIANCE, OBJECT-GROUP        FROM SNMPv2-CONF

                LecDataFrameSize, LecDataFrameFormat,
                AtmLaneAddress, atmfLanEmulation,
                VpiInteger, VciInteger                 FROM LAN-EMULATION-CLIENT-MIB;

                elanMIB MODULE-IDENTITY
                LAST-UPDATED "9602121200Z"
                ORGANIZATION "ATM Forum LAN Emulation Sub-Working Group"
                CONTACT-INFO
                      "
                        The ATM Forum

                        2570 West El Camino Real,
                        Ste 304,
                        Mountain View, CA 94040-1313
                        Tel: 415-949-6711
                        E-mail: info@atmforum.com"

                DESCRIPTION
                    "This is the MIB module for managing ATM
                     Emulated LANs as well as LAN Emulation
                     Configuration Server (LECS).  It provides
                     mechanism for constructing and destroying
                     ELANs, for distributing members of an
                     ELAN and for figuring out the topology
                     of an ELAN.  It also provides management
                     information for LECS."
                ::= { atmfLanEmulation 2 }


            --
            --    Textual Conventions
            --

            IfIndexOrZero  ::= TEXTUAL-CONVENTION
                STATUS     current
                DESCRIPTION
                    "An integer defined as to equivalent
                     to ifIndex in the ifTable defined
                     in RFC 1213 and the value zero."
                 SYNTAX     INTEGER (0..65535)

            ElanLocalIndex ::= TEXTUAL-CONVENTION
                STATUS     current
                DESCRIPTION
                    "A value which uniquely identifies a
```

```
          conceptual row in the elanConfTable.
          This number is only used locally by the agent to
          distinguish between ELANs."
     SYNTAX     Integer32

AtmLaneMask ::= TEXTUAL-CONVENTION
   STATUS     current
   DESCRIPTION
         "A twenty-octet binary string, containing a standard
          ATM Forum address mask."
   SYNTAX      OCTET STRING (SIZE(20))

TlvSelectorIndexType ::= TEXTUAL-CONVENTION
    STATUS     current
    DESCRIPTION
         "An arbitrary integer used in identifying
          a set of TLV encoding.  This number
          is locally managed by the agent."
    SYNTAX     Integer32

PolicySelectorIndexType ::= TEXTUAL-CONVENTION
    STATUS     current
    DESCRIPTION
         "An arbitrary integer used in identifying
          a set of LEC assignment policies.
          This number is locally managed by the
         agent."
    SYNTAX     Integer32

LecsErrLogIndexType ::= TEXTUAL-CONVENTION
    STATUS     current
    DESCRIPTION
         "An arbitrary integer used in identifying
          an error log associated with a LECS."
    SYNTAX     Integer32(1..2147483647)


-- This MIB module consists of the following groups:
--
--  (1) ELAN Administration Group
--      (a) ELAN Admin Policy Table
--
--  (2) ELAN Configuration Group
--      (a) Elan Conf table
--      (b) LES table
--      (c) ELAN policy table
--      (d) LEC assignment table by ATM address
--      (e) LEC assignment table by MAC address
--      (f) LEC assignment table by Route Descriptor
--
--  (3) LECS Group
--     (3.1) LECS Configuration Group
--      (a) LECS configuration table
--      (b) LECS to ELAN mapping table
--      (c) TLV (Type, Length and Value) table
--     (3.2) LECS Fault Management Group
--      (c) LECS log control table
--      (d) LECS log table
--     (3.3) LECS Statistics Group
--      (c) LECS Statistics table
--
--


    elanAdminGroup OBJECT IDENTIFIER ::= { elanMIB 1 }

    elanConfGroup  OBJECT IDENTIFIER ::= { elanMIB 2 }

    elanLecsGroup  OBJECT IDENTIFIER ::= { elanMIB 3 }

    elanLecsConfGroup OBJECT IDENTIFIER
        ::= { elanLecsGroup 1 }
```

```
elanLecsFaultGroup OBJECT IDENTIFIER
    ::= { elanLecsGroup 2 }

elanLecsStatGroup OBJECT IDENTIFIER
    ::= { elanLecsGroup 3 }


--
--  (1) ELAN Administration Group
--       (a) ELAN Admin Policy Registry
-- The following are defined for posssible
--           values of the elanPolicytype object
--

elanAdminPolicyVal  OBJECT IDENTIFIER ::= { elanAdminGroup 1 }

-- assign LEC to an ELAN by it's ATM address & Mask
byAtmAddr OBJECT IDENTIFIER ::= { elanAdminPolicyVal 1 }

-- assign LEC to an ELAN by it's MAC address
byMacAddr OBJECT IDENTIFIER ::= { elanAdminPolicyVal 2 }

-- assign LEC to an ELAN by it's Route Descriptor
byRouteDescriptor OBJECT IDENTIFIER ::= { elanAdminPolicyVal 3 }

-- assign LEC to an ELAN by it's LAN type
byLanType  OBJECT IDENTIFIER ::= { elanAdminPolicyVal 4 }

-- assign LEC to an ELAN by it's Packet size
byPktSize  OBJECT IDENTIFIER ::= { elanAdminPolicyVal 5 }

-- assign LEC to an ELAN by it s ELAN name
byElanName OBJECT IDENTIFIER ::= { elanAdminPolicyVal 6 }

               --
--  (2) ELAN Configuration Group
--
--      (a) ELAN Conf table
--

   elanConfNextId  OBJECT-TYPE
       SYNTAX    ElanLocalIndex
       MAX-ACCESS read-only
       STATUS    current
       DESCRIPTION
            "The next available ELAN index provided
             by the agent.  The value of this object
             can be used as the index to the
             elanConfTable during creation."
       ::= { elanConfGroup 1 }

   elanConfTable   OBJECT-TYPE
       SYNTAX       SEQUENCE OF ElanConfEntry
       MAX-ACCESS  not-accessible
       STATUS      current
       DESCRIPTION
            "This table contains all Emulated LANs
            (ELANs) this agent manages.  An ELAN
            is defined by the ELAN name, a set of
            TLVs, and other parameters.  After an
            ELAN is created, members of the ELAN
            such as the LAN Emulation Server (LES)
            or Client (LEC) can be added to this
            ELAN entry in their repective tables.
            There are four memebers to an ELAN and
            they are LECS, LES, BUS (Broadcast and
            Unknown Server ) and LEC.  The support
            of LECS is optional.  The addition and
            deletion of LECS is done in the
            elanLecsConfGroup defined in this MIB.
            The addition and deletion of the LEC
```

```
          are done in the LEC Assignment tables
          defined in this ELAN Configuration group.
          The addition and deletion of the LES
          are also defined in this group.  The
          addition and deletion of the BUS are
          done in the LES MIB due to the fact that
          LEC learns the BUS address only from
          the LES it corresponds to."
     ::= { elanConfGroup 2 }

elanConfEntry    OBJECT-TYPE
     SYNTAX       ElanConfEntry
     MAX-ACCESS   not-accessible
     STATUS       current
     DESCRIPTION
          "Each entry in this table represents an
          Emulated LAN.  Objects elanConfIndex
          and elanConfRowStatus are required
          during row creation and deletion. "
     INDEX { elanConfIndex }
     ::= { elanConfTable 1 }

ElanConfEntry ::=
     SEQUENCE {
          elanConfIndex
              ElanLocalIndex,
          elanConfName
              DisplayString,
          elanConfTlvIndex
              TlvSelectorIndexType,
          elanConfLanType
              LecDataFrameFormat,
          elanConfMaxFrameSize
              LecDataFrameSize,
          elanConfRowStatus
              RowStatus
          }

elanConfIndex     OBJECT-TYPE
     SYNTAX        ElanLocalIndex
     MAX-ACCESS    not-accessible
     STATUS        current
     DESCRIPTION
           "A value which uniquely identifies a
            conceptual row in the elanConfTable.

            If the conceptual row identified by this value
            of elanConfIndex is recreated following an agent
            restart, the same value of elanConfIndex must be
            used to identify the recreated row."
     ::= { elanConfEntry 1 }

elanConfName      OBJECT-TYPE
     SYNTAX          DisplayString(SIZE(0..32))
     MAX-ACCESS     read-create
     STATUS         current
     DESCRIPTION
            "The name of this Emulated LAN entry.
            When this object length is zero then the
            ELAN name is not specified.  The clients
            assigned to this ELAN will also have
            a zero length string as the ELAN name.

            The value of this object is used in the
            LE CONFIGURE response by the LECS if supported.

            Note that ELAN name may be used as cross
            reference to the LES MIB and BUS MIB
            though not required.  Multiple ELANs
            with no ELAN name specified will cause
            conflicts in reference to LES and BUS MIBs."
     REFERENCE
```

```
              "LAN Emulation Over ATM Specification -
              version 1.0. C5."
          DEFVAL { "" }
          ::= { elanConfEntry 2 }

      elanConfTlvIndex   OBJECT-TYPE
          SYNTAX          TlvSelectorIndexType
          MAX-ACCESS      read-create
          STATUS          current
          DESCRIPTION
                "The value of this object identifies
                one or more rows in the lecsTlvTable which
                applies to this ELAN.  This object is
                set to zero if 1) LECS is not supported.
                or 2) there is no TLV associated with
                this entry."

          ::= { elanConfEntry 3 }


          elanConfLanType     OBJECT-TYPE
              SYNTAX          LecDataFrameFormat
              MAX-ACCESS      read-create
              STATUS          current
              DESCRIPTION
                  "The LAN type of this ELAN entry."
              REFERENCE
                  "LAN Emulation Over ATM Specification -
                  version 1.0. S2."
              DEFVAL { unspecified }
              ::= { elanConfEntry 4 }

          elanConfMaxFrameSize   OBJECT-TYPE
              SYNTAX             LecDataFrameSize
              MAX-ACCESS         read-create
              STATUS             current
              DESCRIPTION
                   "The maximum data frame size of this
                   ELAN entry.  The maximum AAL-5 SDU size
                   of a data frame that this ELAN can support.
                   The value of this object is returned to the
                   LEC in the LE CONFIGURE response."
              REFERENCE
                   "LAN Emulation Over ATM Specification -
                  version 1.0. S3."
              DEFVAL { unspecified }
              ::= { elanConfEntry 5 }

          elanConfRowStatus        OBJECT-TYPE
              SYNTAX               RowStatus
              MAX-ACCESS           read-create
              STATUS               current
              DESCRIPTION
                   "This object is used to create or
                    delete entries in the elanConfTable."
               REFERENCE "RFC 1443, [10] Textual Conventions
                    for version 2 of the Simple Network Management
                    Protocol (SNMPv2)."
              ::= { elanConfEntry 6 }

      --
      --      (b) LES table
      --

          elanLesTable       OBJECT-TYPE
                SYNTAX       SEQUENCE OF ElanLesEntry
                MAX-ACCESS   not-accessible
                STATUS       current
                DESCRIPTION
                     "This table contains all LESs for each
                  ELAN specified in the elanConfTable.
                      Each ELAN can have more than
```

```
                one LES providing LAN Emulation
                services.  Each LES can service only
            one ELAN.  The table is indexed by the
            elanConfIndex which points to the ELAN
            this LES is providing service to, and
            elanLesIndex which unquely identifies
            a LES.  This table is used for configuration
            of an ELAN only, that is, creating a
            LES in this table does not instantiate
            a LES in the network.  It is done in
            the LES MIB."
        ::= { elanConfGroup 3 }

elanLesEntry    OBJECT-TYPE
    SYNTAX      ElanLesEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
         "Each entry in this table represents
          a LES/Emulated LAN pair.  Object
          elanLesAtmAddress besides elanLesRowStatus
          is also required during row creation."
    INDEX { elanConfIndex, elanLesIndex }
    ::= { elanLesTable 1 }

ElanLesEntry ::=
    SEQUENCE {
        elanLesIndex
            Integer32,
        elanLesAtmAddress
            AtmLaneAddress,
        elanLesRowStatus
            RowStatus
    }

elanLesIndex       OBJECT-TYPE
    SYNTAX         Integer32
    MAX-ACCESS     not-accessible
    STATUS         current
    DESCRIPTION
         "An arbitrary number which uniquely
         identifies the LES this entry pertains
        to."
    ::= { elanLesEntry 1 }

elanLesAtmAddress  OBJECT-TYPE
    SYNTAX         AtmLaneAddress
    MAX-ACCESS     read-create
    STATUS         current
    DESCRIPTION
         "The ATM address of the LES entry.
         If LECS is supported, the value of
         this object is the LES ATM address
         LECS returns to the LEC in the
         CONFIGURE response.  If LECS
         is not supported, the value of this
         object pertains to the LES ATM address
         network manager provides to the
        LEC."
    DEFVAL { "" }
    ::= { elanLesEntry 2 }

elanLesRowStatus   OBJECT-TYPE
    SYNTAX         RowStatus
    MAX-ACCESS     read-create
    STATUS         current
    DESCRIPTION
         "This object is used to create or
          delete entries in the elanLesfTable."
     REFERENCE "RFC 1443, [10] Textual Conventions
          for version 2 of the Simple Network Management
         Protocol (SNMPv2)."
```

```
        ::= { elanLesEntry 3 }

--
--      (c) ELAN policy table
--
    elanPolicyTable   OBJECT-TYPE
        SYNTAX      SEQUENCE OF ElanPolicyEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
              "This table contains all policies this
              agent supports for assigning a LEC to
              an ELAN.
              A set of policies with the same or
              different priorities can be selected
              by the entity which provides ELAN
              configuration service such
              as the LECS.  The policy with the
              highest priority or with the smallest
              elanPolicyPriority , is evaluated
              first.  The policies with the same
              elanPolicyPriority are evaluated
              at the same time with the AND operation.
              When LECS receives a configure
              request, it checks it's policies selected
              from this table to determine which
              ELAN and LES the LEC will join.
              This table is indexed by a selector
              index and a policy index.  The policy
              index unquely identifies a policy and
              the selector index allows multiple
              policies be selected by one LECS or
              an entity that is providing ELAN
              configuration service."
        ::= { elanConfGroup 4 }

    elanPolicyEntry    OBJECT-TYPE
        SYNTAX         ElanPolicyEntry
        MAX-ACCESS     not-accessible
        STATUS         current
        DESCRIPTION
            "Each entry in this table represents a
            policy supported by the entity which
            provides ELAN configuration
            services.  Each policy can be used to
            evaluate the CONFIGURE request from
            the LEC in determining which ELAN it
            belongs to. "
        INDEX { elanPolicySelectorIndex, elanPolicyIndex }
        ::= { elanPolicyTable 1 }

    ElanPolicyEntry ::=
        SEQUENCE {
            elanPolicySelectorIndex
                PolicySelectorIndexType,
            elanPolicyIndex
                Integer32,
            elanPolicyPriority
                Integer32,
            elanPolicyType
                AutonomousType,
            elanPolicyRowStatus
                RowStatus
        }

    elanPolicySelectorIndex      OBJECT-TYPE
        SYNTAX        PolicySelectorIndexType
        MAX-ACCESS    not-accessible
        STATUS        current
        DESCRIPTION
            "The value of this object indicates
            a group of policies that can be selected
```

```
                by the ELAN configuration service
                    provider such as the LECS."
            ::= { elanPolicyEntry 1 }

        elanPolicyIndex      OBJECT-TYPE
            SYNTAX   Integer32 (1..65000)
            MAX-ACCESS       not-accessible
            STATUS           current
            DESCRIPTION
                "The value of this object uniquely
                 identifies a single policy entry in
                 this table. "
            ::= { elanPolicyEntry 2 }

        elanPolicyPriority  OBJECT-TYPE
            SYNTAX           Integer32 (1..65000)
            MAX-ACCESS       read-create
            STATUS           current
            DESCRIPTION
                 "The priority of this policy entry.
                  Policies are evaluated by the
                  entity which provides ELAN configuration
                  service (LECS) by their
                  priorities.  Policies with the
                  same priority values should be
                  evludated at the same time with
                  an AND operation.  That is, if
                  one of the policy with the same
                  priority is not met,
                  the evluation should fail.
                  The value 1 has the highest priority. "
::= { elanPolicyEntry 3 }

        elanPolicyType      OBJECT-TYPE
            SYNTAX          AutonomousType
            MAX-ACCESS       read-create
            STATUS           current
            DESCRIPTION
               "
                The value of this object must
                reference a definition of a type
                of policy.  Some of such definition
                exist within the elanAdminPolicyVal
                subtree.  Others may be defined
                within enterprise specific subtrees.
                The agent is not required to
                support every types defined
                within the elanAdminPolicyVal subtree."
            ::= { elanPolicyEntry 4 }

       elanPolicyRowStatus OBJECT-TYPE
           SYNTAX           RowStatus
           MAX-ACCESS       read-create
           STATUS           current
           DESCRIPTION
               "This object is used to create or
                delete entries in the elanPolicyTable."
           REFERENCE "RFC 1443, [10] Textual Conventions
                for version 2 of the Simple Network Management
                Protocol (SNMPv2)."
           ::= { elanPolicyEntry 5 }


      --
      --      (d) LEC assignment table by ATM address
      --

         elanLecAtmAddrTable  OBJECT-TYPE
               SYNTAX       SEQUENCE OF ElanLecAtmAddrEntry
               MAX-ACCESS  not-accessible
               STATUS      current
               DESCRIPTION
```

```
            "This table is used to assign a LEC
             to an ELAN by ATM address.
             When the by ATM address policy is
             used, this table is used to specify
             the LEC's ATM address or portion of an ATM
             address.  The ATM address is used by
             the LECS or other entity which serves
             the LANE configuration function to
             determine the ELAN membership.

             This table is indexed
             by the elanConfIndex which points
             to the ELAN this LEC belongs, the
             elanLesIndex which points to the
             LES this LEC should join, the
             LEC's ATM address and an ATM address
             mask.  Portions of ATM address
             can be used in dertermining ELAN
             membership by using both the ATM
             address and the mask."
        ::= { elanConfGroup 5 }

    elanLecAtmAddrEntry   OBJECT-TYPE
        SYNTAX         ElanLecAtmAddrEntry
        MAX-ACCESS     not-accessible
        STATUS         current
        DESCRIPTION
            "Each entry represents a LEC to ELAN
            binding."
        INDEX { elanConfIndex, elanLesIndex,
                elanLecAtmAddress , elanLecAtmMask }
        ::= { elanLecAtmAddrTable 1 }

    ElanLecAtmAddrEntry ::=
        SEQUENCE {
            elanLecAtmAddress
                AtmLaneAddress,
            elanLecAtmMask
                AtmLaneAddress,
            elanLecAtmRowStatus
                RowStatus
    }

    elanLecAtmAddress      OBJECT-TYPE
        SYNTAX             AtmLaneAddress
        MAX-ACCESS         not-accessible
        STATUS             current
        DESCRIPTION
            "The value of this object is the
            ATM address of a LAN Emulation
            client.  This object and the
            object elanLecAtmMask are used
            to form an ATM address or portion
            of an ATM address to be used
            by the LECS in determining the
            ELAN membership when the policy
            of this LECS is by ATM address."
        ::= { elanLecAtmAddrEntry 1 }

    elanLecAtmMask         OBJECT-TYPE
        SYNTAX             AtmLaneAddress
        MAX-ACCESS         not-accessible
        STATUS             current
        DESCRIPTION
             "The ATM address mask associated with
             the object elanLecAtmAddress.  The value
             of the mask is an ATM address with
             the don't care portion set to zero
             and the valid ATM address portion set to one."
        ::= { elanLecAtmAddrEntry 2 }

    elanLecAtmRowStatus   OBJECT-TYPE
```

```
        SYNTAX            RowStatus
        MAX-ACCESS        read-create
        STATUS            current
        DESCRIPTION
            "This object is used to create or
            destroy entries in this table."
         REFERENCE "RFC 1443, [10] Textual Conventions
              for version 2 of the Simple Network Management
              Protocol (SNMPv2)."
        ::= { elanLecAtmAddrEntry 4 }

--
--      (e) LEC assignment table by MAC address
--
    elanLecMacAddrTable  OBJECT-TYPE
          SYNTAX      SEQUENCE OF ElanLecMacAddrEntry
          MAX-ACCESS  not-accessible
          STATUS      current
          DESCRIPTION
              "This table is used to assign a LEC
               to an ELAN by MAC address.
               When the byMacAddress policy is
               used, this table is used to specify
               the LEC's ELAN membership by ATM
               address.  This table is indexed
               by the elanConfIndex which points
               to the ELAN this LEC belongs, the
               elanLesIndex which points to the
               LES this LEC should join, and the
               LEC's MAC address."
         ::= { elanConfGroup 6 }

    elanLecMacAddrEntry   OBJECT-TYPE
        SYNTAX          ElanLecMacAddrEntry
        MAX-ACCESS      not-accessible
        STATUS          current
        DESCRIPTION
            "Each entry represents a LEC to ELAN
            binding."
        INDEX { elanConfIndex, elanLesIndex,
                elanLecMacAddress  }
        ::= { elanLecMacAddrTable 1 }

    ElanLecMacAddrEntry ::=
        SEQUENCE {
            elanLecMacAddress
                MacAddress,
            elanLecMacRowStatus
                RowStatus
        }

    elanLecMacAddress      OBJECT-TYPE
        SYNTAX            MacAddress
        MAX-ACCESS        not-accessible
        STATUS            current
        DESCRIPTION
            "The value of this object is the
            MAC address of a LAN Emulation
            client.  When a CONFIGURE request
            is received with this MAC address,
            it will be assigned to the ELAN
            which elanConfIndex is pointing
            to."
        ::= { elanLecMacAddrEntry 1 }

    elanLecMacRowStatus   OBJECT-TYPE
        SYNTAX            RowStatus
        MAX-ACCESS        read-create
        STATUS            current
        DESCRIPTION
            "This object is used to create or
            destroy entries in this table."
```

```
        REFERENCE "RFC 1443, [10] Textual Conventions
             for version 2 of the Simple Network Management
             Protocol (SNMPv2)."
        ::= { elanLecMacAddrEntry 2 }


--
--      (f) LEC assignment table by Route Descriptor
--

    elanLecRdTable  OBJECT-TYPE
        SYNTAX      SEQUENCE OF ElanLecRdEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
             "This table is used to assign a LEC
          to an ELAN by Route Descriptor.
             When the by Route Descriptor policy is
          used, this table is used to specify
          the LEC's ELAN membership by Route
          Descriptor.  This table is indexed
          by the elanConfIndex which points
          to the ELAN this LEC belongs, the
          elanLesIndex which points to the
          LES this LEC should join, and the
          LEC's Route Descriptors."
        ::= { elanConfGroup 7 }

    elanLecRdEntry   OBJECT-TYPE
        SYNTAX          ElanLecRdEntry
        MAX-ACCESS      not-accessible
        STATUS          current
        DESCRIPTION
             "Each entry represents a LEC to ELAN
            binding. "
        INDEX { elanConfIndex, elanLesIndex,
                elanLecRdSegId, elanLecRdBridgeNum }
        ::= { elanLecRdTable 1 }
     ElanLecRdEntry ::=
        SEQUENCE {
            elanLecRdSegId
                Integer32,
            elanLecRdBridgeNum
                Integer32,
            elanLecRdRowStatus
                RowStatus
        }

    elanLecRdSegId     OBJECT-TYPE
        SYNTAX            Integer32 (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."
        ::= { elanLecRdEntry 1 }

    elanLecRdBridgeNum    OBJECT-TYPE
        SYNTAX            Integer32 (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."
        ::= { elanLecRdEntry 2 }

    elanLecRdRowStatus   OBJECT-TYPE
        SYNTAX            RowStatus
        MAX-ACCESS        read-create
        STATUS            current
```

```
        DESCRIPTION
             "This object is used to create or
             destroy entries in this table."
              REFERENCE "RFC 1443, [10] Textual Conventions
              for version 2 of the Simple Network Management
             Protocol (SNMPv2)."
        ::= { elanLecRdEntry 4 }

--
 --      (g) LEC assignment table by Packet Size
--

    elanLecPktSizeTable  OBJECT-TYPE
        SYNTAX      SEQUENCE OF ElanLecPktSizeEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "This table is used to assign a LEC
            to an ELAN by Packet Size. When the by
            Packet Size policy is used, this table
            is  used  to  specifythe  LEC's  ELAN
            membership by Packet Size.  This table
            is indexed by the elanConfIndex which
            points to the ELAN this LEC belongs, the
            elanLesIndex which points to the
            LES this LEC should join, and the
            LEC's Packet Size."
        ::= { elanConfGroup 8 }

    elanLecPktSizeEntry   OBJECT-TYPE
        SYNTAX         ElanLecPktSizeEntry
        MAX-ACCESS     not-accessible
        STATUS         current
        DESCRIPTION
             "Each entry represents a LEC to ELAN
            binding. "
        INDEX { elanConfIndex, elanLesIndex,
                elanLecFrameSize }
        ::= { elanLecPktSizeTable 1 }

     ElanLecPktSizeEntry ::=
        SEQUENCE {
            elanLecFrameSize
                LecDataFrameSize,
            elanLecPktSizeRowStatus
                RowStatus
        }

    elanLecFrameSize     OBJECT-TYPE
        SYNTAX             LecDataFrameSize
        MAX-ACCESS         not-accessible
        STATUS             current
        DESCRIPTION
            "The maximum AAL-5 SDU size this LEC can
          support. "
        ::= { elanLecPktSizeEntry 1 }


    elanLecPktSizeRowStatus   OBJECT-TYPE
        SYNTAX             RowStatus
        MAX-ACCESS         read-create
        STATUS             current
        DESCRIPTION
             "This object is used to create or
             destroy entries in this table."
        REFERENCE "RFC 1443, [10] Textual Conventions
              for version 2 of the Simple Network Management
             Protocol (SNMPv2)."
        ::= { elanLecPktSizeEntry 2 }
--
 --      (e) LEC assignment table by ELAN name
--
```

```
elanLecNameTable  OBJECT-TYPE
     SYNTAX      SEQUENCE OF ElanLecNameEntry
     MAX-ACCESS  not-accessible
     STATUS      current
     DESCRIPTION
        "This table is used to assign a LEC
        to an ELAN by ELAN name. When the by
        ELAN name policy is used, this table is
        used to specifythe LEC's ELAN membership
        by ELAN name.  This table is indexed by
        the elanConfIndex which points to the
        ELAN this LEC belongs, the elanLesIndex
        which points to the
        LES this LEC should join, and the
        LEC's ELAN name."
     ::= { elanConfGroup 9 }

elanLecNameEntry   OBJECT-TYPE
     SYNTAX         ElanLecNameEntry
     MAX-ACCESS     not-accessible
     STATUS         current
     DESCRIPTION
         "Each entry represents a LEC to ELAN
         binding. "
     INDEX { elanConfIndex, elanLesIndex,
             elanLecElanName }
     ::= { elanLecNameTable 1 }

 ElanLecNameEntry ::=
     SEQUENCE {
         elanLecElanName
             DisplayString,
         elanLecNameRowStatus
             RowStatus
     }

elanLecElanName      OBJECT-TYPE
     SYNTAX            DisplayString(SIZE(0..32))
     MAX-ACCESS        not-accessible
     STATUS            current
     DESCRIPTION
        "The name of the ELAN this LEC belongs. "
     ::= { elanLecNameEntry 1 }


elanLecNameRowStatus   OBJECT-TYPE
     SYNTAX            RowStatus
     MAX-ACCESS        read-create
     STATUS            current
     DESCRIPTION
         "This object is used to create or
         destroy entries in this table."
     REFERENCE "RFC 1443, [10] Textual Conventions
          for version 2 of the Simple Network Management
          Protocol (SNMPv2)."
     ::= { elanLecNameEntry 2 }
```

```
--
--  (2) LECS Group
--      (2.1) LECS Configuration Group
--            (a) LECS Configuration table
--            (b) LECS to ELAN mapping table
--            (c) LECS TLV table
--            (d) LECS VCC table
--      (2.3) LECS Statistics Group
--            (a) LECS Statistics table
--      (2.2) LECS Fault Management Group
--            (a) LECS Fault Control table
--            (b) LECS Error Log table


        lecsConfNextId  OBJECT-TYPE
            SYNTAX    ElanLocalIndex
            MAX-ACCESS read-only
            STATUS    current
            DESCRIPTION
                "The next available LECS index.  The
                value of this object can be used as
                the index to the lecsTable during
                creation."
            ::= { elanLecsConfGroup 1 }


        lecsConfTable OBJECT-TYPE
            SYNTAX     SEQUENCE OF LecsConfEntry
            MAX-ACCESS not-accessible
            STATUS     current
            DESCRIPTION
                "This table contains the configuration
                information of all LECS this agent
                manages.  This table can also be
                used to create, delete or configure
                a LECS."
            ::= { elanLecsConfGroup 2 }

        lecsConfEntry OBJECT-TYPE
            SYNTAX     LecsConfEntry
            MAX-ACCESS not-accessible
            STATUS     current
            DESCRIPTION
                "Each entry represents a LECS this
                agent maintains.
                Objects lecsAtmIfIndex, lecsAtmAddrSpec,
                and lecsAtmAddrMask cannot be modified
                unless the lecsAdminStatus is set to
                down(2).  And the change does not take
                effect until the lecsAdminStatus is set
                to up(1)."
            INDEX { lecsConfIndex }
            ::= { lecsConfTable 1 }

        LecsConfEntry ::= SEQUENCE {
            lecsConfIndex           Integer32,
            lecsAtmIfIndex          IfIndexOrZero,
            lecsAtmAddrSpec         AtmLaneAddress,
            lecsAtmAddrMask         AtmLaneMask,
            lecsAtmAddrActual       AtmLaneAddress,
            lecsPolicySelIndex      PolicySelectorIndexType,
            lecsLastInitialized     TimeStamp,
            lecsOperStatus          INTEGER,
            lecsAdminStatus         INTEGER,
            lecsRowStatus           RowStatus
        }

        lecsConfIndex OBJECT-TYPE
            SYNTAX     Integer32
            MAX-ACCESS not-accessible
            STATUS     current
```

```
    DESCRIPTION
         "An arbitrary integer which represents
         a LECS this agent manages.
         If the conceptual row identified by this value
         of lecsConfIndex is recreated following an agent
         restart, the same value of elanConfIndex must be
         used to identify the recreated row."
    ::= { lecsConfEntry 1 }

lecsAtmIfIndex OBJECT-TYPE
    SYNTAX     IfIndexOrZero
    MAX-ACCESS read-create
    STATUS     current
    DESCRIPTION
         "An ATM interface which the LECS receives
          CONFIGURE requests
          from.  This value must match an existing
          value  in the   ifTable.  This object
          is set to zero when the ATM interface
          is not specified or there is more than
          one ATM interface used by the LECS."
    DEFVAL { '0'H }
    ::= { lecsConfEntry 2 }

lecsAtmAddrSpec OBJECT-TYPE
    SYNTAX     AtmLaneAddress
    MAX-ACCESS read-create
    STATUS     current
    DESCRIPTION
         "An ATM  address specified by  the
         network or local management that,
         with the ATM address mask, determines
         a  portion of  the  ATM address  that
         the  LECS on the designated ATM interface
         will use  to derive the actual ATM
         address from the network or ILMI.  The derived
         ATM address is specified in the object
         lecsAtmAddrActual, which is used to
         receive CONFIGURE requests.
         The value of this object, if not
         specified, is defaulted to the
         well-known LECS ATM address specified
         in the LAN Emulation Spec. 1.0."
     DEFVAL { '47007900000000000000000000000A03E00000100'H }
    ::= { lecsConfEntry 3 }

lecsAtmAddrMask OBJECT-TYPE
    SYNTAX     AtmLaneMask
    MAX-ACCESS read-create
    STATUS     current
    DESCRIPTION
          "The ATM address mask associated with
          the object lecsAtmAddrSpec.  The value
          of the mask is an ATM address with
          the don't care portion set to zero
          and the valid ATM address portion set to one."
     DEFVAL { 'FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF'H }
    ::= { lecsConfEntry 4 }

lecsAtmAddrActual OBJECT-TYPE
    SYNTAX     AtmLaneAddress
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
         " The resulting ATM address that the
         LECS is accepting CONFIGURE
         requests on the interface indicated
         by the object lecsAtmIfIndex.
         This  address is   the  result of  the
         specified   ATM address,  its   mask   and
         interaction through the ILMI  with  the
         switch. Note  that   this object is only
```

```
              valid  when the corresponding lecsOperStatus
             is 'up'."
        ::= { lecsConfEntry 5 }

    lecsPolicySelIndex       OBJECT-TYPE
        SYNTAX     PolicySelectorIndexType
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
             "The set of policies used
             by this LECS in dertermining
             requesters ELAN membership.
             The policies are defined in
             the elanPolicyTable.  The value
             of this object must exist in the
             elanPolicyTable."
        ::= { lecsConfEntry 6 }

    lecsLastInitialized OBJECT-TYPE
        SYNTAX     TimeStamp
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
             "The value of the sysUpTime since
              LECS has last entered the state
              indicated by the object lecsOperStatus."
        ::= { lecsConfEntry 7 }

    lecsOperStatus OBJECT-TYPE
        SYNTAX     INTEGER {
                   other(1), -- unspecified
                   up(2),    -- LECS is accepting
                             -- CONFIGURE request
                   down(3)   -- LECS is not accepting
                             -- CONFIGURE request
                  }
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
                 " This  object reflects the actual
                 state of  the LECS which  may differ
                 from that   of the  lecsAdminStatus
                 object.   This     can occur    when
                 the  interface ifOperStatus     is
                 'down'   but the   corresponding
                 lecsAdminStatus is 'up'."
        ::= { lecsConfEntry 8 }

    lecsAdminStatus OBJECT-TYPE
        SYNTAX     INTEGER {
              up(1),          -- LECS is accepting
                              -- CONFIGURE request
              down(2) }       -- LECS is not accepting
                              -- CONFIGURE request
        MAX-ACCESS read-create
        STATUS     current
        DESCRIPTION
                 "The desired  state of the  LECS on
                 this interface as prescribed by the
                 operator.   The actions of the agent
                 will,  if  at all possible,  eventually
                 result in the desired  state being
                 reflected in the lecsOperStatus."
        DEFVAL  { up }
        ::= { lecsConfEntry 9 }

    lecsRowStatus OBJECT-TYPE
        SYNTAX     RowStatus
        MAX-ACCESS read-create
        STATUS     current
        DESCRIPTION
             "This object is used to create or
```

```
                delete entries in this table."
           REFERENCE "RFC 1443, [10] Textual Conventions
                for version 2 of the Simple Network Management
                Protocol (SNMPv2)."
           ::= { lecsConfEntry 10 }

--
-- LECS to ELAN mapping table
--

lecsElanTable OBJECT-TYPE
   SYNTAX     SEQUENCE OF LecsElanEntry
   MAX-ACCESS not-accessible
   STATUS     current
   DESCRIPTION
        "This table contains the mapping between
        ELANs and LECS.

        When a LECS is deleted from the lecsConfTable,
        all entries associated with this entry will
        also be deleted."
   ::= { elanLecsConfGroup 3 }

lecsElanEntry OBJECT-TYPE
   SYNTAX     LecsElanEntry
   MAX-ACCESS not-accessible
   STATUS     current
   DESCRIPTION
             "Each entry represents an ELAN to LECS
             mapping."
   INDEX { elanConfIndex, lecsConfIndex }
   ::= { lecsElanTable 1 }

 LecsElanEntry ::= SEQUENCE {
    lecsElanRowStatus  RowStatus
}

 lecsElanRowStatus  OBJECT-TYPE
     SYNTAX         RowStatus
     MAX-ACCESS     read-create
     STATUS         current
     DESCRIPTION
        "This object is used to create or
        delete an entry from this table."
     REFERENCE "RFC 1443, [10] Textual Conventions
        for version 2 of the Simple Network Management
        Protocol (SNMPv2)."
     ::= { lecsElanEntry 1 }

--
 --      (c) TLV (Type, Length and Value) table
--

 lecsTlvTable      OBJECT-TYPE
     SYNTAX      SEQUENCE OF LecsTlvEntry
     MAX-ACCESS  not-accessible
     STATUS      current
     DESCRIPTION
         "This table contains currently configured
         sets of TLVs in this LECS. These sets can
         include not only the standard TLVs specified
         in the LAN Emulation Spec. 1.0 [1] but
         also the additional parameters
         exchanged between the LECS and LEC.
         This table is indexed by a selector
         index, which allows more than one
         TLV to be selected by an ELAN; and
         the TLV tag, which specified the type
         of the TLV; and a TLV index which
         is used to distinguish between different
         entries with the same TLV tag.
         How does the LECS treat the TLVs
```

```
        that are not specified in this table
        in the CONFIGURE requests are not
        defined in this MIB."
::= { elanLecsConfGroup 4 }

lecsTlvEntry        OBJECT-TYPE
    SYNTAX          LecsTlvEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "Each entry represents a set of TLV
        encodings for an ELAN represented
        by the elanConfIndex.  Objects lecsTlvVal
        and lecsTlvRowStatus are required during
        row creation."
INDEX { lecsTlvSelectorIndex,
        lecsTlvTag, lecsTlvIndex }
::= { lecsTlvTable 1 }

   LecsTlvEntry ::=
       SEQUENCE {
            lecsTlvSelectorIndex
                TlvSelectorIndexType,
            lecsTlvTag
                OCTET STRING,
            lecsTlvIndex
                Integer32,
            lecsTlvVal
                OCTET STRING,
            lecsTlvRowStatus
                RowStatus
       }

   lecsTlvSelectorIndex      OBJECT-TYPE
       SYNTAX          TlvSelectorIndexType
       MAX-ACCESS      not-accessible
       STATUS          current
       DESCRIPTION
            "The value of this object indicates
            a group of TLVs that can be selected."
       ::= { lecsTlvEntry 1 }

   lecsTlvTag          OBJECT-TYPE
       SYNTAX          OCTET STRING (SIZE(4))
       MAX-ACCESS      not-accessible
       STATUS          current
       DESCRIPTION
            "The value of this object represents the
            type of the contents of the lecsTlvVal
            field in the entry.  The OUI field
            in this object occupies the first
            ( or most significant )3 octets.
            The OUI value 00-A0-3E is used for
            the standard values defined by the ATM
            Forum specification."
       ::= { lecsTlvEntry 2 }

   lecsTlvIndex        OBJECT-TYPE
       SYNTAX          Integer32 ( 1..2147483647 )
       MAX-ACCESS      not-accessible
       STATUS          current
       DESCRIPTION
            "The index of this TLV encoding entry.
             The value of this object can be used
             to distinguish between different entries
             with the same lecsTlvTag value."
       ::= { lecsTlvEntry 3 }

   lecsTlvVal          OBJECT-TYPE
       SYNTAX          OCTET STRING (SIZE(0..1024))
       MAX-ACCESS      read-create
       STATUS          current
```

```
        DESCRIPTION
             "The value of this TLV entry.  Note that
              the length is implied by the length of
              the OCTET STRING."
        ::= { lecsTlvEntry 4 }

    lecsTlvRowStatus  OBJECT-TYPE
        SYNTAX        RowStatus
        MAX-ACCESS    read-create
        STATUS        current
        DESCRIPTION
             "This object is used to create or
             destroy entries in this table. "
         REFERENCE "RFC 1443, [10] Textual Conventions
              for version 2 of the Simple Network Management
              Protocol (SNMPv2)."
        ::= { lecsTlvEntry 5 }

--
--      (d) LECS Config VCC table
--

lecsVccTable      OBJECT-TYPE
     SYNTAX      SEQUENCE OF LecsVccEntry
     MAX-ACCESS  not-accessible
     STATUS      current
     DESCRIPTION
         "This table contains all the CONFIG VCCs of the
          LECS.  The CONFIG VCC is used by the LEC to send/
          receive ATM LE CONFIGURE request/response to/from
          the LECS.  This table is writable if PVC is used
          and read only if SVC is used. "
::= { elanLecsConfGroup 5 }

lecsVccEntry       OBJECT-TYPE
     SYNTAX          LecsVccEntry
     MAX-ACCESS      not-accessible
     STATUS          current
     DESCRIPTION
         "Each entry represents a CONFIG VCC
          between a pair of LEC and LECS."
 INDEX { lecsConfIndex, lecsVccIfIndex,
         lecsVccVpi, lecsVccVci }
::= { lecsVccTable 1 }

   LecsVccEntry ::=
       SEQUENCE {
           lecsVccIfIndex
               IfIndexOrZero,
           lecsVccVpi
               VpiInteger,
           lecsVccVci
               VciInteger,
           lecsVccRowStatus
               RowStatus
       }

   lecsVccIfIndex       OBJECT-TYPE
        SYNTAX        IfIndexOrZero
        MAX-ACCESS    not-accessible
        STATUS        current
        DESCRIPTION
             "The ATM interface which the CONFIG VCC is established.
              This value must be an existing value in the
              ifTable.  The value of this object is set to zero
              when the ATM interface is an internal connection. "
        ::= { lecsVccEntry 1 }

   lecsVccVpi         OBJECT-TYPE
        SYNTAX        VpiInteger
        MAX-ACCESS    not-accessible
        STATUS        current
```

```
        DESCRIPTION
            "The VPI value of the CONFIG VCC.  The object
             lecsVccIfIndex, lecsVccVci and this object
             uniquely identifies a VCC within an ATM system."
        ::= { lecsVccEntry 2 }

    lecsVccVci      OBJECT-TYPE
        SYNTAX        VciInteger
        MAX-ACCESS    not-accessible
        STATUS        current
        DESCRIPTION
            "The VCI value of the CONFIG VCC. The object
             lecsVccIfIndex, lecsVccVpi and this object
             uniquely identifies a VCC within an ATM system "
        ::= { lecsVccEntry 3 }

    lecsVccRowStatus  OBJECT-TYPE
        SYNTAX        RowStatus
        MAX-ACCESS    read-create
        STATUS        current
        DESCRIPTION
            "This object is used to create or
            destroy entries in this table. "
        REFERENCE "RFC 1443, [10] Textual Conventions
             for version 2 of the Simple Network Management
             Protocol (SNMPv2)."
        ::= { lecsVccEntry 4 }

--
--  (2.2) LECS Statistics Group
--     (a) LECS Statstistics table
--

    lecsStatsTable OBJECT-TYPE
        SYNTAX      SEQUENCE OF LecsStatsEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
                " A (conceptual) table of statistics
                associated with all
                LECS instances on the device."
        ::= { elanLecsStatGroup 1 }

    lecsStatsEntry OBJECT-TYPE
        SYNTAX     LecsStatsEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
                " A (conceptual) row in the
                lecsStatsTable which corresponds
                to the statistics kept by a particular
                instance of a LECS."
        AUGMENTS { lecsConfEntry }
        ::= { lecsStatsTable 1 }

    LecsStatsEntry ::= SEQUENCE {
        lecsStatSuccessful
            Counter32,
        lecsStatInBadFrames
            Counter32,
        lecsStatInvalidParam
            Counter32,
        lecsStatInsufRes
            Counter32,
        lecsStatAccDenied
            Counter32,
        lecsStatInvalidReq
            Counter32,
        lecsStatInvalidDest
            Counter32,
        lecsStatInvalidAddr
            Counter32,
```

```
    lecsStatNoConf
        Counter32,
    lecsStatConfError
        Counter32,
    lecsStatInsufInfo
        Counter32

}

lecsStatSuccessful  OBJECT-TYPE
     SYNTAX          Counter32
     MAX-ACCESS      read-only
     STATUS          current
     DESCRIPTION
          "The number of CONFIGURE requests
          successfully granted since the agent
          was last initialized."
     ::= { lecsStatsEntry 1 }

lecsStatInBadFrames  OBJECT-TYPE
     SYNTAX          Counter32
     MAX-ACCESS      read-only
     STATUS          current
     DESCRIPTION
          "The number of mal formed CONFIGURE
          requests dropped by the LECS."
     ::= { lecsStatsEntry 2 }

 lecsStatInvalidParam   OBJECT-TYPE
     SYNTAX         Counter32
     MAX-ACCESS     read-only
     STATUS         current
     DESCRIPTION
          "The number of CONFIGURE requests
          rejected due to the invalid request
          parameters error."
     REFERENCE
          "LAN Emulation Over ATM Spec. v1.0. Table 13"
     ::= { lecsStatsEntry 3 }

 lecsStatInsufRes    OBJECT-TYPE
     SYNTAX         Counter32
     MAX-ACCESS     read-only
     STATUS         current
     DESCRIPTION
          "The number of CONFIGURE requests rejected due
          to the insufficent resources to grant request
          error."
     REFERENCE
          "LAN Emulation Over ATM Spec. v1.0. Table 13"
     ::= { lecsStatsEntry 4 }

 lecsStatAccDenied   OBJECT-TYPE
     SYNTAX         Counter32
     MAX-ACCESS     read-only
     STATUS         current
     DESCRIPTION
          "The number of CONFIGURE requests rejected due
          to the access denied error."
     REFERENCE
          "LAN Emulation Over ATM Spec. v1.0. Table 13"
     ::= { lecsStatsEntry 5 }

 lecsStatInvalidReq  OBJECT-TYPE
     SYNTAX         Counter32
     MAX-ACCESS     read-only
     STATUS         current
     DESCRIPTION
          "The number of CONFIGURE requests rejected due
          to the invalid requester-id error."
     REFERENCE
          "LAN Emulation Over ATM Spec. v1.0. Table 13"
```

```
            ::= { lecsStatsEntry 6 }

        lecsStatInvalidDest OBJECT-TYPE
            SYNTAX         Counter32
            MAX-ACCESS     read-only
            STATUS         current
            DESCRIPTION
                 "The number of CONFIGURE requests rejected due
                 to the invalid destination error."
            REFERENCE
                 "LAN Emulation Over ATM Spec. v1.0. Table 13"
            ::= { lecsStatsEntry 7 }

        lecsStatInvalidAddr OBJECT-TYPE
            SYNTAX         Counter32
            MAX-ACCESS     read-only
            STATUS         current
            DESCRIPTION
                 "The number of CONFIGURE requests rejected due
                 to the invalid ATM address error."
            REFERENCE
                 "LAN Emulation Over ATM Spec. v1.0. Table 13"
            ::= { lecsStatsEntry 8 }

        lecsStatNoConf   OBJECT-TYPE
            SYNTAX         Counter32
            MAX-ACCESS     read-only
            STATUS         current
            DESCRIPTION
                 "The number of CONFIGURE requests rejected due
                 to the LE Client is not recognized error."
            REFERENCE
                 "LAN Emulation Over ATM Spec. v1.0. Table 13"
            ::= { lecsStatsEntry 9 }

        lecsStatConfError   OBJECT-TYPE
            SYNTAX         Counter32
            MAX-ACCESS     read-only
            STATUS         current
            DESCRIPTION
                 "The number of CONFIGURE requests rejected due
                 to the LE_CONFIGURE error."
            REFERENCE
                 "LAN Emulation Over ATM Spec. v1.0. Table 13"
            ::= { lecsStatsEntry 10 }

        lecsStatInsufInfo   OBJECT-TYPE
            SYNTAX         Counter32
            MAX-ACCESS     read-only
            STATUS         current
            DESCRIPTION
                 "The number of CONFIGURE requests rejected due
                 to the insufficient information error."
            REFERENCE
                 "LAN Emulation Over ATM Spec. v1.0. Table 13"
            ::= { lecsStatsEntry 11 }

--
--  (2.3) LECS Fault Management Group - optional
--       (a) LECS Error log control table
--       (b) LECS error log table
--

       lecsErrCtlTable OBJECT-TYPE
            SYNTAX      SEQUENCE OF LecsErrCtlEntry
            MAX-ACCESS not-accessible
            STATUS     current
            DESCRIPTION
                   "This table contains error log control
                    information of all LECS instances.
                    This table is an extention to the
                    lecsConfTable.  It is used to enable
```

```
                or disable error logs for a particular
                LECS entry."
         ::= { elanLecsFaultGroup 1 }

    lecsErrCtlEntry OBJECT-TYPE
         SYNTAX     LecsErrCtlEntry
         MAX-ACCESS not-accessible
         STATUS     current
         DESCRIPTION
               "Each entry represents a LECS entry
               in the lecsConfTable."
         AUGMENTS { lecsConfEntry }
         ::= { lecsErrCtlTable 1 }

     LecsErrCtlEntry ::=
         SEQUENCE {
             lecsErrCtlAdminStatus
                 INTEGER,
             lecsErrCtlOperStatus
                 INTEGER,
             lecsErrCtlClearLog
                 INTEGER,
             lecsErrCtlMaxEntries
                 INTEGER,
             lecsErrCtlLastEntry
                 LecsErrLogIndexType
             }

     lecsErrCtlAdminStatus     OBJECT-TYPE
             SYNTAX        INTEGER {
                 enable(1),        -- enable error log
                 disable(2)        -- disable error log
             }
     MAX-ACCESS    read-create
     STATUS        current
     DESCRIPTION
          "This object is used to enable/disable error
               logging for the LECS."
     ::= { lecsErrCtlEntry 1 }

     lecsErrCtlOperStatus             OBJECT-TYPE
         SYNTAX        INTEGER {
              other(1),        -- not specified
              active(2),        -- error logging
              outOfRes(3),        -- Out of buffer error
              failed(4),        -- failed to start
                              -- error log for reasons
                              -- other than out of
                              -- resources
              disabled(5)      -- the error logging capability
                              -- is disabled
         }
         MAX-ACCESS    read-only
         STATUS        current
         DESCRIPTION
            "This object is used to indicate
            the result of a set operation to the
            object lecsErrCtlAdminStatus.
            If the error log was successfully
            started, it is in active(2) mode.
            Otherwise, it is set to either
            outOfRes(3) or failed(4) for
            the respective reasons."
         ::= { lecsErrCtlEntry 2 }

     lecsErrCtlClearLog             OBJECT-TYPE
         SYNTAX        INTEGER {
            noOp(1),        -- read only value
            clear(2)        -- clear the error log associated
                          -- with this LECS entry
         }
         MAX-ACCESS        read-create
```

```
            STATUS                 current
            DESCRIPTION
                "This object is used to clear the error log
                 entries associated with this LECS."
            ::= { lecsErrCtlEntry 3 }

        lecsErrCtlMaxEntries        OBJECT-TYPE
            SYNTAX        INTEGER ( 1..65535 )
            MAX-ACCESS    read-only
            STATUS        current
            DESCRIPTION
               "The maximum entries of the error
               log a LECS can support."
            ::= { lecsErrCtlEntry 4 }

        lecsErrCtlLastEntry        OBJECT-TYPE
            SYNTAX        LecsErrLogIndexType
            MAX-ACCESS    read-create
            STATUS        current
            DESCRIPTION
                "The index to the last entry in the error
                log table associated with this LECS."
            ::= { lecsErrCtlEntry 5 }

--
--
--   LECS error log table
--

lecsErrLogTable OBJECT-TYPE
    SYNTAX     SEQUENCE OF LecsErrLogEntry
    MAX-ACCESS not-accessible
    STATUS     current
    DESCRIPTION
        "This table contains error logs
        of the LECS instances enabled
        in the lecsErrCtlTable.  This table
        is indexed by the LECS instance
        index and an arbitrary integer
        uniquely identifies an error
        log."
    ::= { elanLecsFaultGroup 2 }

lecsErrLogEntry OBJECT-TYPE
    SYNTAX     LecsErrLogEntry
    MAX-ACCESS not-accessible
    STATUS     current
    DESCRIPTION
        "Each entry represents a LEC that
         was rejected due to a violation
         against the policies or an error."
    INDEX { lecsConfIndex, lecsErrLogIndex }
    ::= { lecsErrLogTable 1 }

LecsErrLogEntry ::=
    SEQUENCE {
        lecsErrLogIndex
            LecsErrLogIndexType,
        lecsErrLogAtmAddr
            AtmLaneAddress,
        lecsErrLogErrCode
            INTEGER,
        lecsErrLogTime
            TimeStamp
    }

lecsErrLogIndex      OBJECT-TYPE
    SYNTAX          LecsErrLogIndexType
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "An arbitrary integer which uniquely
```

```
                      identifies an error log entry. The first
                      entry after reset or clearing the error
                      log is an assigned value (2^32-1).  Succeding
                      entries are assigned with descending values
                      consecutively.  Entries after 1 are discarded.The
enabling/disabling of
                      the error log capability is done in
                      the lecsErrCtlTable."
                 ::= { lecsErrLogEntry 1 }

             lecsErrLogAtmAddr OBJECT-TYPE
                 SYNTAX        AtmLaneAddress
                 MAX-ACCESS    read-only
                 STATUS        current
                 DESCRIPTION
                      "The ATM address of the requester
                      which sends the CONFIGURE request
                      and causes the error to occur.
                      The corresponding error code is
                      specified in the object lecsErrLogErrCode."
                 ::= { lecsErrLogEntry 2 }

             lecsErrLogErrCode OBJECT-TYPE
                 SYNTAX        INTEGER (0..22)
                 MAX-ACCESS    read-only
                 STATUS        current
                 DESCRIPTION
                      "The Error code which indicates the
                      cause of the error triggered by the
                      CONFIGURE request sent by the
                      requester indicated by the value of
                      the object lecsErrLogAtmAddr."
                 ::= { lecsErrLogEntry 3 }

             lecsErrLogTime    OBJECT-TYPE
                 SYNTAX        TimeStamp
                 MAX-ACCESS    read-only
                 STATUS        current
                 DESCRIPTION
                      "The sysUpTime when this entry was logged by
                      the LECS."
                 ::= { lecsErrLogEntry 4 }



         -- Conformance Information

         elanMIBConformance       OBJECT IDENTIFIER ::= { elanMIB 4 }

         elanMIBGroups            OBJECT IDENTIFIER ::=
                                  { elanMIBConformance 1 }

         elanMIBCompliances       OBJECT IDENTIFIER ::=
                                  { elanMIBConformance 2 }

         -- Compliance Statements

         elanMIBCompliance         MODULE-COMPLIANCE
                 STATUS            current
                 DESCRIPTION       "The compliance statement for SNMP
                      entities which support ATM LAN Emulation
                      ELAN MIB."


                 MODULE -- this module
                        MANDATORY-GROUPS  { elanCConfGroup }

                        GROUP   elanLecAssignByAtmGroup
                        DESCRIPTION
                             "This group is mandatory only for thoes
                             agent that implements the LEC assignment
```

```
                policy 'by ATM address'."

            GROUP   elanLecAssignByMacGroup
            DESCRIPTION
                "This group is mandatory only for thoes
                agent that implements the LEC assignment
                policy 'by MAC address'."

            GROUP   elanLecAssignByRdGroup
            DESCRIPTION
                "This group is mandatory only for thoes
                agent that implements the LEC assignment
                policy 'by Route Descriptor'."

            GROUP   lecsCStatGroup
            DESCRIPTION
              "This group is mandatory only for those
              agent that supports LECS."

            GROUP   lecsCGroup
            DESCRIPTION
              "This group is mandatory only for those
              agent that supports LECS."

            GROUP   lecsCFaultGroup
            DESCRIPTION
              "This group is mandatory only to those agents
              that support LECS and Fault Management."

      ::= { elanMIBCompliances 1 }

      -- Units of Conformance

      elanCConfGroup    OBJECT-GROUP
            OBJECTS {
                      elanConfNextId,
                      elanConfName,
                      elanConfTlvIndex,
                      elanConfLanType,
                      elanConfMaxFrameSize,
                      elanConfRowStatus,
                      elanLecPktSizeRowStatus,
                      elanLecNameRowStatus,
                      elanLesAtmAddress,
                      elanLesRowStatus,
                      elanPolicyPriority,
                      elanPolicyType,
                      elanPolicyRowStatus
                 }
            STATUS     current
            DESCRIPTION
              "A collection of objects providing configuration
               information about an Emulated LAN."
            ::= { elanMIBGroups 1 }

     elanLecAssignByAtmGroup  OBJECT-GROUP
           OBJECTS {
               elanLecAtmRowStatus
           }
           STATUS    current
           DESCRIPTION
               "A collection of objects providing ATM addresses
              of LECs."
           ::= { elanMIBGroups 2 }


     elanLecAssignByMacGroup OBJECT-GROUP
           OBJECTS {
                     elanLecMacRowStatus
           }
           STATUS    current
           DESCRIPTION
```

```
            "A collection of objects providing MAC addresses
           of LECs."
        ::= { elanMIBGroups 3 }


    elanLecAssignByRdGroup OBJECT-GROUP
        OBJECTS {
                elanLecRdRowStatus
        }
        STATUS    current
        DESCRIPTION
            "A collection of objects providing Route Descriptors
           of LECs."
        ::= { elanMIBGroups 4 }

    lecsCStatGroup     OBJECT-GROUP
        OBJECTS {
                         lecsStatSuccessful,
        lecsStatInBadFrames,
                         lecsStatInvalidParam,
                         lecsStatInsufRes,
                         lecsStatAccDenied,
                         lecsStatInvalidReq,
                         lecsStatInvalidDest,
                         lecsStatInvalidAddr,
                         lecsStatNoConf,
                         lecsStatConfError,
                         lecsStatInsufInfo
        }
        STATUS     current
        DESCRIPTION
            "A collection of objects providing information
           about LECS statistics."
        ::= { elanMIBGroups 5 }

    lecsCGroup     OBJECT-GROUP
        OBJECTS {
             lecsConfNextId,
             lecsAtmIfIndex,
             lecsAtmAddrSpec,
             lecsAtmAddrMask,
             lecsAtmAddrActual,
             lecsElanRowStatus,
             lecsPolicySelIndex,
             lecsTlvVal,
             lecsTlvRowStatus,
             lecsVccRowStatus,
             lecsLastInitialized,
             lecsOperStatus,
             lecsAdminStatus,
             lecsRowStatus }
        STATUS     current
        DESCRIPTION
             "A group of objects used for LECS
             management only."
        ::= { elanMIBGroups 6 }

    lecsCFaultGroup  OBJECT-GROUP
        OBJECTS {
                lecsErrCtlAdminStatus,
                lecsErrCtlOperStatus,
                lecsErrCtlClearLog,
                lecsErrCtlMaxEntries,
                lecsErrCtlLastEntry,
                lecsErrLogTime,
                lecsErrLogAtmAddr,
                lecsErrLogErrCode
        }
        STATUS     current
        DESCRIPTION
             "A group of objects used for LECS
              fault management only."
```

```
                ::= { elanMIBGroups 7 }

        END
```

```
LAN-EMULATION-LES-MIB DEFINITIONS ::= BEGIN

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

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

    MODULE-COMPLIANCE, OBJECT-GROUP
        FROM SNMPv2-CONF

    LeArpTableEntryType,
    AtmLaneAddress,
    VpiInteger,
    VciInteger,
    atmfLanEmulation
        FROM LAN-EMULATION-CLIENT-MIB

    IfIndexOrZero, AtmLaneMask
        FROM LAN-EMULATION-ELAN-MIB;

lesMIB MODULE-IDENTITY
    LAST-UPDATED "9602121200Z"
    ORGANIZATION "ATM Forum LAN Emulation Sub-Working Group"
    CONTACT-INFO
        "
          The ATM Forum

          2570 West EL camino Real,
          Ste 304, Mountain View,
          CA 94040-1313, USA
          Tel: 415-578-6860
          E-mail: info@atmforum.com"

    DESCRIPTION
        "This is the MIB module for managing ATM
        LAN Emulation Servers."
    ::= { atmfLanEmulation 3 }


--
--    Textual Conventions
--

LecId ::= TEXTUAL-CONVENTION
    STATUS      current
    DESCRIPTION
            "C14 LE Client Identifier."
    SYNTAX       INTEGER( 0..65279 )

BusConfIndex ::= TEXTUAL-CONVENTION
    STATUS      current
    DESCRIPTION
        "A value which uniquely identifies a BUS.
        This number is only used locally by the agent to
        distinguish between BUSs."
    SYNTAX       Integer32

LesLocalIndex ::= TEXTUAL-CONVENTION
    STATUS      current
    DESCRIPTION
        "A value which uniquely identifies a
        conceptual row in the lesConfTable.
        This number is only used locally by the agent to
        distinguish between LESs."
```

```
        SYNTAX     Integer32

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

               S2  LAN Type.  The type of LAN that the LE Server is
servicing."
        REFERENCE
              "ATM Forum LAN Emulation Over ATM Specification, V1.0,
              Section 5.1.2."
        SYNTAX      INTEGER {
                        aflane8023(2),
                        aflane8025(3)
                       }

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

           S3  Maximum Data Frame Size.  The maximum AAL-5 SDU size.
"
        REFERENCE
              "ATM Forum LAN Emulation Over ATM Specification, V1.0,
              Section 5.1.2."
        SYNTAX      INTEGER {
                        max1516(2),
                        max4544(3),
                        max9234(4),
                        max18190(5)
                       }

    LesErrLogIndexType ::= TEXTUAL-CONVENTION
        STATUS     current
        DESCRIPTION
                "An arbitrary integer used in identifying
                an erro log associated with a LES."
         SYNTAX     Integer32(1..2147483647)

    -- This MIB module consists of the following groups:
    --
    --  (1) LES Conf Group
    --      (a) LES table
    --      (b) LES VCC table
    --      (c) BUS table
    --      (d) ATM ARP table
    --      (e) ATM Route Descriptor ARP table
    --      (f) LES-LEC table
    --
    --  (2) LES Monitoring Group
    --      (a) LES Statistics Table
    --
    --  (3) LES-LEC Monitoring Group
    --      (a) LES-LEC statistics table
    --  (4) LES Fault Management Group
    --      (a) LES Error Log Control Table
    --      (b) LES Error Log Table
    --

    lesConfGroup  OBJECT IDENTIFIER ::= { lesMIB 1 }

    lesStatGroup  OBJECT IDENTIFIER ::= { lesMIB 2 }

    lesLecStatGroup OBJECT IDENTIFIER ::= { lesMIB 3 }

    lesFaultGroup OBJECT IDENTIFIER ::= { lesMIB 4 }

    lesConfNextId  OBJECT-TYPE
        SYNTAX    LesLocalIndex
        MAX-ACCESS read-only
```

```
    STATUS    current
    DESCRIPTION
        "The next available LES index.  The
        value of this object can be used as
        the index by the network manager to
        create an entry in the lesConfTable."
    ::= { lesConfGroup 1 }

--
--  LES table
--

lesConfTable    OBJECT-TYPE
    SYNTAX      SEQUENCE OF LesConfEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "This table contains all LAN Emulation
        Servers this agent manages.  The LES
        is one of the components in the Emulated
        LAN which implements the control coordination function.
        It is the address resolution server for a given ELAN.
        The LES provides a facility for registering and resolving
        MAC addresses and/or route descriptors to ATM
        addresses.  There can be multiple LES per ELAN
        but a LES can serve only one ELAN."
    ::= { lesConfGroup 2 }

lesConfEntry     OBJECT-TYPE
    SYNTAX       LesConfEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
        "Each entry in this table represents a
        LES.  The parameters in each entry apply
        to one emulated LAN served by one LES.
        Objects lesLanType and lesMaxFrameSize
      are also required besides lesRowStatus
      during row creation."
    INDEX { lesConfIndex }
    ::= { lesConfTable 1 }

LesConfEntry ::=
    SEQUENCE {
        lesConfIndex
            LesLocalIndex,
        lesAtmAddrSpec
            AtmLaneAddress,
        lesAtmAddrMask
            AtmLaneMask,
        lesAtmAddrActual
            AtmLaneAddress,
        lesElanName
            DisplayString,
        lesLanType
            LesLecDataFrameFormat,
        lesLastChange
            TimeStamp,
        lesMaxFrameSize
            LesLecDataFrameSize,
        lesControlTimeOut
            INTEGER,
        lesOperStatus
            INTEGER,
        lesAdminStatus
            INTEGER,
        lesRowStatus
            RowStatus
        }

lesConfIndex       OBJECT-TYPE
    SYNTAX         LesLocalIndex
```

```
     MAX-ACCESS    not-accessible
     STATUS        current
     DESCRIPTION
         "A value which uniquely identifies a
         conceptual row in the lesConfTable.

         If the conceptual row identified by this value
         of lesConfIndex is recreated following an agent
         restart, the same value of lesConfIndex must be
         used to identify the recreated row."
    ::= { lesConfEntry 1 }

lesAtmAddrSpec    OBJECT-TYPE
    SYNTAX          AtmLaneAddress
    MAX-ACCESS      read-create
    STATUS          current
    DESCRIPTION
         "An ATM  address specified by  the
         network or local management that,
         with the ATM address mask, determines
         a  portion of  the  ATM address  that
         the  LES on the designated ATM interface
         will use  to derive the actual ATM
         address from the network or ILMI.  The derived
         ATM address is specified in the object
         lesAtmAddrActual, which is used to
         receive ATM ARP requests."
    REFERENCE
          "LAN Emulation Over ATM Specification -
         version 1.0. S1."
    ::= { lesConfEntry 2 }

lesAtmAddrMask OBJECT-TYPE
    SYNTAX     AtmLaneMask
    MAX-ACCESS read-create
    STATUS     current
    DESCRIPTION
         "The ATM address mask associated with
         the object lesAtmAddrSpec.  The value
         of the mask is an ATM address with
         the don't care portion set to zero
         and the valid ATM address portion set to one."
    REFERENCE
          "LAN Emulation Over ATM Specification -
         version 1.0. S1."
     DEFVAL { 'FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF'H }
    ::= { lesConfEntry 3 }

lesAtmAddrActual OBJECT-TYPE
    SYNTAX     AtmLaneAddress
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
           " The resultant ATM address  in use by  the LES.  This
           object is a product of the specified ATM address, mask
           and  interaction  with  the network.  This object is
           created by the agent."
    REFERENCE
          "LAN Emulation Over ATM Specification -
         version 1.0. S1."
    ::= { lesConfEntry 4 }

lesElanName    OBJECT-TYPE
    SYNTAX          DisplayString (SIZE (0..32))
    MAX-ACCESS      read-create
    STATUS          current
    DESCRIPTION
         "The name of the Emulated LAN this LES
         is providing service for.  This object
         may be used to identify the ELAN the LES
         is in."
    REFERENCE
```

```
          "LAN Emulation Over ATM Specification -
          version 1.0. C5."
      DEFVAL { "" }
      ::= { lesConfEntry 5 }

  lesLanType          OBJECT-TYPE
      SYNTAX          LesLecDataFrameFormat
      MAX-ACCESS      read-create
      STATUS          current
      DESCRIPTION
          "The type of the ATM Emulated LAN this
          LES is providing service to."
      REFERENCE
          "LAN Emulation Over ATM Specification -
          version 1.0. S2."
      ::= { lesConfEntry 6 }

   lesLastChange OBJECT-TYPE
      SYNTAX     TimeStamp
      MAX-ACCESS read-only
      STATUS     current
      DESCRIPTION
            "The  value of   sysUpTime  when  this LES
            has entered the state indicated by the
            object lesOperStatus."
      ::= { lesConfEntry 7 }

  lesMaxFrameSize     OBJECT-TYPE
      SYNTAX          LesLecDataFrameSize
      MAX-ACCESS      read-create
      STATUS          current
      DESCRIPTION
          "The maximum AAL-5 SDU size of a data
          frame that the LE service can guarantee
          not to drop because it is too large.
"
      REFERENCE
          "LAN Emulation Over ATM Specification -
          version 1.0. S3."
      ::= { lesConfEntry 8 }

  lesControlTimeOut   OBJECT-TYPE
      SYNTAX          INTEGER (10..300)
      UNITS           "seconds"
      MAX-ACCESS      read-create
      STATUS          current
      DESCRIPTION
          "Time out period used for timing out most
          request/response control frame interactions.
          This is the time a Client has to issue a join
          request to a LES after a control direct VCC is
          established with a LES."
      REFERENCE
          "LAN Emulation Over ATM Specification -
          version 1.0. S4."
      DEFVAL     { 120 }
      ::= { lesConfEntry 9 }

  lesOperStatus   OBJECT-TYPE
      SYNTAX      INTEGER {
          other(1),       -- unknown state
          up(2),          -- LES is up and running
          down(3)         -- LES is up down or not available
      }
      MAX-ACCESS  read-only
      STATUS      current
      DESCRIPTION
          "The operational state of this LES entry.
          When in 'up' state the LES will respond
          to LEC requests.  Any other state the
          LES is notavailable for service and may
          release all the existing VCCs and refuse
```

```
        service to all clients."
   ::= { lesConfEntry 11 }

lesAdminStatus OBJECT-TYPE
   SYNTAX     INTEGER {
       up(2),          -- LES is up and running
       down(3)         -- LES is up down or not available
   }
   MAX-ACCESS read-create
   STATUS     current
   DESCRIPTION
        "  The  desired  state   of  the  designated   LES  as
        prescribed by the operator. The  actions of the  agent
        will, if at  all  possible, eventually  result  in the
        desired state being  reflected in  the lesOperStatus."
   DEFVAL { up }
   ::= { lesConfEntry 12 }

lesRowStatus OBJECT-TYPE
   SYNTAX     RowStatus
   MAX-ACCESS read-create
   STATUS     current
   DESCRIPTION
        "This object is used to create or
        delete entries in the elanConfTable."
   REFERENCE "RFC 1443, [10] Textual Conventions
        for version 2 of the Simple Network Management
        Protocol (SNMPv2)."
   ::= { lesConfEntry 13 }

--
--  LES VCC table
--

lesVccTable      OBJECT-TYPE
    SYNTAX       SEQUENCE OF LesVccEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
        "This table contains all the Control Distribute
        VCCs used by the LES to distribute control traffic
        to the participating LECs.  The Control Distribute
        VCC can either be point-to-point or point-to-
        multipoint calls.  This table is read only if
        SVCs are used and writable if PVCs are used."
    ::= { lesConfGroup 3 }

lesVccEntry       OBJECT-TYPE
    SYNTAX        LesVccEntry
    MAX-ACCESS    not-accessible
    STATUS        current
    DESCRIPTION
        "Each entry in this table represents a
        LES entry and it's associated Control Distribute VCC. "
    INDEX { lesConfIndex, lesVccAtmIfIndex,
            lesVccCtlDistVpi, lesVccCtlDistVci }
    ::= { lesVccTable 1 }

LesVccEntry ::=
    SEQUENCE {
        lesVccAtmIfIndex
            IfIndexOrZero,
        lesVccCtlDistVpi
            VpiInteger,
        lesVccCtlDistVci
            VciInteger,
        lesVccRowStatus
            RowStatus
    }

lesVccAtmIfIndex  OBJECT-TYPE
      SYNTAX      IfIndexOrZero
```

```
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
           "The ATM interface which the Control
           Distribute VCC is running on.
           This value must match an existing
           value  in the   ifTable.
           The value of this object is set
           to zero when the ATM interface is
           an internal connection."
        ::= { lesVccEntry 1 }

lesVccCtlDistVpi  OBJECT-TYPE
        SYNTAX     VpiInteger
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
           "The VPI value of the Control
           Distribute VCC.  The object
           lesVccAtmIfIndex, lesVccCtlDistVci
           and the value of this object uniquely
           identfies a VCC within a ATM host."
        ::= { lesVccEntry 2 }

lesVccCtlDistVci  OBJECT-TYPE
        SYNTAX     VciInteger
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
           "The VCI value of the Control
           Distribute VCC.  The object
           lesVccAtmIfIndex, lesVccCtlDistVci
           and the value of this object uniquely
           identfies a VCC within a ATM host."
        ::= { lesVccEntry 3 }

lesVccRowStatus   OBJECT-TYPE
        SYNTAX     RowStatus
        MAX-ACCESS read-create
        STATUS     current
        DESCRIPTION
             "This object is used to create or
             delete entries in the elanConfTable."
        REFERENCE "RFC 1443, [10] Textual Conventions
             for version 2 of the Simple Network Management
             Protocol (SNMPv2)."
        ::= { lesVccEntry 4 }



--
--  BUS table
--

lesBusTable      OBJECT-TYPE
    SYNTAX       SEQUENCE OF LesBusEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
        "This table contains the BUSs paired
        with the LESs found in the lesConfTable.
        The BUS (Broadcast and Unknown Server)
        handles data sent by a LE client to
        the broadcast MAC address, all multicast
        traffic, and initial unicast frames
        which are sent by a LAN Emulation Client
        before the data direct target ATM address
        has been resolved."
    ::= { lesConfGroup 4 }

lesBusEntry       OBJECT-TYPE
    SYNTAX        LesBusEntry
```

```
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
      "Each entry in this table represents aLES/BUS
      pair.  Object  lesBusAddress  is  required
      duringrow creation. This table is indexed by
      lesConfIndex  and lesBusConfIndexto  show  the
      pairing relationship betweenthe LES and BUS."
     INDEX { lesConfIndex, lesBusConfIndex }
    ::= { lesBusTable 1 }

LesBusEntry ::=
    SEQUENCE {
        lesBusConfIndex
            BusConfIndex,
        lesBusAddress
            AtmLaneAddress
        }

    lesBusConfIndex      OBJECT-TYPE
        SYNTAX           BusConfIndex
        MAX-ACCESS       not-accessible
        STATUS           current
        DESCRIPTION
            "A value which uniquely identifies a
            BUS ATM address.

            If the BUS ATM address identified by this value
            of lesBusConfIndex is recreated following an agent
            restart, the same value of lesBusConfIndex must be
            used to identify the recreated BUS ATM address."
        ::= { lesBusEntry 1 }

    lesBusAddress      OBJECT-TYPE
        SYNTAX         AtmLaneAddress
        MAX-ACCESS     read-only
        STATUS         current
        DESCRIPTION
            "The ATM address of the BUS. This BUS is associated
            with LES specified by the lesConfIndex. A BUS may
            have several ATM addresses, this object provides
            the ATM address the LES returns in response to the
            LE-ARP request by the LEC."
        REFERENCE
            "LAN Emulation Over ATM Specification -
          version 1.0. S6."
        ::= { lesBusEntry 2 }


--
--  ATM LE-ARP table (by MAC address)
--
--  This table provides LE-ARP table for MAC-to-ATM addresses.
--  The entries are created by the agent and Network manager
--  depending on the entry type.
--        The entries can be any one of the following types:
--
--     viaRegister - This entry was registered by the LEC.
--
--     staticVolatile - This entry was created by Network manager.
--                This static entry WILL NOT survive
--                restart of the client.   Agent may allow this entry
--                to be created but may require LES to be in operational
--               state.
--
--
--     staticNonVolatile - This entry was created by Network manager.
--          This static entry will survive
--          restart of the client.   Agent may allow this entry
--                to be created but may require LES to be in
--                operational state.
--
```

```
--
--
--
--

lesLeArpMacTable  OBJECT-TYPE
    SYNTAX      SEQUENCE OF LesLeArpMacEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "This table provides access to an ATM LAN Emulation
        Server's MAC-to-ATM ARP table.  It contains entries
        for unicast addressed, the broadcast address.

        When the entry is for broadcast MAC address the
        corresponding ATM address is of a BUS.
        When the entry is for unicast MAC
        address the corresponding ATM address represents
        a LEC."
    ::= { lesConfGroup 5 }

lesLeArpMacEntry   OBJECT-TYPE
    SYNTAX       LesLeArpMacEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
        "An ATM LAN Emulation ARP table entry containing
        information about the binding of one MAC address
        to one ATM address."
    INDEX { lesConfIndex, lesLeArpMacAddr }
    ::= { lesLeArpMacTable 1 }

LesLeArpMacEntry ::=
    SEQUENCE {
        lesLeArpMacAddr
            MacAddress,
        lesLeArpLecId
            LecId,
        lesLeArpAtmAddr
            AtmLaneAddress,
        lesLeArpEntryType
            LeArpTableEntryType,
        lesLeArpRowStatus
            RowStatus
        }

lesLeArpMacAddr     OBJECT-TYPE
    SYNTAX        MacAddress
    MAX-ACCESS    not-accessible
    STATUS        current
    DESCRIPTION
        "The MAC address for which this table entry provides
        a translation.  Since ATM LAN Emulation uses an LE ARP
        protocol to locate the Broadcast and Unknown Server,
        the value of this object could be the broadcast MAC
        address.

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

lesLeArpLecId     OBJECT-TYPE
    SYNTAX        LecId
    MAX-ACCESS    read-only
    STATUS        current
    DESCRIPTION
        "The LE Client Identifier (LECID) of this
        entry.  Each LE Client requires a LECID
        assigned by the LE Server during the
        Join phase.
        If this entry is for a BUS then this object
```

```
            value will be zero."
         ::= { lesLeArpMacEntry 2 }


    lesLeArpAtmAddr      OBJECT-TYPE
       SYNTAX          AtmLaneAddress
       MAX-ACCESS      read-create
       STATUS          current
       DESCRIPTION
             "The ATM address of the Broadcast & Unknown Server
              or LAN Emulation Client whose MAC address is stored
              in 'lesLeArpMacAddr'.
              This volume may be registered by a LAN Emulation
              Client or specified by network management."
       ::= { lesLeArpMacEntry 3 }

    lesLeArpEntryType     OBJECT-TYPE
       SYNTAX           LeArpTableEntryType
       MAX-ACCESS       read-create
       STATUS           current
       DESCRIPTION
           "This object is used to indicate how
           this LE-ARP entry was created.
           The possible values for this object are:

             viaRegister(1),  -  agent
             staticVolatile(2), - network manager
             staticNonVolatile(3) - network manager

           This object is filled in by agent or network
           manager depending on the type."

       REFERENCE
            "LAN Emulation Client MIB definition."
     DEFVAL { staticVolatile }
       ::= { lesLeArpMacEntry 4 }

    lesLeArpRowStatus     OBJECT-TYPE
       SYNTAX           RowStatus
       MAX-ACCESS       read-create
       STATUS           current
       DESCRIPTION
           "This object is used to create or
           delete entries in the elanConfTable."
        REFERENCE "RFC 1443, [10] Textual Conventions
           for version 2 of the Simple Network Management
           Protocol (SNMPv2)."
       ::= { lesLeArpMacEntry 5 }

--
--  ATM LE-ARP table by Route Descriptors
--
--  This table provides ARP cache for Route Descriptor-to-ATM addresses.
--  The entries are created by the agent and Network manager
--  depending on the entry type.
--  The entries can be any one of the following types:
--
--     viaRegister - This entry was registered by the LEC.
--
--     staticVolatile - This entry was created by Network manager.
--          This static entry WILL NOT survive
--          restart of the client.   Agent may allow this entry
--          to be created but may require LES to be in operational
--          state.
--
--
--     staticNonVolatile - This entry was created by Network manager.
--          This static entry will survive
--          restart of the client.   Agent may allow this entry
--          to be created but may require LES to be in
--          operational state.
```

```
lesLeArpRdTable  OBJECT-TYPE
    SYNTAX      SEQUENCE OF LesLeArpRdEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
         "This table provides access to an ATM LAN Emulation
         Server's RouteDescriptor-to-ATM ARP cache.
         The entries in this table are set-up by the agent
         or network manager depending on the entry type.

         The Route Descriptors are presented as Segment Id
         (ring number) and Bridge number."
    ::= { lesConfGroup 6 }

lesLeArpRdEntry   OBJECT-TYPE
    SYNTAX       LesLeArpRdEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
         "An ATM LAN Emulation ARP cache entry containing
         information about the binding of one Route
         Descriptor to one ATM address."
    INDEX { lesConfIndex, lesLeArpRdSegId,
            lesLeArpRdBridgeNum }
    ::= { lesLeArpRdTable 1 }

   LesLeArpRdEntry ::=
       SEQUENCE {
           lesLeArpRdSegId
               INTEGER,
           lesLeArpRdBridgeNum
               INTEGER,
           lesLeArpRdLecId
               LecId,
           lesLeArpRdAtmAddr
               AtmLaneAddress,
           lesLeArpRdEntryType
               LeArpTableEntryType,
           lesLeArpRdRowStatus
               RowStatus
           }

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

   lesLeArpRdBridgeNum    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 LES."
       ::= { lesLeArpRdEntry 2 }


   lesLeArpRdLecId   OBJECT-TYPE
       SYNTAX        LecId
       MAX-ACCESS    read-only
       STATUS        current
       DESCRIPTION
            "The LE Client Identifier (LECID) of this
            entry.  Each LE Client requires a LECID
            assigned by the LE Server during the
```

```
                          Join phase.
"
                     ::= { lesLeArpRdEntry 3 }

                 lesLeArpRdAtmAddr      OBJECT-TYPE
                    SYNTAX          AtmLaneAddress
                    MAX-ACCESS      read-create
                    STATUS          current
                    DESCRIPTION
                          "The ATM address associated with the
                          Route Descriptor."
                    ::= { lesLeArpRdEntry 4 }

                 lesLeArpRdEntryType     OBJECT-TYPE
                    SYNTAX           LeArpTableEntryType
                    MAX-ACCESS       read-create
                    STATUS           current
                    DESCRIPTION
                        "This object is used to indicate how
                        this LE-ARP entry was learned:

                          viaRegister(1),  -  agent
                          staticVolatile(2), - network manager
                          staticNonVolatile(3) - network manager

                        This object is filled in by agent or network
                        manager depending on the type."
                   DEFVAL { staticVolatile }
                    ::= { lesLeArpRdEntry 5 }

                 lesLeArpRdRowStatus     OBJECT-TYPE
                    SYNTAX           RowStatus
                    MAX-ACCESS       read-create
                    STATUS           current
                    DESCRIPTION
                        "This object is used to create or
                        delete entries in the elanConfTable."
                     REFERENCE "RFC 1443, [10] Textual Conventions
                        for version 2 of the Simple Network Management
                        Protocol (SNMPv2)."
                    ::= { lesLeArpRdEntry 6 }

            --
            --  LES-LEC table
            --

             lesLecTableLastChange OBJECT-TYPE
                 SYNTAX       TimeStamp
                 MAX-ACCESS   read-only
                 STATUS       current
                 DESCRIPTION
                      "The value of sysUptime when an entry was created/deleted."
                 ::= { lesConfGroup 7 }


            lesLecTable      OBJECT-TYPE
                 SYNTAX       SEQUENCE OF LesLecEntry
                 MAX-ACCESS   not-accessible
                 STATUS       current
                 DESCRIPTION
                      "This table contains all LAN Emulation
                      clients serviced by LESs specified in
                      the lesConfTable.  This table can be
                      used to retrieve the topology of an
                      ELAN, in particular, the LES to LECs
                      mapping information.

                      An entry in this table is filled in
                      by the agent when a LEC registers successfully
                      with the LES.  Objects lesLecCtlDirectVpi
                      and lesLecCtlDirectVci can be modified  by the
                      network manager after creation if PVC is used."
```

```
    ::= { lesConfGroup 8 }

lesLecEntry       OBJECT-TYPE
    SYNTAX        LesLecEntry
    MAX-ACCESS    not-accessible
    STATUS        current
    DESCRIPTION
        "Each entry represents a LANE client to LES
        binding."
    INDEX { lesConfIndex, lesLecIndex }
    ::= { lesLecTable 1 }

LesLecEntry ::=
    SEQUENCE {
        lesLecIndex
            INTEGER,
        lesLecAtmAddr
            AtmLaneAddress,
        lesLecProxy
            TruthValue,
        lesLecId
            LecId,
        lesLecAtmIfIndex
            IfIndexOrZero,
        lesLecCtlDirectVpi
            VpiInteger,
        lesLecCtlDirectVci
            VciInteger,
        lesLecLastChange
            TimeStamp,
        lesLecState
            INTEGER,
        lesLecRowStatus
            RowStatus
        }

lesLecIndex        OBJECT-TYPE
    SYNTAX         INTEGER (1..65535)
    MAX-ACCESS     not-accessible
    STATUS         current
    DESCRIPTION
        "An arbitrary integer which uniquely identifies
        a LEC in this table."
    ::= { lesLecEntry 1 }

lesLecAtmAddr      OBJECT-TYPE
    SYNTAX         AtmLaneAddress
    MAX-ACCESS     read-only
    STATUS         current
    DESCRIPTION
        "The ATM address of the LANE client.  This
        is the primary ATM address of the LEC used
        in joining phase."
    REFERENCE
        "ATM Forum LAN Emulation Over ATM Specification, V1.0,
        Section 5.4.3"
    ::= { lesLecEntry 2 }

lesLecProxy        OBJECT-TYPE
    SYNTAX         TruthValue
    MAX-ACCESS     read-only
    STATUS         current
    DESCRIPTION
        "Whether this client is acting as a
        proxy.  Proxy clients are allowed
        to represent unregistered MAC addresses,
        and receive copies of LE_ARP_REQUEST
        packets for such addresses."
    REFERENCE
        "ATM Forum LAN Emulation Over ATM Specification, V1.0,
        Section 5.1.1"
    DEFVAL { false }
```

```
    ::= { lesLecEntry 3 }

lesLecId          OBJECT-TYPE
    SYNTAX        LecId
    MAX-ACCESS    read-only
    STATUS        current
    DESCRIPTION
         "The LE Client Identifier (LECID) of this
         entry.  Each LE Client requires a 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."
    DEFVAL { 0 }
    ::= { lesLecEntry 4 }

lesLecAtmIfIndex      OBJECT-TYPE
   SYNTAX       IfIndexOrZero
   MAX-ACCESS   read-create
   STATUS       current
   DESCRIPTION
       "The ifIndex of ATM port where this LEC entry
       has established the control direct VCC to
       the LES.  The value of this object
       maps to an existing ifIndex value in the
       ifTable of MIB-II.
       When an internal connection is used, this object is
     set to zero."
   ::= { lesLecEntry 5 }

 lesLecCtlDirectVpi OBJECT-TYPE
   SYNTAX     VpiInteger
   MAX-ACCESS read-create
   STATUS     current
   DESCRIPTION
         " The VPI of the bi-directional control
         direct connection between the LEC and LES."
   ::= { lesLecEntry 6 }

 lesLecCtlDirectVci OBJECT-TYPE
   SYNTAX     VciInteger
   MAX-ACCESS read-create
   STATUS     current
   DESCRIPTION
         " The VCI of the bi-directional control
         direct connection between the LEC and LES."
   ::= { lesLecEntry 7 }


lesLecLastChange  OBJECT-TYPE
   SYNTAX         TimeStamp
   MAX-ACCESS     read-only
   STATUS         current
   DESCRIPTION
      "The value of sysUpTime of this entry when the LEC enters
       the state indicated by the object lesLecState."
   ::= { lesLecEntry 8 }

lesLecState       OBJECT-TYPE
   SYNTAX         INTEGER {
       other(1),           -- LEC state is unknown
       noLesConnect(2),    -- LEC is not connected to the LES
       lesConnect(3),      -- LEC has established a
                           -- VCC connection to the LES
       joining(4),         -- LEC JOIN request has been
                           -- received by the LES
       addLec(5),          -- LES is setting up Control Distribute
                       -- VCC to this LEC
       joinedLes(6)        -- LEC has joined to the LES
                           -- successfully
```

```
        }
        MAX-ACCESS     read-only
        STATUS         current
        DESCRIPTION
             "This object is used to indicate the
             status this LEC entry is in from the
             LES's perspective.  The state of this
             object is updated by the agent as
             it discovers various phases of this LEC."
        ::= { lesLecEntry 9 }

      lesLecRowStatus OBJECT-TYPE
        SYNTAX     RowStatus
        MAX-ACCESS read-create
        STATUS     current
        DESCRIPTION
               " This object provides a way for the network manager
               to selectively remove a LE Client from the designated
               LES.  Or in a system where PVCs are used, this table
               is used to configure Control Direct VCCs between LES
               and LEC."
         REFERENCE "RFC 1443, [10] Textual Conventions
               for version 2 of the Simple Network Management
               Protocol (SNMPv2)."
        ::= { lesLecEntry 10 }

--
--  LES statistics table
--

lesStatTable    OBJECT-TYPE
    SYNTAX      SEQUENCE OF LesStatEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
         "This table contains all counters the
         LES maintain.  This table is an extention
         to the lesConfTable.  It provides performance
         and fault counters on a per LES basis."
    ::= { lesStatGroup 1 }

lesStatEntry     OBJECT-TYPE
    SYNTAX       LesStatEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
         "Each entry in this table contains a
         LES and its counters."
    AUGMENTS { lesConfEntry }
    ::= { lesStatTable 1 }

   LesStatEntry ::=
       SEQUENCE {
           lesStatJoinOk
               Counter32,
           lesStatVerNotSup
               Counter32,
           lesStatInvalidReqParam
               Counter32,
           lesStatDupLanDest
               Counter32,
           lesStatDupAtmAddr
               Counter32,
           lesStatInsRes
               Counter32,
           lesStatAccDenied
               Counter32,
           lesStatInvalidReqId
               Counter32,
           lesStatInvalidLanDest
               Counter32,
           lesStatInvalidAtmAddr
```

```
                Counter32,
            lesStatInBadPkts
                Counter32,
            lesStatOutRegFails
                Counter32,
             lesStatLeArpIn
                Counter32,
             lesStatLeArpFwd
                Counter32
            }

lesStatJoinOk      OBJECT-TYPE
    SYNTAX         Counter32
    MAX-ACCESS     read-only
    STATUS         current
    DESCRIPTION
         "Number of successful Join responses
         send out by the LAN Emulation Server."
    ::= { lesStatEntry 1 }

lesStatVerNotSup   OBJECT-TYPE
    SYNTAX         Counter32
    MAX-ACCESS     read-only
    STATUS         current
    DESCRIPTION
         "Number of version not supported errors.
"
    REFERENCE
         "ATM Forum LAN Emulation Over ATM Specification, V1.0,
         Table 13."
    ::= { lesStatEntry 2 }

lesStatInvalidReqParam      OBJECT-TYPE
    SYNTAX         Counter32
    MAX-ACCESS     read-only
    STATUS         current
    DESCRIPTION
         "Number of invalid request parameters errors.
"
    REFERENCE
         "ATM Forum LAN Emulation Over ATM Specification, V1.0,
         Table 13."
    ::= { lesStatEntry 3 }

lesStatDupLanDest OBJECT-TYPE
    SYNTAX         Counter32
    MAX-ACCESS     read-only
    STATUS         current
    DESCRIPTION
         "Number of duplicate LAN destination errors.
"
    ::= { lesStatEntry 4 }

lesStatDupAtmAddr OBJECT-TYPE
    SYNTAX         Counter32
    MAX-ACCESS     read-only
    STATUS         current
    DESCRIPTION
         "Number of duplicate ATM address errors.
"
    REFERENCE
         "ATM Forum LAN Emulation Over ATM Specification, V1.0,
         Table 13."
    ::= { lesStatEntry 5 }

lesStatInsRes      OBJECT-TYPE
    SYNTAX         Counter32
    MAX-ACCESS     read-only
    STATUS         current
    DESCRIPTION
         "Number of insufficient resources to grant
        errors.
```

```
"
        REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification, V1.0,
            Table 13."
        ::= { lesStatEntry 6 }

    lesStatAccDenied  OBJECT-TYPE
        SYNTAX        Counter32
        MAX-ACCESS    read-only
        STATUS        current
        DESCRIPTION
            "Number of access denied for security
            reasons errors.
"
        REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification, V1.0,
            Table 13."
        ::= { lesStatEntry 7 }

    lesStatInvalidReqId OBJECT-TYPE
        SYNTAX        Counter32
        MAX-ACCESS    read-only
        STATUS        current
        DESCRIPTION
            "Number of invalid LEC ID errors.
"
        REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification, V1.0,
            Table 13."
        ::= { lesStatEntry 8 }

    lesStatInvalidLanDest OBJECT-TYPE
        SYNTAX        Counter32
        MAX-ACCESS    read-only
        STATUS        current
        DESCRIPTION
            "Number of invalid LAN destination errors.
"
        REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification, V1.0,
            Table 13."
        ::= { lesStatEntry 9 }

    lesStatInvalidAtmAddr OBJECT-TYPE
        SYNTAX        Counter32
        MAX-ACCESS    read-only
        STATUS        current
        DESCRIPTION
            "Number of invalid ATM address errors.
"
        REFERENCE
            "ATM Forum LAN Emulation Over ATM Specification, V1.0,
            Table 13."
        ::= { lesStatEntry 10 }

    lesStatInBadPkts OBJECT-TYPE
        SYNTAX        Counter32
        MAX-ACCESS    read-only
        STATUS        current
        DESCRIPTION
            "Number of mal formed ATM ARP requests
          received by the LES."
      ::= { lesStatEntry 11 }

    lesStatOutRegFails OBJECT-TYPE
        SYNTAX        Counter32
        MAX-ACCESS    read-only
        STATUS        current
        DESCRIPTION
            "Number of registration failures sent
            out by this LES."
        ::= { lesStatEntry 12 }
```

```
    lesStatLeArpIn OBJECT-TYPE
         SYNTAX     Counter32
         MAX-ACCESS read-only
         STATUS     current
         DESCRIPTION
              "The total number of LE_ARP_REQUEST frames the LES has
              accepted since its last initialization."
         ::= { lesStatEntry 13 }

    lesStatLeArpFwd OBJECT-TYPE
         SYNTAX     Counter32
         MAX-ACCESS read-only
         STATUS     current
         DESCRIPTION
              "The number of LE_ARP_REQUESTs that the LES forwarded
              onto the clients (either via the control distribute or
              individually over each control direct) rather than
              answering directly.  This may be due to implementation
              decision (forward all requests) or because the resolution
              to the request did not reside in the LES's LE ARP cache."
         ::= { lesStatEntry 14 }


 --
 --  LES-LEC Statistics Table
 --

 lesLecStatTable    OBJECT-TYPE
     SYNTAX       SEQUENCE OF LesLecStatEntry
     MAX-ACCESS  not-accessible
     STATUS       current
     DESCRIPTION
          "This table contains all LE-ARP request
          related counters and error counts on
          a per LEC-LES pair basis."
     ::= { lesLecStatGroup 1 }

 lesLecStatEntry     OBJECT-TYPE
     SYNTAX        LesLecStatEntry
     MAX-ACCESS   not-accessible
     STATUS        current
     DESCRIPTION
          "Each entry in this table represents a
          LEC and its ARP counters.  This table
          is an extention to the lesLecTable."
     AUGMENTS { lesLecEntry }
     ::= { lesLecStatTable 1 }

LesLecStatEntry ::=
    SEQUENCE {
        lesLecRecvs
            Counter32,
        lesLecSends
            Counter32,
        lesLecInRegReq
            Counter32,
        lesLecInUnReg
            Counter32,
        lesLecInLeArpUcast
            Counter32,
        lesLecInLeArpBcast
            Counter32,
        lesLecInLeArpResp
            Counter32,
        lesLecInNArp
            Counter32
        }

lesLecRecvs      OBJECT-TYPE
    SYNTAX       Counter32
    MAX-ACCESS   read-only
```

```
    STATUS      current
    DESCRIPTION
         "Number of requests received from this
         LEC.  This includes all control
         frames as well as LE-ARP requests."
    ::= { lesLecStatEntry 1 }

lesLecSends     OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
         "Number of requests or responses
         sent to the LEC entry from this
        LES."
    ::= { lesLecStatEntry 3 }

lesLecInRegReq     OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
         "Number of Register requests received
         from this LEC.  "
    ::= { lesLecStatEntry 4 }

lesLecInUnReg     OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
         "Number of UnRegister requests received
         from this LEC.  "
    ::= { lesLecStatEntry 5 }

lesLecInLeArpUcast     OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
         "Number of LE-ARP requests for UNICAST
         address received from this LEC. "
    ::= { lesLecStatEntry 6 }

lesLecInLeArpBcast    OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
         "Number of LE-ARP requests for MULTICAST
         and Broadcast address received from this LEC. "
    ::= { lesLecStatEntry 7 }

lesLecInLeArpResp     OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
         "Number of LE-ARP responses
         received from this LEC. "
    ::= { lesLecStatEntry 8 }

lesLecInNArp      OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
         "Number of NARP requests
         received from this LEC. "
    ::= { lesLecStatEntry 10 }

--
```

```
-- LES Fault Mangement Group
--    (a) LES Error Control Table
--    (b) LES Error Log Table
--

   lesErrCtlTable OBJECT-TYPE
        SYNTAX     SEQUENCE OF LesErrCtlEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
               "This table contains error log control
                information of all LES instances.
                This table is an extention to the
                lesConfTable.  It is used to enable
                or disable error logs for a particular
               LES entry."
        ::= { lesFaultGroup 1 }

   lesErrCtlEntry OBJECT-TYPE
        SYNTAX     LesErrCtlEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
              "Each entry represents a LES entry
              in the lesConfTable."
        AUGMENTS { lesConfEntry }
        ::= { lesErrCtlTable 1 }

    LesErrCtlEntry ::=
        SEQUENCE {
            lesErrCtlAdminStatus
                INTEGER,
            lesErrCtlOperStatus
                INTEGER,
            lesErrCtlClearLog
                INTEGER,
            lesErrCtlMaxEntries
                INTEGER,
            lesErrCtlLastEntry
                LesErrLogIndexType
            }

    lesErrCtlAdminStatus      OBJECT-TYPE
            SYNTAX         INTEGER {
                enable(1),         -- enable error log
                disable(2)         -- disable error log
            }
            MAX-ACCESS    read-write
            STATUS        current
            DESCRIPTION
                "This object is used to enable/disable error
                logging for the LES."
            ::= { lesErrCtlEntry 1 }

    lesErrCtlOperStatus              OBJECT-TYPE
        SYNTAX        INTEGER {
             other(1),        -- not specified
             active(2),        -- error logging
             outOfRes(3),        -- Out of buffer error
             failed(4),        -- failed to start
                             -- error log for reasons
                             -- other than out of
                             -- resources
             disabled(5)      -- error logging is disabled
        }
        MAX-ACCESS    read-only
        STATUS        current
        DESCRIPTION
           "This object is used to indicate
           the result of a set operation to the
           object lesErrCtlAdminStatus.
           If the error log was successfully
```

```
        started, it is in active(2) mode.
        Otherwise, it is set to either
        outOfRes(3) or failed(4) for
        the respective reasons."
    ::= { lesErrCtlEntry 2 }

lesErrCtlClearLog            OBJECT-TYPE
    SYNTAX        INTEGER {
        noOp(1),         -- read only value
        clear(2)         -- clear the error log associated
                       -- with this LES entry
    }
    MAX-ACCESS        read-write
    STATUS                current
    DESCRIPTION
         "This object is used to clear the error log
          entries associated with this LES."
    ::= { lesErrCtlEntry 3 }

lesErrCtlMaxEntries        OBJECT-TYPE
    SYNTAX        INTEGER ( 1..65535 )
    MAX-ACCESS    read-only
    STATUS        current
    DESCRIPTION
        "The maximum entries of the error
        log a LES can support."
    ::= { lesErrCtlEntry 4 }

lesErrCtlLastEntry        OBJECT-TYPE
    SYNTAX        LesErrLogIndexType
    MAX-ACCESS    read-write
    STATUS        current
    DESCRIPTION
         "The index to the last entry in the error
          log table for this LES."
   ::= { lesErrCtlEntry 5 }

--
--
--   LES error log table
--

lesErrLogTable OBJECT-TYPE
    SYNTAX     SEQUENCE OF LesErrLogEntry
    MAX-ACCESS not-accessible
    STATUS     current
    DESCRIPTION
         "This table contains error logs
         of the LES instances enabled
         in the lesErrCtlTable.  This table
         is indexed by the LES instance
         index and an arbitrary integer
         uniquely identifies an error
        log."
    ::= { lesFaultGroup 2 }

lesErrLogEntry OBJECT-TYPE
    SYNTAX     LesErrLogEntry
    MAX-ACCESS not-accessible
    STATUS     current
    DESCRIPTION
         "Each entry represents a JOIN or REGISTER that
          was rejected due to an error."
    INDEX { lesConfIndex, lesErrLogIndex }
    ::= { lesErrLogTable 1 }

LesErrLogEntry ::=
    SEQUENCE {
        lesErrLogIndex
            LesErrLogIndexType,
        lesErrLogAtmAddr
            AtmLaneAddress,
```

```
                lesErrLogErrCode
                    INTEGER,
                lesErrLogTime
                    TimeStamp
            }

        lesErrLogIndex     OBJECT-TYPE
            SYNTAX          LesErrLogIndexType
            MAX-ACCESS      not-accessible
            STATUS          current
            DESCRIPTION
                "An arbitrary integer which uniquely
                identifies an error log entry. The first
                entry after reset or clearing the error
                log is an assigned value (2^32-1).  Succeding
                entries are assigned with descending values
                consecutively.  Entries after 1 are discarded. The
enabling/disabling of
                the error log capability is done in
                the lesErrCtlTable."
            ::= { lesErrLogEntry 1 }

        lesErrLogAtmAddr OBJECT-TYPE
            SYNTAX        AtmLaneAddress
            MAX-ACCESS    read-only
            STATUS        current
            DESCRIPTION
                "The ATM address of the requester
                which sends the JOIN or REGISTER request
                and causes the error to occur.
                The corresponding error code is
                specified in the object lesErrLogErrCode."
            ::= { lesErrLogEntry 2 }

        lesErrLogErrCode OBJECT-TYPE
            SYNTAX        INTEGER (0..22)
            MAX-ACCESS    read-only
            STATUS        current
            DESCRIPTION
                "The Error code which indicates the
                cause of the error triggered by the
                JOIN or REGISTER request sent by the
                requester indicated by the value of
                the object lesErrLogAtmAddr."
            ::= { lesErrLogEntry 3 }

        lesErrLogTime    OBJECT-TYPE
            SYNTAX        TimeStamp
            MAX-ACCESS    read-only
            STATUS        current
            DESCRIPTION
                "The sysUpTime when this entry was logged by
                the LES."
            ::= { lesErrLogEntry 4 }



    -- Conformance Information

    lesMIBConformance       OBJECT IDENTIFIER ::= { lesMIB 5 }

    lesMIBGroups            OBJECT IDENTIFIER ::=
                              { lesMIBConformance 1 }

    lesMIBCompliances       OBJECT IDENTIFIER ::=
                              { lesMIBConformance 2 }

    -- Compliance Statements

    lesMIBCompliance         MODULE-COMPLIANCE
            STATUS            current
```

```
                DESCRIPTION      "The compliance statement for SNMP
                     entities which support the ATM LAN Emulation
 LES MIB."


                MODULE -- this module
                      MANDATORY-GROUPS  { lesCConfGroup,
                                          lesCStatGroup,
                                      lesLecCStatGroup,
                                      lesFaultCGroup }


                OBJECT lesVccRowStatus
                MIN-ACCESS read-only
                DESCRIPTION
                     " Implementations that do not support LES over
                        PVCs are not required to allow write/create access
                        to the lesVccRowStatus object."


                OBJECT lesLecAtmIfIndex
                MIN-ACCESS read-only
                DESCRIPTION
                        " Implementations that do not support LECs over
                        PVCs are not required to allow write/create access
                        to the lesLecAtmIfIndex object."

                OBJECT lesLecCtlDirectVpi
                MIN-ACCESS read-only
                DESCRIPTION
                        " Implementations that do not support LECs over
                        PVCs are not required to allow write/create access
                        to the lesLecCtlDirectVpi object."

                OBJECT lesLecCtlDirectVci
                MIN-ACCESS read-only
                DESCRIPTION
                        " Implementations that do not support LECs over
                        PVCs are not required to allow write/create access
                        to the lesLecCtlDirectVci object."

                OBJECT lesLecRowStatus
                WRITE-SYNTAX INTEGER { destroy(6) }
                DESCRIPTION
                     " Implementations that do not support LECs over
                        PVCs need only support the destroy enumeration
                        of the RowStatus textual convention."

                GROUP   lesRdGroup
                DESCRIPTION
                        "This group is mandatory only for those
                         LESs that support elan802.5."

                ::= { lesMIBCompliances 1 }

                -- Units of Conformance

                lesCConfGroup    OBJECT-GROUP
                      OBJECTS { lesConfNextId,
                       lesAtmAddrSpec,
                       lesAtmAddrMask,
                       lesAtmAddrActual,
                       lesElanName,
                       lesLanType,
                       lesLastChange,
                       lesControlTimeOut,
                       lesMaxFrameSize,

                       lesVccRowStatus,

                       lesOperStatus,
                       lesAdminStatus,
```

```
          lesRowStatus,
          lesBusAddress,
          lesLeArpLecId,
          lesLeArpAtmAddr,
          lesLeArpEntryType,
          lesLeArpRowStatus,

          lesLecTableLastChange,
          lesLecAtmIfIndex,
          lesLecProxy,
          lesLecAtmAddr,
          lesLecId,
          lesLecCtlDirectVpi,
          lesLecCtlDirectVci,
          lesLecLastChange,
          lesLecRowStatus,
          lesLecState

        }
        STATUS     current
        DESCRIPTION
           "A collection of objects providing configuration
            information about the LAN Emulation Services."
        ::= { lesMIBGroups 1 }

lesRdGroup    OBJECT-GROUP
        OBJECTS {
          lesLeArpRdLecId,
          lesLeArpRdAtmAddr,
          lesLeArpRdEntryType,
          lesLeArpRdRowStatus }
        STATUS     current
        DESCRIPTION
            "A group of objects used for 802.5
            ATM LAN Emulation management only."
        ::= { lesMIBGroups 2 }

 lesCStatGroup    OBJECT-GROUP
        OBJECTS {
          lesStatJoinOk,
          lesStatInBadPkts,
          lesStatOutRegFails,
          lesStatVerNotSup,
          lesStatInvalidReqParam,
          lesStatDupLanDest,
          lesStatDupAtmAddr,
          lesStatInsRes,
          lesStatAccDenied,
          lesStatInvalidReqId,
          lesStatInvalidLanDest,
          lesStatInvalidAtmAddr,
          lesStatLeArpIn,
          lesStatLeArpFwd
        }
        STATUS   current
        DESCRIPTION
            "A collection of objects providing
            monitoring information about the
            LES."
        ::= { lesMIBGroups 3 }

 lesLecCStatGroup  OBJECT-GROUP
        OBJECTS {
          lesLecRecvs,
          lesLecSends,
          lesLecInRegReq,
          lesLecInUnReg,
          lesLecInLeArpUcast,
          lesLecInLeArpBcast,
          lesLecInLeArpResp,
          lesLecInNArp
        }
```

```
            STATUS    current
            DESCRIPTION
                 "A collection of objects providing
                 monitoring information about the
                 LEC to/from LES traffic."
            ::= { lesMIBGroups 4 }

      lesFaultCGroup  OBJECT-GROUP
            OBJECTS {
                 lesErrCtlAdminStatus,
                 lesErrCtlOperStatus,
                 lesErrCtlClearLog,
                 lesErrCtlMaxEntries,
                 lesErrCtlLastEntry,
                 lesErrLogAtmAddr,
                 lesErrLogErrCode,
                 lesErrLogTime
          }
          STATUS   current
            DESCRIPTION
                 "A collection of objects providing
               fault management for the LES."
            ::= { lesMIBGroups 5 }


END
```

```
LAN-EMULATION-BUS-MIB DEFINITIONS ::= BEGIN

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

        MODULE-COMPLIANCE, OBJECT-GROUP FROM SNMPv2-CONF

        TEXTUAL-CONVENTION, RowStatus, DisplayString,
                TimeStamp        FROM SNMPv2-TC

        AtmLaneAddress,
        VpiInteger,
        VciInteger,
        atmfLanEmulation
                FROM LAN-EMULATION-CLIENT-MIB

        IfIndexOrZero, AtmLaneMask
                FROM LAN-EMULATION-ELAN-MIB;


 busMIB MODULE-IDENTITY
        LAST-UPDATED "9602121200Z"
        ORGANIZATION "ATM Forum LAN Emulation Sub-Working Group"
        CONTACT-INFO
             "
                The ATM Forum

                2570 West El Camino Real, Ste 304,
                Mountain View, CA 94040-1313
                E-mail: info@atmforum.com"

DESCRIPTION
        " The MIB module for the management of LANE broadcast and
        known servers. This MIB should be used in conjunction
        with the ELAN and LES MIBs."
::= { atmfLanEmulation 4}

    --
    --    Textual Conventions
    --

     BusLocalIndex ::= TEXTUAL-CONVENTION
         STATUS     current
         DESCRIPTION
             "An unique number identifying an
             BUS.  This number is only
             used locally by the agent to
             distinguish between BUS."
         SYNTAX     Integer32

     BusErrLogIndexType ::= TEXTUAL-CONVENTION
         STATUS     current
         DESCRIPTION
                 "An arbitrary integer used in identifying
                 an erro log associated with a BUS."
         SYNTAX     Integer32(1..2147483647)


    -- This MIB module consists of the following groups:
    --
    --  (1) BUS Conf Group
    --      (a) BUS table
    --      (b) BUS-VCC table
    --      (c) BUS-LEC table
    --
    --  (2) Stat Group
    --      (d) BUS Stat table
    --      (e) BUS-LEC Statistics table
```

```
--
--  (3) Fault Management Group
--       (a) error control table
--       (b) error log table


busConfGroup  OBJECT IDENTIFIER ::= { busMIB 1 }

busStatGroup  OBJECT IDENTIFIER ::= { busMIB 2 }

busFaultGroup OBJECT IDENTIFIER ::= { busMIB 3 }

--
--  (1) BUS Conf Group
--  (a) BUS table
--


busConfNextId  OBJECT-TYPE
    SYNTAX     BusLocalIndex
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
        "The next available BUS index.  The
        value of this object can be used as
        the index by the network manager to
        create an entry in the busConfTable."
    ::= { busConfGroup 1 }

busConfTable   OBJECT-TYPE
    SYNTAX     SEQUENCE OF BusConfEntry
    MAX-ACCESS not-accessible
    STATUS     current
    DESCRIPTION
        "This table contains all LAN Emulation
        Broadcast and Unknown Servers (BUS) this
        agent manages.  The BUS handles data sent
        by an LE Client to the broadcast MAC
        address, all multicast traffic, and initial unicast
        frames which are sent by a LEC before the data
        direct target ATM address has been resolved.
        There can be multiple BUSs per ELAN, but a
        BUS can service only one ELAN."
    ::= { busConfGroup 2 }

busConfEntry  OBJECT-TYPE
    SYNTAX     BusConfEntry
    MAX-ACCESS not-accessible
    STATUS     current
    DESCRIPTION
        "Each entry in this table represents a
        BUS.  The parameters in each entry apply
        to one emulated LAN served by one BUS.
        Object busRowStatus is required during
        row creation and deletion.  Object busElanName
        is used to indicate the ELAN this BUS
        is servicing and is used to cross reference
        tables defined in the LAN Emulation Server MIB.
        Note that objects busAtmAddrSpec and busAtmAddrMask
        are used to configure the ATM address of a BUS.
        The BUS typically derives it's ATM address
        from the switch or the network and the actual
        ATM address used is indicated in the object
        busAtmAddrActual."
    INDEX      { busConfIndex }
    ::= { busConfTable 1 }


BusConfEntry ::= SEQUENCE {
    busConfIndex                  BusLocalIndex,
    busConfAtmAddrSpec            AtmLaneAddress,
    busConfAtmAddrMask            AtmLaneMask,
```

```
    busConfAtmAddrActual            AtmLaneAddress,
    busConfElanName                 DisplayString,
    busConfLastChange               TimeStamp,
    busConfMaxFrameAge              INTEGER,
    busConfOperStatus               INTEGER,
    busConfAdminStatus              INTEGER,
    busConfRowStatus                RowStatus
}

busConfIndex OBJECT-TYPE
    SYNTAX     BusLocalIndex
    MAX-ACCESS not-accessible
    STATUS     current
    DESCRIPTION
      "A value which uniquely identifies a
      conceptual row in the busConfTable.

      If the conceptual row identified by this value
      of busConfIndex is recreated following an agent
      restart, the same value of busConfIndex must be
      used to identify the recreated row."
    ::= { busConfEntry 1 }

busConfAtmAddrSpec OBJECT-TYPE
    SYNTAX     AtmLaneAddress
    MAX-ACCESS read-create
    STATUS     current
    DESCRIPTION
      "An ATM  address specified by  the
      network or local management that,
      with the ATM address mask, determines
      a  portion of  the  ATM address  that
      the  BUS on the designated ATM interface
      will use  to derive the actual ATM
      address from the network or ILMI.  The derived
      ATM address is specified in the object
      busAtmAddrActual, which is used to
      receive multicast or broadcast traffic."
    ::= { busConfEntry 2 }

busConfAtmAddrMask OBJECT-TYPE
    SYNTAX     AtmLaneMask
    MAX-ACCESS read-create
    STATUS     current
    DESCRIPTION
      "The ATM address mask associated with
      the object busAtmAddrSpec.  The value
      of the mask is an ATM address with
      the don't care portion set to zero
      and the valid ATM address portion set to one."
     DEFVAL { 'FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF'H }
    ::= { busConfEntry 3 }

busConfAtmAddrActual OBJECT-TYPE
    SYNTAX     AtmLaneAddress
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
        " The resultant ATM address  in use by  the BUS.  This
        object is a product of the specified ATM address, mask
        and  interaction  with  the network.  This object is
        created by the agent."
    ::= { busConfEntry 4 }

busConfElanName OBJECT-TYPE
    SYNTAX     DisplayString (SIZE (0..32))
    MAX-ACCESS read-create
    STATUS     current
    DESCRIPTION
      "The name of the ELAN this BUS is providing service to."
    ::= { busConfEntry 5 }
```

```
 busConfLastChange OBJECT-TYPE
    SYNTAX     TimeStamp
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
            "The  value of the sysUpTime  when  this BUS
            has entered the state indicated by the
            object busConfOperStatus."
    ::= { busConfEntry 6 }

busConfMaxFrameAge   OBJECT-TYPE
      SYNTAX        INTEGER (1..4)
      MAX-ACCESS    read-create
      STATUS        current
      DESCRIPTION
          "Time out period for a frame that has been
          received but not been transmitted by BUS
          to all relevant Multicast Send VCCs or
          Multicast Forward VCCs."
      REFERENCE
           "LAN Emulation Over ATM Specification -
          version 1.0. S5."
      DEFVAL { 1 }
      ::= { busConfEntry 7 }

 busConfOperStatus OBJECT-TYPE
    SYNTAX     INTEGER {
           other(1),        -- unknown state
           up(2),           -- BUS is up and running
           down(3)          -- BUS is down or not available
       }
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
           "The operational state of this BUS entry.
            When in 'up' state the BUS will forward
            LEC traffic.  Any other state the
            BUS is not available for service and may
            release all the existing VCCs and refuse
            service to all clients."
    ::= { busConfEntry 8 }

 busConfAdminStatus OBJECT-TYPE
    SYNTAX     INTEGER {
           up(2),           -- BUS is up and running
           down(3)          -- BUS is down or not available
       }
    MAX-ACCESS read-create
    STATUS     current
    DESCRIPTION
            "  The  desired  state   of  the  designated   BUS  as
            prescribed by the operator. The  actions of the  agent
            will, if at  all  possible, eventually  result  in the
            desired state being  reflected in  the busOperStatus."
    DEFVAL     { up }
    ::= { busConfEntry 9 }

 busConfRowStatus OBJECT-TYPE
    SYNTAX     RowStatus
    MAX-ACCESS read-create
    STATUS     current
    DESCRIPTION
            "This object is used to create or
            delete entries in the busConfTable."
     REFERENCE "RFC 1443, [10] Textual Conventions
         for version 2 of the Simple Network Management
        Protocol (SNMPv2)."
    ::= { busConfEntry 10 }

--
--  (b) BUS VCC table
--
```

```
busVccTable      OBJECT-TYPE
    SYNTAX       SEQUENCE OF BusVccEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
        "This table contains all the Multicast Forward
        VCCs used by the BUS to forward multicast traffic
        to the participating LECs.  The Multicast Forward
        VCC can either be point-to-point or point-to-
        multipoint calls.  This table is read only if
        SVCs are used and writable if PVCs are used."
    ::= { busConfGroup 3 }

busVccEntry       OBJECT-TYPE
    SYNTAX        BusVccEntry
    MAX-ACCESS    not-accessible
    STATUS        current
    DESCRIPTION
        "Each entry in this table represents a
         Multicast Forward VCC of the BUS."
     INDEX { busConfIndex, busVccAtmIfIndex,
             busVccMtFwdVpi, busVccMtFwdVci }
    ::= { busVccTable 1 }

BusVccEntry ::=
    SEQUENCE {
        busVccAtmIfIndex
            IfIndexOrZero,
        busVccMtFwdVpi
            VpiInteger,
        busVccMtFwdVci
            VciInteger,
        busVccRowStatus
            RowStatus
    }

busVccAtmIfIndex  OBJECT-TYPE
       SYNTAX     IfIndexOrZero
       MAX-ACCESS not-accessible
       STATUS     current
       DESCRIPTION
          "The ATM interface which the Multicast
          Forward VCC is running on.
          This value must match an existing
          value  in the   ifTable.
          The value of this object is set
          to zero when the ATM interface is
          undefined."
       ::= { busVccEntry 1 }

busVccMtFwdVpi    OBJECT-TYPE
       SYNTAX     VpiInteger
       MAX-ACCESS not-accessible
       STATUS     current
       DESCRIPTION
          "The VPI value of the Multicast
          Forward VCC.  The object
          busVccAtmIfIndex, busVccMtFwdVci
          and the value of this object uniquely
          identfies a VCC within a ATM host."
       ::= { busVccEntry 2 }

busVccMtFwdVci    OBJECT-TYPE
       SYNTAX     VciInteger
       MAX-ACCESS not-accessible
       STATUS     current
       DESCRIPTION
          "The VCI value of the Multicast
          Forward VCC.  The object
          busVccAtmIfIndex, busVccMtFwdVpi
          and the value of this object uniquely
          identfies a VCC within a ATM host."
```

```
        ::= { busVccEntry 3 }

busVccRowStatus   OBJECT-TYPE
        SYNTAX     RowStatus
        MAX-ACCESS read-create
        STATUS     current
        DESCRIPTION
             "This object is used to create or
             delete entries in the busConfTable."
        REFERENCE "RFC 1443, [10] Textual Conventions
             for version 2 of the Simple Network Management
             Protocol (SNMPv2)."
        ::= { busVccEntry 4 }



--
--  (b) BUS - LEC table
--

busLecTableLastChange OBJECT-TYPE
    SYNTAX     TimeStamp
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
            "The value of sysUpTime when an entry of the
             busLecTable was created/deleted."
    ::= { busConfGroup 4 }

busLecTable    OBJECT-TYPE
    SYNTAX     SEQUENCE OF BusLecEntry
    MAX-ACCESS not-accessible
    STATUS     current
    DESCRIPTION
             " This table contains the BUS and the
             actual LECs being serviced by the BUS.
             It can be used as the actual mapping
             between BUS and LEC.

             This table provides information
             for Multicast send VCCs
             between BUS and clients.  Objects
             busLecMcastSendAtmIfIndex, busLecMcastSendVpi,
             and busLecMcstSendVci can only be modified
             if PVC is used."
    ::= { busConfGroup 5 }

busLecEntry OBJECT-TYPE
    SYNTAX       BusLecEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
         "Each entry represents a BUS to LEC
         mapping."
    INDEX { busConfIndex , busLecAtmAddr }
    ::= { busLecTable 1 }

BusLecEntry::=
        SEQUENCE {
         busLecAtmAddr
             AtmLaneAddress,
         busLecMcastSendAtmIfIndex
             IfIndexOrZero,
         busLecMcastSendVpi
             VpiInteger,
         busLecMcastSendVci
             VciInteger,
         busLecRowStatus
             RowStatus
    }

busLecAtmAddr     OBJECT-TYPE
```

```
    SYNTAX        AtmLaneAddress
    MAX-ACCESS    not-accessible
    STATUS        current
    DESCRIPTION
            "The ATM address of the LEC.  This
            is the primary ATM address of the LEC."
    ::= { busLecEntry 1 }

busLecMcastSendAtmIfIndex  OBJECT-TYPE
    SYNTAX     IfIndexOrZero
    MAX-ACCESS read-create
    STATUS     current
    DESCRIPTION
            "The  ATM  interface  index this BUS uses for
            Multicast Send traffic.  The value of this
         object has to exist in the ifTable in MIB II
         unless an internal connection is used.  When
         an internal connection is used, this object
         is set to zero."
    ::= { busLecEntry 2 }

busLecMcastSendVpi         OBJECT-TYPE
    SYNTAX     VpiInteger
    MAX-ACCESS read-create
    STATUS     current
    DESCRIPTION
            " The   virtual  path  identifier used to receive
            multicast traffic by this BUS."
    ::= { busLecEntry 4 }

busLecMcastSendVci         OBJECT-TYPE
    SYNTAX     VciInteger
    MAX-ACCESS read-create
    STATUS     current
    DESCRIPTION
            " The   virtual  channel  identifier used to receive
            multicast traffic by this BUS."
    ::= { busLecEntry 5 }

busLecRowStatus OBJECT-TYPE
    SYNTAX     RowStatus
    MAX-ACCESS read-create
    STATUS     current
    DESCRIPTION
            " This object provides a way for the network manager
            to selectively remove a LE Client from the designated
            BUS.  Or in a system where PVCs are used, this table
            is used to create Multicast Send VCCs between BUS
           and LEC."
::= { busLecEntry 6 }


--
--  (2) BUS Stat Group
--
--  (a) BUS Statistic table
--

busStatTable    OBJECT-TYPE
    SYNTAX      SEQUENCE OF BusStatEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "This table contains all counters the
        BUS maintain.  This table is an extention
        to the busConfTable."
    ::= { busStatGroup 1 }


busStatEntry OBJECT-TYPE
    SYNTAX     BusStatEntry
    MAX-ACCESS not-accessible
```

```
    STATUS     current
    DESCRIPTION
        "Each entry in this table contains a
        BUS and its counters."
    AUGMENTS { busConfEntry }
    ::= { busStatTable 1 }

BusStatEntry ::= SEQUENCE {
  busStatInDiscards            Counter32,
    busStatInOctets                  Counter32,
    busStatInUcastFrms               Counter32,
    busStatInMcastFrms              Counter32,
    busStatFrmTimeOuts              Counter32,
    busStatMcastSendRefused                Counter32,
    busStatMcastFwdFailure                Counter32
}

busStatInDiscards OBJECT-TYPE
    SYNTAX     Counter32
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
            "  The number of frames discarded due to resource
         error."
    ::= { busStatEntry 1 }

busStatInOctets OBJECT-TYPE
    SYNTAX     Counter32
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
            "  The number of  octets that  this  BUS has received
            since its initialization."
    ::= { busStatEntry 2 }

busStatInUcastFrms OBJECT-TYPE
    SYNTAX     Counter32
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
            " The number  of  frames that  the BUS  has  received
            which were  unicast data frames  and all control
            frames (i.e. they were flooded from the client)."
    ::= { busStatEntry 3 }

busStatInMcastFrms OBJECT-TYPE
    SYNTAX     Counter32
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
            "  The number of  frames  that  the BUS has  received
            which were multicast frames."
    ::= { busStatEntry 4 }

busStatFrmTimeOuts OBJECT-TYPE
    SYNTAX     Counter32
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
            " The number of frames dropped by
            the BUS due to time out."
    ::= { busStatEntry 5 }

busStatMcastSendRefused OBJECT-TYPE
    SYNTAX     Counter32
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
            "  The number of  multicast send VCCconnection setup attempts
to the BUS which were refused."
    ::= { busStatEntry 6 }
```

```
             busStatMcastFwdFailure OBJECT-TYPE
                 SYNTAX     Counter32
                 MAX-ACCESS read-only
                 STATUS     current
                 DESCRIPTION
                          "  The number of  multicast forward VCCconnection setup
attempts from the BUS which were unsuccessful
                         for any reason."
                 ::= { busStatEntry 7 }


            --
            --  (b) BUS - LEC statistics table
            --

            busLecStatTable    OBJECT-TYPE
                SYNTAX       SEQUENCE OF BusLecStatEntry
                MAX-ACCESS  not-accessible
                STATUS      current
                DESCRIPTION
                     "This table contains all LEC counters
                     the BUS maintains.  This table can also
                     be used to retrieve all LECs a BUS
                     is providing service to."
                ::= { busStatGroup 2 }

            busLecStatEntry     OBJECT-TYPE
                SYNTAX       BusLecStatEntry
                MAX-ACCESS   not-accessible
                STATUS       current
                DESCRIPTION
                     "Each entry in this table represents a
                     LEC and its counters."
                AUGMENTS { busLecEntry }
                ::= { busLecStatTable 1 }

               BusLecStatEntry ::=
                   SEQUENCE {
                       busLecRecvs         Counter32,
                       busLecForwards          Counter32,
                       busLecDiscards          Counter32
                      }


               busLecRecvs      OBJECT-TYPE
                   SYNTAX      Counter32
                   MAX-ACCESS  read-only
                   STATUS      current
                   DESCRIPTION
                        "Number of Multicast, Broadcast and
                        Unknown Forward requests received by
                        the BUS from this LEC."
                   ::= { busLecStatEntry 1 }

               busLecForwards  OBJECT-TYPE
                   SYNTAX      Counter32
                   MAX-ACCESS  read-only
                   STATUS      current
                   DESCRIPTION
                        "Number of Multicast, Broadcast and
                        Unkown Forward requests forwarded by
                        the BUS from this LEC.  The value
                        of this object indicate how many requests
                        have been forwarded by the BUS."
                   ::= { busLecStatEntry 2 }

               busLecDiscards  OBJECT-TYPE
                   SYNTAX      Counter32
                   MAX-ACCESS  read-only
                   STATUS      current
                   DESCRIPTION
                        "Number of Multicast, Broadcast and
```

```
                 Unkown Forward requests discarded by
                 the BUS from this LEC.  The value
                 of this object indicate how many requests
                 have been discarded by the BUS."
             ::= { busLecStatEntry 3 }




--
-- BUS Fault Mangement Group
--    (a) BUS Error Control Table
--    (b) BUS Error Log Table
--

    busErrCtlTable OBJECT-TYPE
         SYNTAX      SEQUENCE OF BusErrCtlEntry
         MAX-ACCESS not-accessible
         STATUS      current
         DESCRIPTION
                "This table contains error log control
                 information of all BUS instances.
                 This table is an extention to the
                 busConfTable.  It is used to enable
                 or disable error logs for a particular
                 BUS entry."
         ::= { busFaultGroup 1 }

    busErrCtlEntry OBJECT-TYPE
         SYNTAX      BusErrCtlEntry
         MAX-ACCESS not-accessible
         STATUS      current
         DESCRIPTION
               "Each entry represents a BUS entry
               in the busConfTable."
         AUGMENTS { busConfEntry }
         ::= { busErrCtlTable 1 }

     BusErrCtlEntry ::=
         SEQUENCE {
             busErrCtlAdminStatus
                 INTEGER,
             busErrCtlOperStatus
                 INTEGER,
             busErrCtlClearLog
                 INTEGER,
             busErrCtlMaxEntries
                 INTEGER,
             busErrCtlLastEntry
                 BusErrLogIndexType
             }

     busErrCtlAdminStatus      OBJECT-TYPE
             SYNTAX         INTEGER {
                 enable(1),         -- enable error log
                 disable(2)         -- disable error log
             }
             MAX-ACCESS    read-write
             STATUS        current
             DESCRIPTION
                  "This object is used to enable/disable error
                 logging for the BUS."
             ::= { busErrCtlEntry 1 }

     busErrCtlOperStatus             OBJECT-TYPE
         SYNTAX        INTEGER {
              other(1),        -- not specified
              active(2),        -- error logging
              outOfRes(3),        -- Out of buffer error
              failed(4),        -- failed to start
                              -- error log for reasons
                              -- other than out of
```

```
                          -- resources
            disabled(5)      -- error loggin was disabled
        }
        MAX-ACCESS    read-only
        STATUS        current
        DESCRIPTION
            "This object is used to indicate
            the result of a set operation to the
            object busErrCtlAdminStatus.
            If the error log was successfully
            started, it is in active(2) mode.
            Otherwise, it is set to either
            outOfRes(3) or failed(4) for
            the respective reasons."
        ::= { busErrCtlEntry 2 }

    busErrCtlClearLog            OBJECT-TYPE
        SYNTAX        INTEGER {
            noOp(1),         -- read only value
            clear(2)         -- clear the error log associated
                           -- with this BUS entry
        }
        MAX-ACCESS         read-write
        STATUS                 current
        DESCRIPTION
             "This object is used to clear the error log
              entries associated with this BUS."
        ::= { busErrCtlEntry 3 }

    busErrCtlMaxEntries         OBJECT-TYPE
        SYNTAX        INTEGER ( 1..65535 )
        MAX-ACCESS    read-only
        STATUS        current
        DESCRIPTION
            "The maximum entries of the error
            log a BUS can support."
        ::= { busErrCtlEntry 4 }

    busErrCtlLastEntry         OBJECT-TYPE
        SYNTAX        BusErrLogIndexType
        MAX-ACCESS    read-write
        STATUS        current
        DESCRIPTION
             "The index to the last entry in the error
              log table for this BUS."
        ::= { busErrCtlEntry 5 }

    --
    --
    --   BUS error log table
    --

    busErrLogTable OBJECT-TYPE
        SYNTAX      SEQUENCE OF BusErrLogEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
             "This table contains error logs
             of the BUS instances enabled
             in the busErrCtlTable.  This table
             is indexed by the BUS instance
             index and an arbitrary integer
             uniquely identifies an error
             log."
        ::= { busFaultGroup 2 }

    busErrLogEntry OBJECT-TYPE
        SYNTAX     BusErrLogEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
             "Each entry represents aan error detected by the BUS."
```

```
                 INDEX { busConfIndex, busErrLogIndex }
                 ::= { busErrLogTable 1 }

             BusErrLogEntry ::=
                 SEQUENCE {
                     busErrLogIndex
                         BusErrLogIndexType,
                     busErrLogAtmAddr
                         AtmLaneAddress,
                     busErrLogErrCode
                         INTEGER,
                     busErrLogTime
                         TimeStamp
                 }

             busErrLogIndex     OBJECT-TYPE
                 SYNTAX           BusErrLogIndexType
                 MAX-ACCESS       not-accessible
                 STATUS           current
                 DESCRIPTION
                      "An arbitrary integer which uniquely
                      identifies an error log entry. The first
                      entry after reset or clearing the error
                      log is an assigned value (2^32-1).  Succeding
                      entries are assigned with descending values
                      consecutively.  Entries after 1 are discarded.The
enabling/disabling of
                      the error log capability is done in
                      the busErrCtlTable."
                 ::= { busErrLogEntry 1 }

             busErrLogAtmAddr OBJECT-TYPE
                 SYNTAX        AtmLaneAddress
                 MAX-ACCESS    read-only
                 STATUS        current
                 DESCRIPTION
                      "The primary ATM address of the LE Client on whose Multicast
                      Send VCC the error occured.
                      The corresponding error code is
                      specified in the object busErrLogErrCode."
                 ::= { busErrLogEntry 2 }

             busErrLogErrCode OBJECT-TYPE
                 SYNTAX        INTEGER {
                                outOfRes(1),      -- Out of resources error
                                badCtlFrame(2),   -- Malformed control frame
                                badDataFrame(3),  -- Malformed data frame, i.e.
 -- the frame received by the
                                                  -- BUS is either too big or too small
                               other(4)           -- any other errors
                               }
                 MAX-ACCESS    read-only
                 STATUS        current
                 DESCRIPTION
                      "The Error code which indicates the
                      cause of the error."
                 ::= { busErrLogEntry 3 }

             busErrLogTime    OBJECT-TYPE
                 SYNTAX        TimeStamp
                 MAX-ACCESS    read-only
                 STATUS        current
                 DESCRIPTION
                      "The sysUpTime when this entry was logged by
                      the BUS."
                 ::= { busErrLogEntry 4 }



         busMIBConformance OBJECT IDENTIFIER ::= { busMIB 4 }
```

```
busMIBGroups OBJECT IDENTIFIER ::= { busMIBConformance 1 }

busMIBCompliances OBJECT IDENTIFIER ::= { busMIBConformance 2 }

busMIBCompliance MODULE-COMPLIANCE
    STATUS current
    DESCRIPTION
        " The compliance statement for SNMP entities that support
        the ATM LAN Emulation BUS MIB."

MODULE -- this module
        MANDATORY-GROUPS    { busCConfGroup, busCStatGroup, busCFaultGroup }

        OBJECT busConfAtmAddrSpec
        MIN-ACCESS read-only
        DESCRIPTION
             " Those implementations that do not support seperately
         configured broadcast/unknown and LE servers are note required to
         provide write access to the busConfAtmAddrSpec object."

        OBJECT busConfAtmAddrMask
        MIN-ACCESS read-only
        DESCRIPTION
             " Those implementations that do not support seperately
         configured broadcast/unknown and LE servers are note required to
         provide write access to the busConfAtmAddrMask object."

        OBJECT busConfElanName
        MIN-ACCESS read-only
        DESCRIPTION
             " Those implementations that do not support seperately
         configured broadcast/unknown and LE servers are note required to
         provide write access to the busConfElanName object."

        OBJECT busConfAdminStatus
        MIN-ACCESS read-only
        DESCRIPTION
             " Those implementations that do not support seperately
         configured broadcast/unknown and LE servers are note required to
         provide write access to the busConfAdminStatus object."

        OBJECT busConfRowStatus
        MIN-ACCESS read-only
        DESCRIPTION
             " Those implementations that do not support seperately
         configured broadcast/unknown and LE servers are note required to
         provide write access to the busConfRowStatus object."

        OBJECT busVccRowStatus
        MIN-ACCESS read-only
        DESCRIPTION
             " Implementations that do not support broadcast/unknown
         servers over PVCs are not required to allow write/create
         access to the busVccRowStatus object."

        OBJECT busLecMcastSendAtmIfIndex
        MIN-ACCESS read-only
        DESCRIPTION
             " Implementations that do not support LECs over PVCs
         are note required to allow write/create access to the
         busLecMcastSendAtmIfIndex object."

        OBJECT busLecMcastSendVpi
        MIN-ACCESS read-only
        DESCRIPTION
             " Implementations that do not support LECs over
         PVCs are note required to allow write/create access to
         the busLecMcastSendVpi object."

        OBJECT busLecMcastSendVci
        MIN-ACCESS read-only
```

```
        DESCRIPTION
             " Implementations that do not support LECs over
        PVCs are note required to allow write/create access to
        the busLecMcastSendVci object."

        OBJECT busLecRowStatus
        WRITE-SYNTAX INTEGER { destroy(6) }
        DESCRIPTION
             " Implementations that do not support LECs over
        PVCs need only support the destroy enumeration of the
        RowStatus textual convention."

      GROUP busLecCGroup
      DESCRIPTION
             "This group is optional."

        ::= { busMIBCompliances 1 }

-- Units of Conformance


busCConfGroup OBJECT-GROUP
        OBJECTS   { busConfNextId,
                busConfAtmAddrSpec,
                busConfAtmAddrMask,
                busConfAtmAddrActual,
                busConfElanName,
                busConfLastChange,
                busConfMaxFrameAge,
                busConfOperStatus,
                busConfAdminStatus,
                busConfRowStatus,
                busVccRowStatus,
                busLecTableLastChange,
                busLecMcastSendAtmIfIndex,
                busLecMcastSendVpi,
                busLecMcastSendVci,
                busLecRowStatus
                }
        STATUS    current
        DESCRIPTION
                " A collection of objects for the
                managing of BUS operation."
        ::= { busMIBGroups 1 }

busCStatGroup OBJECT-GROUP
        OBJECTS   {
                busStatInOctets,
             busStatInDiscards,
                busStatInUcastFrms,
                busStatInMcastFrms,
                busStatFrmTimeOuts,
                busStatMcastSendRefused,
                busStatMcastFwdFailure
        }
        STATUS    current
        DESCRIPTION
                " A collection of objects providing information
                about BUS statistics."
        ::= { busMIBGroups 2 }

busCFaultGroup OBJECT-GROUP
        OBJECTS   {
                busErrLogAtmAddr,
              busErrLogErrCode,
                busErrLogTime,
                busErrCtlAdminStatus,
                busErrCtlOperStatus,
                busErrCtlClearLog,
                busErrCtlMaxEntries,
                busErrCtlLastEntry
        }
```

```
                STATUS    current
                DESCRIPTION
                        " A collection of objects providing information
                        about BUS statistics."
                ::= { busMIBGroups 3 }

        busLecCGroup OBJECT-GROUP
                OBJECTS   {
                        busLecRecvs,
                        busLecForwards,
                        busLecDiscards
                }
                STATUS    current
                DESCRIPTION
                        " A collection of objects providing information
                        about LEC-BUS statistics."
                ::= { busMIBGroups 4 }
END
```