The ATM Forum

Technical Committee

# LAN Emulation Over ATM
# Version 1.0

**af-lane-0021.000**

**January, 1995**

# List of Figures

# List of Tables

# 1. Introduction

Most data traffic in existing Customer Premises Networks (CPNs) is sent over Local Area Networks (LANs), such as Ethernet/IEEE 802.3 and IEEE 802.5 networks. The services provided by today's LANs differ from those of ATM, for example:

1. The messages may be characterized as connectionless, versus the connection-oriented approach of ATM;

2. Broadcast and multicast are easily accomplished through the shared medium of a LAN;

3. LAN MAC addresses, based on manufacturing serial numbers, are independent of the network topology.

In order to use the vast base of existing LAN application software, it is necessary to define an ATM service, herein called "LAN Emulation," that emulates services of existing LANs across an ATM network and can be supported via a software layer in end systems.

Indeed, if such a LAN Emulation service is provided for an ATM network, then end systems (e.g. workstations, servers, bridges, etc.) can connect to the ATM network while the software applications interact as if they are attached to a traditional LAN. Also, this service will support interconnection of ATM networks with traditional LANs by means of today's bridging methods. This will allow interoperability between software applications residing on ATM-attached end systems and on traditional LAN end systems.

The LAN Emulation service will be important to the acceptance of ATM, since it provides a simple and easy means for running existing LAN applications in the ATM environment. Networking customers have indicated in recent surveys, including the ATM Forum-sponsored McQuillan Consulting survey (June, 1993), [1] that coexistence with existing networks and transition to ATM need to be addressed. Likewise, the surveys have concluded that ATM for work group LANs and LAN backbones are the top planned uses. Customers expect to continue to use existing LAN applications as they migrate to ATM.

To emulate a LAN-like service, different types of emulation can be defined, ranging from emulating the MAC service (e.g. that of IEEE 802.x LANs) up to emulating the services of network and transport layers. This specification defines a MAC service emulation, including encapsulation of MAC frames (user data frames). This approach to LAN emulation provides support for the maximum number of existing applications.

The LAN Emulation Service is one of several B-ISDN services [16]. LAN Emulation is a specific type of a conversational service for data information.

## LAN-SPECIFIC CHARACTERISTICS TO BE EMULATED

### Connectionless Services

LAN stations today are able to send data without previously establishing connections. LAN Emulation provides the appearance of such a connectionless service to the participating end systems.

### Multicast Services

The LAN emulation service supports the use of multicast MAC addresses (e.g. broadcast, group, or functional MAC addresses). The need for a multicast service for LAN Emulation comes from classical LANs where end stations share the same media.

Note that supporting broadcast/multicast traffic does not necessarily mean that all messages addressed to a multicast MAC address must be distributed to every station. A large number of today's LAN protocols use broadcast or multicast messages. A service could be established to intercept these messages and forward them directly to their destinations instead of broadcasting them to every station. A simpler alternative would be to forward multicast messages to all stations and then rely upon filtering in those stations, as is done in existing LANs. This simpler approach is adopted in LAN Emulation.

### MAC Driver Interfaces In ATM Stations

The main objective of the LAN emulation service is to enable existing applications to access an ATM network via protocol stacks like APPN, NetBIOS, IPX, AppleTalk etc. as if they were running over traditional LANs. Since in today's implementations these protocol stacks are communicating with a MAC driver, the LAN emulation service has to offer the same MAC driver service primitives, thus keeping the upper protocol layers unchanged.

There are today some "standardized" interfaces for MAC device drivers: e.g. NDIS (Network Driver Interface Specification)[2], ODI (Open Data-Link Interface)[3] and DLPI (Data Link Provider Interface)[18]. They specify how to access a MAC driver. Each of them has its own primitives and parameter sets, but the essential services/functions are the same. LAN Emulation provides these interfaces and services to the upper layers (see Section 3.1).

**Emulated LANs**

In some environments there might be a need to configure multiple, separate domains within a single network. This requirement leads to the definition of an "emulated LAN" which comprises a group of ATM-attached devices. This group of devices would be logically analogous to a group of LAN stations attached to an Ethernet/IEEE 802.3 or 802.5 LAN segment.

Several emulated LANs (ELANs) could be configured within an ATM network, and membership in an emulated LAN is independent of where an end system is physically connected. An end system could belong to multiple emulated LANs.

Since multiple emulated LANs over a single ATM network are logically independent a broadcast frame originating from a member of a particular emulated LAN is distributed only to the members of that emulated LAN.

**Interconnection With Existing LANs**

As mentioned before, the LAN emulation service provides not only connectivity between ATM-attached end systems, but also connectivity with LAN-attached stations. This includes connectivity both from ATM stations to LAN stations as well as LAN stations to LAN stations across ATM. MAC layer LAN Emulation is defined in such a way that existing bridging methods can be employed, as they are defined today. Note that bridging methods include both Transparent Bridging and Source Routing Bridging[4].

## 1.1 Purpose of Document

This document specifies an implementation agreement for the LAN Emulation Service, including the following:

- Architectural Framework and Service Interfaces
- Service Components and Functions
- Frame Formats
- Client/Server Protocols and Procedures

## 1.2 Terminology

The following acronyms and terminology are used throughout this document:

| | |
|---|---|
| AAL | ATM Adaptation Layer |
| ARE | All Routes Explorer |

| | |
|---|---|
| ATM | Asynchronous Transfer Mode |
| B-LLI | Broadband Low Layer Information |
| BN | Bridge Number |
| BPP | Bridge Port Pair (Source Routing Descriptor) |
| BPDU | Bridge Protocol Data Unit |
| BUS | Broadcast and Unknown Server |
| CPCS | Common Part Convergence Sublayer |
| CPN | Customer Premises Network |
| DA | Destination MAC address |
| ELAN | Emulated Local Area Network |
| IE | Information Element |
| IEEE | Institute of Electrical and Electronics Engineers |
| IETF | Internet Engineering Task Force |
| IP | Internet Protocol |
| LAN | Local Area Network |
| LD | LAN Destination |
| LE | LAN Emulation |
| LE_ARP | LAN Emulation Address Resolution Protocol |
| LEC | LAN Emulation Client |
| LECID | LAN Emulation Client Identifier |
| LECS | LAN Emulation Configuration Server |
| LES | LAN Emulation Server |
| LSB | Least Significant Bit |
| LTH | Length Field |
| MAC | Medium Access Control |
| MIB | Management Information Base |
| MSB | Most Significant Bit |
| MTU | Message Transfer Unit |
| NDIS | Network Driver Interface Specification |
| NSR | Non-Source Routed |
| ODI | Open Data-Link Interface |
| OSI | Open Systems Interconnection |
| OUI | Organizational Unit Identifier |
| PDU | Protocol Data Unit |
| QOS/QoS | Quality of Service |
| RC | Routing Control |

| | |
|---|---|
| RD | Route Descriptor |
| RFC | Request For Comment (Document Series) |
| RI | Routing Information |
| RII | Routing Information Indicator |
| RT | Routing Type |
| SA | Source MAC address |
| SAP | Service Access Point |
| SAAL | Signaling AAL |
| SDU | Service Data Unit |
| SR | Source Routing (Bridging) |
| SRF | Specifically Routed Frame |
| SRT | Source Routing Transparent |
| SSCS | Service Specific Convergence Sublayer |
| STE | Spanning Tree Explorer |
| TB | Transparent Bridging |
| TCP | Transmission Control Protocol |
| TLV | Type / Length / Value |
| UNI | User-Network Interface |
| VCC | Virtual Channel Connection |
| VPC | Virtual Path Connection |
| VCI | Virtual Channel Identifier |
| VPI | Virtual Path Identifier |

This document uses normative statements throughout as follows:

**Table 1. Normative Statements**

| Statement | Verbal Form[1] |
|---|---|
| Requirement | MUST/MUST NOT |
| Recommendation | SHOULD/SHOULD NOT |
| Permission | MAY |

## 1.3 References


[1] McQuillan Consulting, *Early Adopters of ATM - A Report to the ATM Forum*, June, 1993.

[2] 3COM/Microsoft, *LAN Manager: Network Driver Interface Specification*, October 8, 1990.

[3] Novell Incorporated, *Open Data-Link Interface Developer's Guide,* March 20, 1992.

[4] ISO/IEC 10038: ANSI/IEEE Std. 802.1D *Information processing systems - Local Area Networks - MAC Sublayer Interconnection (MAC Bridges).*

[5] The ATM Forum, *ATM User-Network Interface Specification, Version 3.0,* September 10, 1993.

[6] IEEE Std. 802: *IEEE Standards for Local and Metropolitan Area Networks: Overview and Architecture.*

[7] ISO / IEC 10039 *Information technology - Telecommunications and information exchange between systems -Medium access control service definition.*

[8] ISO 8802-2: ANSI/IEEE Std. 802.2 *Information processing systems - Local area networks - Part 2: Logical Link Control.*

[9] ISO 8802-2: ANSI/IEEE Std. 802.5P (Draft) *Information processing systems - Local area networks - Part 2: Logical Link Control Supplement - Route Determination Entity.*

[10] ISO / IEC 8802-3: ANSI/IEEE Std. 802.3 *Information processing systems - Local area networks - Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications.*

[11] ISO / IEC 8802-5: ANSI/IEEE Std. 802.5 *Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - LAN/MAN-type specific requirements - Part 5: Token ring access method and physical layer specifications.*

[12] ISO / IEC 7023 DTR *Information technology - Telecommunications and information exchange between systems - Standard Group MAC Addresses.*

[13] ISO / IEC 2382-25 *Information technology - Vocabulary - Part 25: Local area networks.*

[14] ITU-T I.363 *B-ISDN ATM Adaptation Layer(AAL) Specification, Section 6 (AAL5).*


[1]Verbal forms are based on ISO except for “Requirements,” where ISO uses the terms “SHALL/SHALL NOT” instead of “MUST/MUST NOT” in this document.

[15] *The Ethernet - A Local Area Network*, Digital Equipment Corporation, Intel,and Xerox, AA-K759B-TK, November, 1982.

[16] ITU-T Recommendation I.211 *Integrated Services Digital Network (ISDN) - General Structure and Service Capabilities - B-ISDN Service Aspects,* Geneva, 1991.

[17] The ATM Forum, *ATM User-Network Interface Version 3.1 (UNI 3.1) Specification,* July 21, 1994.

[18] UNIX International, *Data Link Provider Interface (DLPI) Specification*, Revision 2.0.0, OSI Work Group, August 1991.

[19] ITU-T Recommendation X.6 "Public Data Networks: Services and Facilities: Multicast Service Definition" March 1993.

[20] IETF, RFC 1577 *Classical IP and ARP Over ATM,* Network Working Group, January, 1994.

[21] IETF, RFC 1626 *Default IP MTU for use over AAL5,* Network Working Group, May, 1994.

## 1.4 ATM Network Service Assumptions

This LAN Emulation Over ATM specification is based on the ATM Forum User-Network Interface Specification, Version 3.0 [5] or later. The specification provides example Information Element codings for UNI 3.0 and 3.1[17].

# 2. Description of LAN Emulation Service

## 2.1 Architectural Overview

### 2.1.1 Basic Concepts

LAN Emulation enables the implementation of emulated LANs over an ATM network. An emulated LAN provides communication of user data frames among all its users, similar to a physical LAN. One or more emulated LANs could run on the same ATM network. However, each of the emulated LANs is independent of the others and users cannot communicate directly across emulated LAN boundaries. Note that communication between emulated LANs is possible only through routers or bridges (possibly implemented in the same end station).

This document specifies the operation of a single emulated LAN only. Each emulated LAN is one of two types: Ethernet/IEEE 802.3 or IEEE 802.5 (Token Ring). Each emulated LAN is composed of a set of LAN Emulation Clients (LE Clients, or LECs) and a single LAN Emulation Service (LE Service). This LE Service consists of an LE Configuration Server (LECS), an LE Server (LES) and a Broadcast and Unknown Server (BUS). Each LE Client is part of an ATM end station. It represents a set of users, identified by their MAC Address. The LE Service may be part of an end station or a switch: it may be centralized or distributed over a number of stations.

Communication among LE Clients and between LE Clients and the LE Service is performed over ATM virtual channel connections (VCCs). Each LE Client must communicate with the LE Service over control and data VCCs. Emulated LANs operate in any of the following environments: Switched Virtual Circuit (SVC), Permanent Virtual Circuit (PVC) or mixed SVC/PVC.

In a PVC-only LAN there are no call setup and close down procedures; instead layer management is used to set up and clear connections. In this PVC environment the layer management is responsible for both setting up and clearing connections and has full responsibility for ensuring that the emulated LAN functions correctly.

### 2.1.2 Architectural Perspective

The architecture of a communication system emphasizes the logical divisions of the system and how they fit together. This document incorporates the following types of architectural views:

- The (internal) layer interfaces that specify the interaction between the LAN Emulation entity and the other entities within the end-station.
- The user-to-network interface that specifies the interaction between an LE Client and the LE Service over the ATM network.
- "Network-to-network" interfaces are beyond the scope of this document.

#### 2.1.2.1 Layer Interfaces

In this architectural model, the layers interact by way of well-defined service interfaces, providing services as specified in Section 3. In general, the interface requirements are as follows (see Figure 1):

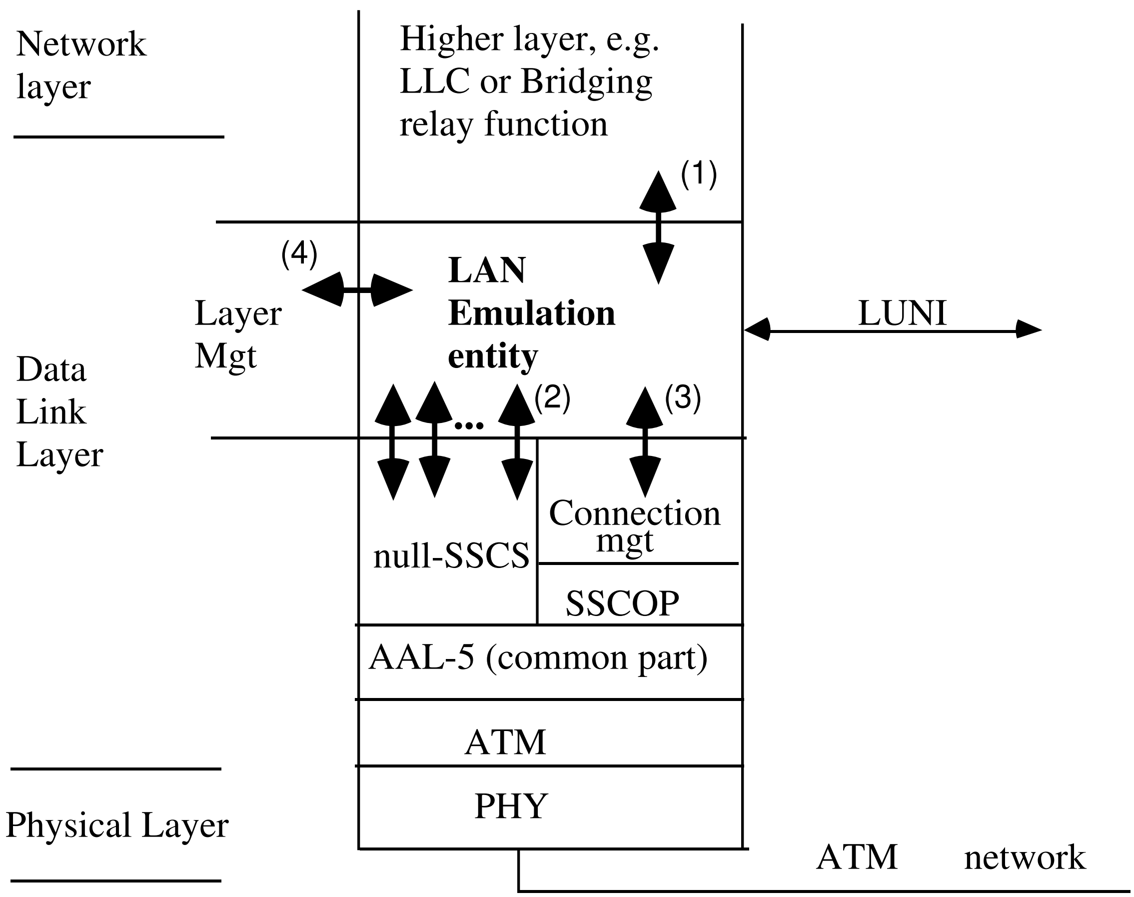



**Figure 1. The Layered Architecture of LAN Emulation**

(1) The interface between the LAN Emulation layer and the Higher layers includes facilities for transmitting and receiving user data frames.

(2) The interfaces between the LAN Emulation layer and the ATM Adaptation Layer (AAL) include facilities for transmitting and receiving AAL-5 frames. AAL-5 utilizes lower layers, including the ATM and PHY. Interface service access points are identified by SAP-IDs (that have a one to one mapping to VCCs).

(3) The interface between the LAN Emulation entity and the Connection management entity includes facilities to request the setup or release of virtual connections. This entity handles both SVCs and/or PVCs.

(4) The interface between the LAN Emulation entity and the Layer Management entity includes facilities to initialize and control the LAN Emulation entity and to return status information.

#### 2.1.2.2 LAN Emulation User to Network Interface (LUNI)

In this architectural model, the LE Clients and the LE Service interact by way of a well defined interface, using PDUs and implementing protocols as specified later in this document. In general, the interface requirements are as follows (see Figure 2):

(1) Initialization:

- obtaining the ATM-Address(es) of the LE Services that are available on a particular ATM network, and
- joining or leaving a particular emulated LAN specified by the ATM Address of the LE Service.
- declaring whether this LE Client wants to receive address resolution requests for all the frames with unregistered destinations.

(2) Registration: Informing the LE Service of the following:

- the list of individual MAC Addresses that the LE Client represents
- the list of Source Route descriptors (i.e., Segment/Bridge pairs) that the LE Client represents for Source Route bridging

(3) Address resolution: Obtaining the ATM Address representing the LE Client with a particular MAC Address (Unicast, Broadcast or Segment/Bridge pair).

(4) Data transfer: Moving the data from the source to the destination by:

- encapsulation of the LE-SDU (Service Data Unit) in an AAL-5 frame and transmission by the LE Client
- forwarding the AAL-5 frame by the LE Service (if applicable)
- receiving and decapsulating the AAL-5 frame by the LE Client

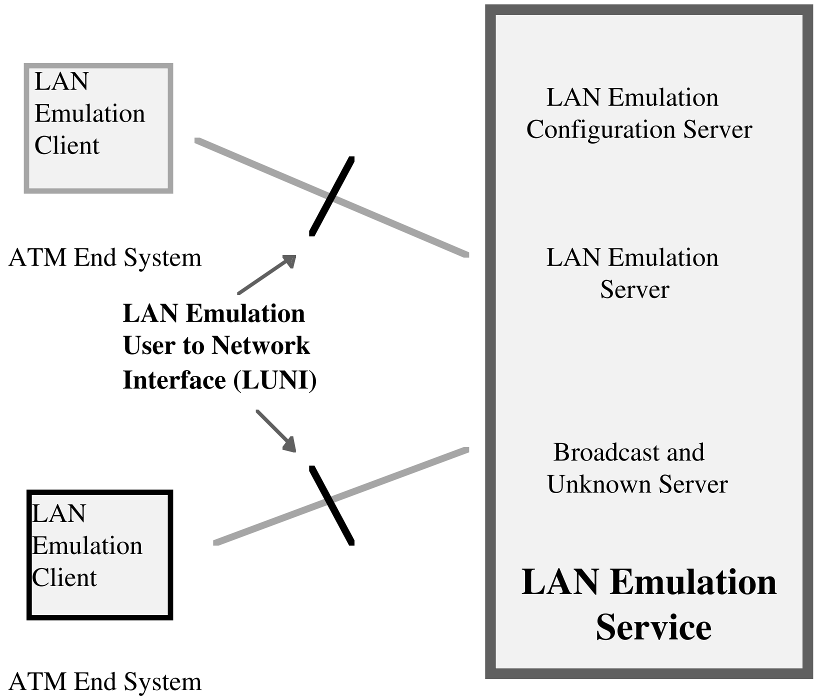



**Figure 2. The LAN Emulation User to Network Interface (LUNI)**

### 2.1.3 Implementation Perspective

Users connect to the LAN Emulation service via LE Clients. LE Clients are typically implemented in ATM end stations, either as part of the software driver (between the Operating System and the ATM hardware) or on a special processor that is part of the ATM Adapter (the ATM specific hardware).

LAN Emulation is expected to be used in either of two configurations:

(1) Intermediate Systems (e.g., bridges or routers). These devices enable the communication among "legacy" LANs over ATM backbone networks.

(2) End Stations (e.g., hosts or PCs). These devices enable the communication between ATM end stations and end stations on "legacy" LAN or among ATM end stations.

The LE Service might be implemented in an ATM intermediate system or an end station (e.g., a bridge, router or dedicated work station). Alternatively it may be "part of the ATM network", namely, implemented in switches or other ATM specific devices.

A possible implementation might be a single (centralized) LE Service. An alternative implementation could be a distributed one, e.g., where a number of servers operate in parallel and provide the redundancy required for error recovery.

The LE Service could also be co-located with one or more LE Clients - potentially saving on hardware costs.

## 2.2 LAN Emulation Components

### 2.2.1 Entities

The components of an Emulated LAN network include Clients (e.g. ATM workstations and ATM bridges) each having at least one LE Client entity, and the components of the LE Service (an LE Server, a Broadcast and Unknown Server, and a LAN Emulation Configuration Server.).

Any of the LAN emulation service components may be distributed over multiple physical entities or may be collapsed into fewer physical entities, even a single one. This document does not specify the means by which multiple physical entities cooperate to share the function of one LAN emulation service component.

#### 2.2.1.1 LE Client (LEC)

The LAN emulation Client is the entity in end systems which performs data forwarding, address resolution, and other control functions. This provides a MAC level emulated Ethernet/IEEE 802.3 or IEEE 802.5 service interface to higher level software and implements the LUNI interface when communicating with other entities within the Emulated LAN.

#### 2.2.1.2 LE Server (LES)

The LE Server implements the control coordination function for the Emulated LAN. The LE Server provides a facility for registering and resolving MAC addresses and/or route descriptors to ATM addresses. Clients may register the LAN destinations they represent with the LE Server. A Client will also query the LE Server when the client wishes to resolve a MAC address and/or route descriptor to an ATM address. The LE Server will either respond directly to the client or forward the query to other clients so they may respond.

#### 2.2.1.3 Broadcast and Unknown Server (BUS)

The Broadcast and Unknown Server handles data sent by an LE Client to the broadcast MAC address ('FFFFFFFFFFFF'), 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 (before a data direct VCC has been established).

Note, the LAN Emulation architecture supports visibility of a single Broadcast and Unknown Server at the LUNI. All broadcast, multicast and unknown traffic to and from an LE Client passes through this single entity. Distributed and redundant implementations of the BUS are explicitly permitted with this architecture. The multicast server function provided in the BUS is required as part of LAN Emulation to provide the connectionless data delivery characteristics of a shared network to LAN Emulation Clients. The main task is to distribute data with Multicast MAC addresses (e.g. group, broadcast, and functional addresses); to deliver initial Unicast data, where the MAC address hasn’t been resolved to a direct ATM connection; and to distribute data with explorer Source Routing information.

The multicast function provided in the BUS may be implemented by an underlying ATM Multicast Service (or some other mechanism). The BUS multicast function is assumed to be consistent with ITU-T Recommendation X.6 Multicast Service Definition [19].

An LEC sends data frames to the BUS which serializes the frames and re-transmits them to a group of attached LECs. Serialization is required to prevent AAL5 frames from different sources from being interleaved.

In an SVC environment the BUS needs to participate in the LE Address Resolution Protocol (LE_ARP) to enable a LEC to locate the BUS. The BUS also handles ATM connections and manages its distribution group.

This BUS must always exist in the Emulated LAN and all LECs must join its distribution group.

#### 2.2.1.4 LE Configuration Server (LECS)

The LE Configuration Server implements the assignment of individual LE clients to different emulated LANs. Based upon its own policies, configuration database and information provided by clients, it assigns any client which requests configuration information to a particular emulated LAN service by giving the client the LES's ATM address. This method supports the ability to assign a client to an emulated LAN based on either the physical location (ATM address) or the identity of a LAN destination which it is representing.

It is optional for the LEC to obtain information from the LECS using the configuration protocol. The LECS allows the LEC to automatically configure.

### 2.2.2 Connections

A LAN Emulation Client has separate VCCs for control traffic such as LE_ARP requests and for data traffic to transfer encapsulated IEEE 802.3 or IEEE 802.5 frames. Each VCC carries traffic for only one Emulated LAN. The VCCs form a mesh of connections between the LECs and other LAN Emulation entities such as the LECS, LES and the BUS.

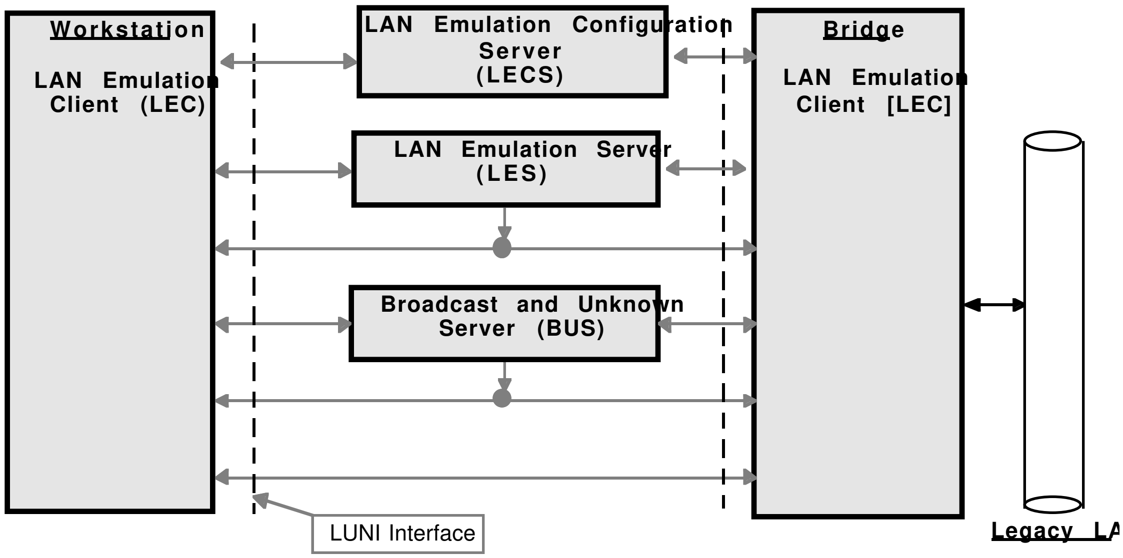



**Figure 3. Basic LAN Emulation Client Connections across LUNI**

Figure 3 shows an example of the set of connections across the LUNI interface in a simple configuration of two LECs, the LECS, the LES and the BUS. Other combinations may exist. See additional explanation below.

### 2.2.2.1 Control Connections

A Control VCC links the LEC to the LECS. Control VCCs also link the LEC to the LES and carry LE_ARP traffic and control frames. The control VCCs never carry data frames. Control VCCs are set up as part of the LEC initialization phase and are shown in Figure 4 and 5.

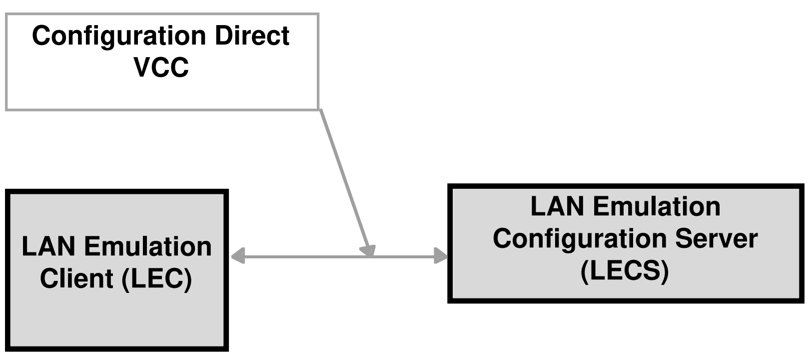



**Figure 4. LAN Emulation Client to LAN Emulation Configuration Server (LECS) Control Connection**

#### 2.2.2.1.1 Configuration Direct VCC

This bi-directional VCC may be set up by the LEC (or other entity) as part of the LECS Connect phase and is used to obtain configuration information, including the address of the LES. The entity may still maintain this VCC while participating in the emulated LAN. It may continue to keep it open for further queries to the LE Configuration Service while participating in the

Emulated LAN. The Configuration Direct VCC may be used to inquire about an LE Client other than the one to which the Configuration Direct VCC is attached. This connection is signaled using B-LLI to indicate it carries "LE Control" packet formats (see Signaling Section 3.3.2).

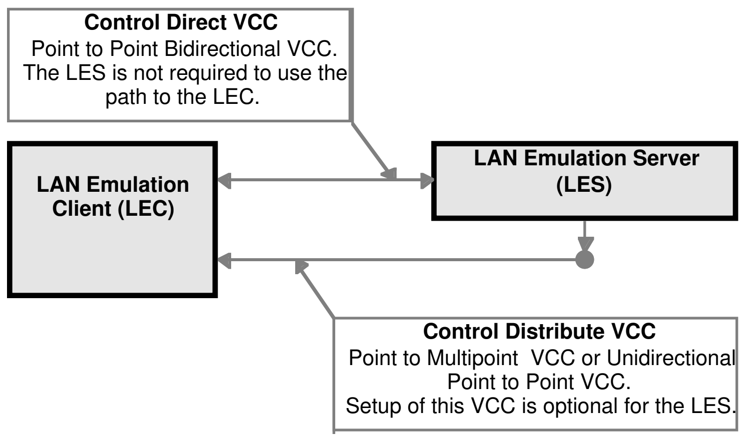



**Figure 5. LAN Emulation Client to LAN Emulation Server (LES) Control Connections**

### 2.2.2.1.2 Control Direct VCC

The LEC sets up a bi-directional point-to-point VCC to the LES for sending control traffic. This is set up by the LEC as part of the initialization phase. Since the LES has the option to use the return path to send control data to the LEC, this requires the LEC to accept control traffic from this VCC.

The LEC and LES must maintain this VCC while participating in the Emulated LAN.

### 2.2.2.1.3 Control Distribute VCC

The LES may optionally set up a unidirectional point-to-point or point-to-multipoint control VCC to the LEC for distributing control traffic. This VCC may be set up by the LES as part of the initialization phase. If set up, the LEC is required to accept the Control Distribute VCC regardless of type.

The LEC and LES must maintain this VCC while participating in the Emulated LAN.

### 2.2.2.2 Data Connections

Data VCCs connect the LECs to each other and to the Broadcast and Unknown Server. These carry Ethernet/IEEE 802.3 or IEEE 802.5 data frames as well as flush messages (defined later). Apart from flush messages, data VCCs never carry control traffic.

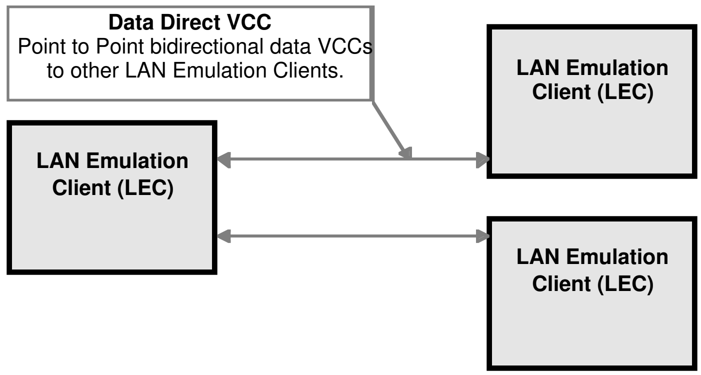



**Figure 6. LAN Emulation Client to Client Data Connections**

### 2.2.2.2.1 Data Direct VCC

Bi-directional point to point VCCs are established between LECs that want to exchange unicast data traffic. Figure 6 illustrates the Data Direct VCC. Connections from an LE Client to the BUS are shown in Figure 7.

When an LE Client has a packet to send and the ATM address for that destination is unknown, the LE Client shall generate an LE_ARP request to ascertain the ATM address for that destination. Once the LEC receives a reply to the LE_ARP it shall set up a point-to-point VCC, if not already established, over which to send all subsequent data to that LAN destination.

The LAN emulation client that issues an LE_ARP request and receives an LE_ARP response is responsible for initiating the signaling to establish this bi-directional Data Direct VCC with the responding client named in the LE_ARP response.

If the client has insufficient resources to set up a Data Direct VCC it must not continue to send frames to the Broadcast and Unknown Server that it should have sent on the Data Direct VCC. In this case it should tear down an existing Data Direct VCC to another Client in order to free up resources to allow the new Data Direct VCC to be set up.

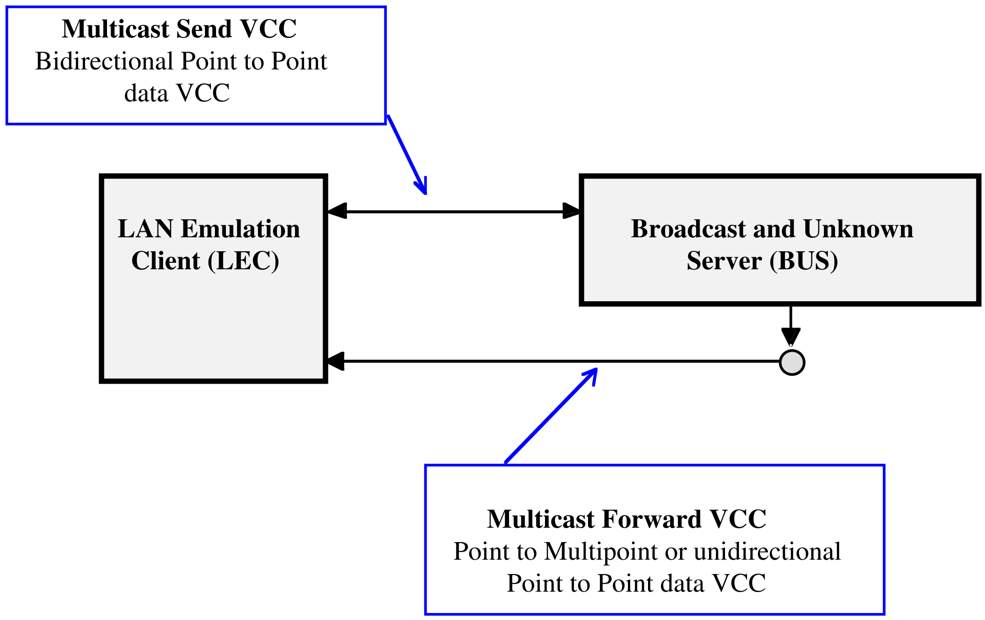



**Figure 7. LAN Emulation Client to BUS Connections**

### 2.2.2.2.2 Multicast Send VCC

An LEC sets up a bi-directional point-to-point Multicast Send VCC to the BUS. This VCC is set up using the same process as for Data Direct VCCs. The LEC first sends an LE_ARP and, when it receives the LE_ARP response, initiates signaling to establish this bi-directional VCC (the Multicast Send VCC) to the BUS.

This VCC is used for sending multicast data to the BUS and for sending initial unicast data. The BUS may use the return path on this VCC to send data to the LEC, so this requires the LEC to accept traffic from this VCC.

The LEC must maintain this VCC while participating in the Emulated LAN.

### 2.2.2.2.3 Multicast Forward VCC

After the LEC has set up the Multicast Send VCC, the BUS initiates the signaling for the Multicast Forward VCC to the LEC. This VCC is used for distributing data from the BUS. It can be either a point-to-multipoint VCC or a unidirectional point-to-point VCC. The LEC is required to accept the Multicast Forward VCC regardless of type. A Multicast Forward VCC from the BUS must be established before a LEC participates in the Emulated LAN.

The LEC must attempt to maintain this VCC while participating in the Emulated LAN.

The BUS may forward frames to a LEC on either the Multicast Send VCC or the Multicast Forward VCC. A LEC will not receive duplicate frames forwarded from the BUS on both the Multicast Send and the Multicast Forward VCC, but must be able to accept frames on either VCC.

## 2.3 Functions of the LAN Emulation Service

### 2.3.1 Initialization

This section discusses the Initial state, the Configuration function and the Join phase function. The Initial state refers to the parameters that are configured at the "beginning of time". The initialization is completed after the Join and Initial Registration processes have completed and the connections to the BUS have been established. At this point the LE Client becomes operational (See Figure 8. Initialization, Recovery and Terminating the ELAN.

#### 2.3.1.1 Initial State

In the Initial State there are parameters (such as addresses, Emulated LAN name, max frame size, etc.)[2] that are known to the LE Server and LE Clients, respectively, about themselves, before participating in the Configuration and Join phase functions.

#### 2.3.1.2 LECS Connect Phase

In the LECS Connect Phase, the LE Client establishes a Configuration Direct VCC to the LE Configuration Server.

#### 2.3.1.3 Configuration Phase

This is the phase in which the LE Client discovers the LE Service in preparation for the Join phase. In its simplest form the LE Client may use preconfigured parameters. If the LEC supports the optional LE Configuration protocol, it can use this to discover the LE Service.

#### 2.3.1.4 Join Phase

In the Join phase of ATM LAN emulation initialization, the LE Client establishes its control connections to the LE Server. The Join procedure may have two outcomes: success or failure.

Once the Join phase has successfully completed, the LE Client has been assigned a unique LE Client identifier (LECID), knows the emulated LAN's maximum frame size and its LAN type (Ethernet/IEEE 802.3 or IEEE 802.5), and has established the Control VCC(s) with the LE Server.

#### 2.3.1.5 Initial Registration

After joining, an LE Client may register any number of MAC addresses and/or Route Descriptors. This is in addition to the single MAC address that can be registered as part of the Join phase. Initial Registration allows an LE Client to verify the uniqueness of its local addresses before completing initialization and becoming operational.

#### 2.3.1.6 Connecting to the BUS

In order to establish a connection to the BUS, the LE Client LE_ARPs for the all ones broadcast MAC address and proceeds to set up the connection. The BUS then establishes the Multicast Forward VCC to the LE Client.

---

[2]See Section 5.1 for the complete list.

#### 2.3.1.7 Initialization Phases, Recovery and Termination

The figure below shows the steps in Initialization and paths for termination and recovery from various phases. The conditions for termination and recovery are discussed in sections which follow.

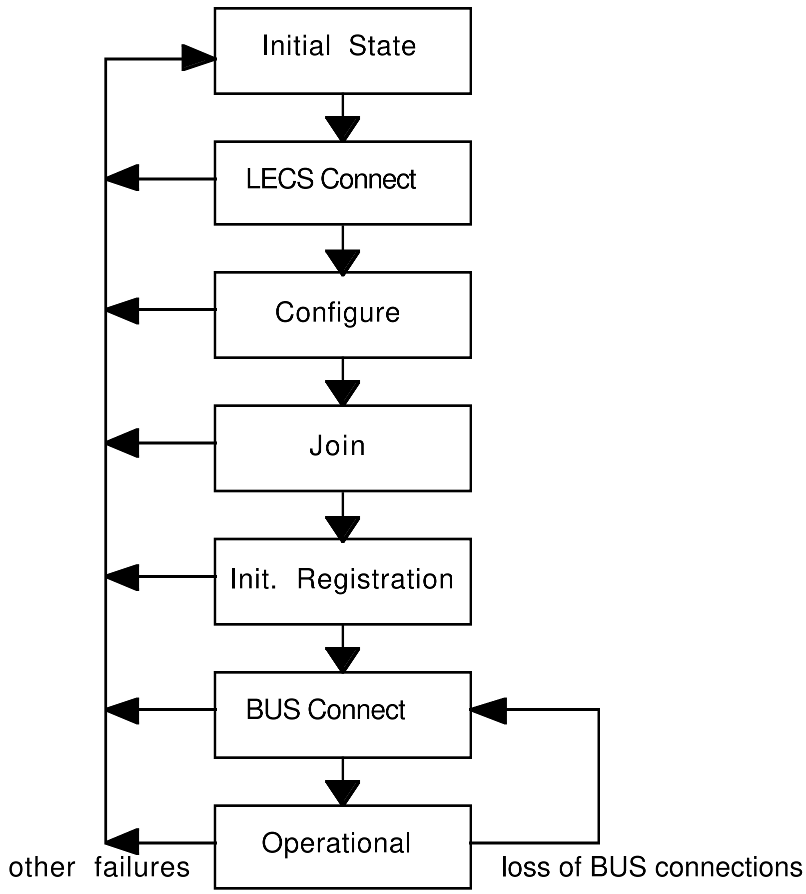



**Figure 8. Initialization, Recovery and Terminating the ELAN**

### 2.3.2 Registration

The address registration function is the mechanism by which Clients provide address information to the LAN Emulation Server. An intelligent LE Server may respond to address resolution requests if LECs register their LAN destinations (defined as MAC addresses or, for source routing IEEE 802.5 LANs only, route descriptors) with the LE Server. The LAN destinations may also be unregistered as the state of the client changes. A Client must either register all LAN destinations for which it is responsible or join as a Proxy.

### 2.3.3 Address Resolution

Address resolution is the procedure by which a client associates a LAN destination with the ATM address of another client or the BUS. Address resolution allows clients to set up data direct VCCs to carry frames.

When a LAN emulation client is presented with a frame for transmission whose LAN destination is unknown to that client, it must issue a LAN emulation address resolution protocol (LE_ARP) request frame to the LES over its control point-to-point VCC.

The LES may either (1) forward this LE_ARP frame to the appropriate client(s) using the control distribute VCC or one or more Control Direct VCCs. Different LES implementations may use different distribution algorithms. If a client responds to a forwarded LE_ARP request with an LE_ARP reply, that reply is also sent and relayed over the control VCCs to the original requester.

Or alternatively (2), instead of forwarding the LE_ARP, the LES may issue an LE_ARP reply on behalf of a client that has registered the requested LAN destination with the LES.

A LAN emulation client must respond to an LE_ARP request that it receives asking for a LAN destination it has registered with the LES, or for which it is a proxy.

Each LE Client maintains a cache of LE_ARP replies, and uses a two-period time-out mechanism to age entries in this cache. The Aging Time period is used for all entries learned from LE_ARP responses whose FLAGS field's Remote Address flag was zero. That is, responses for registered LAN destinations are always timed out with the Aging Time. For aging entries learned from LE_ARP replies with the Remote Address FLAGS bit set to 1 and for entries learned by observing source addresses on Data VCCs, which time-out to use is determined by the state of the LE Client's Topology Change flag. When this flag is SET, such entries are aged using the Forward Delay Time. When this flag is CLEAR, such entries are aged using the Aging Time parameter. The state of this flag may be altered either by local management action or by reception of LE_TOPOLOGY_REQUEST messages (see Section 7.1).

### 2.3.4 Connection Management

In switched virtual connection (SVC) environments the LAN Emulation entities (e.g. LEC, LES and BUS) set up connections between each other using UNI signaling. The connections at a minimum use best effort quality of service. The method of connection set up is summarized as follows:

#### 2.3.4.1 Call Establishment

So that the called party can initiate data transfer on a Data Direct VCC, the ATM layer call establishment procedures are augmented by additional protocol.

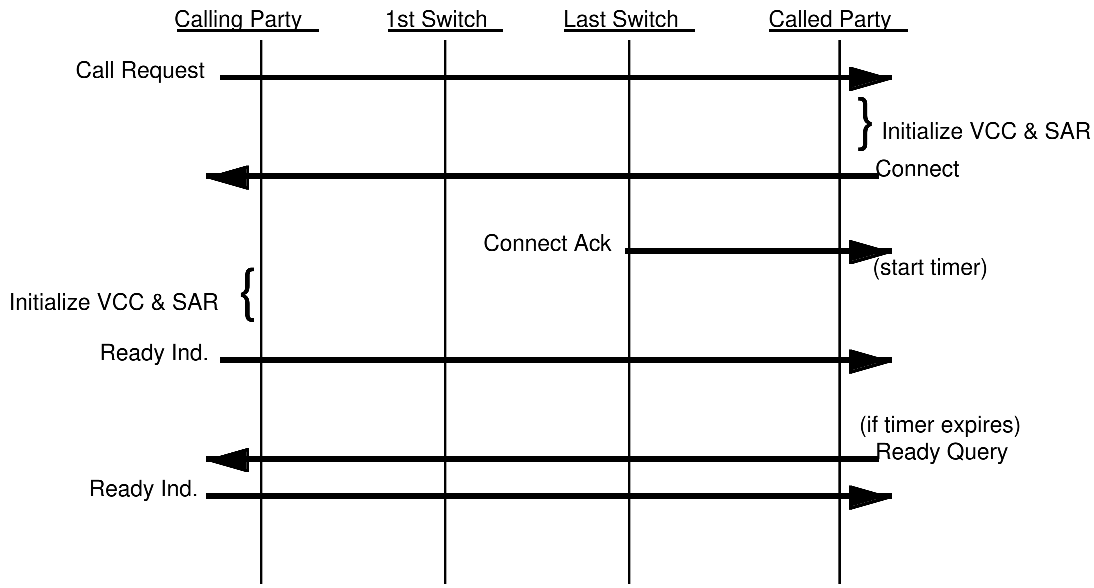



**Figure 9. Call Establishment: Ready Indicate/Query**

In Figure 9, the SETUP, CONNECT, and CONNECT_ACK messages are the SVC call setup messages [5,17]. The READY_IND. and READY_QUERY messages are defined in this specification.

When a call is being set up the called party must not send its CONNECT message until it is ready to receive frames on the new VCC. Thus, the calling party should be able to assume that it can transmit frames after it has received the CONNECT message.

Since the CONNECT_ACK message which is received by the called party may be generated by its local switch and is not an end-to-end indication from the calling party, it may be received by the called party before the calling party has received its CONNECT message. The calling party can only set itself up to receive frames on the VCC after it receives a CONNECT message which indicates the allocation of VPI/VCI numbers. Thus, there is no guarantee for the called party that its initial data will be received by the calling party until it receives some end-to-end indication from the far end.

The calling party must send a READY_IND message as soon as it is ready to receive frames on the newly-established VCC. At that point, the calling party considers call establishment to be complete. The calling party may also send data as soon as it is ready to receive frames on the newly-established VCC. Data may be sent before or after the sending of the READY_IND.

Of course, it is possible that the READY_IND message may be lost. To recover, the called party is responsible for timing the arrival of the READY_IND message. If the timer expires, the called party sends data or a READY_QUERY message on the VCC. Either party shall always respond to receipt of a READY_QUERY message on an active VCC by transmitting a READY_IND message.

#### 2.3.4.2 Tear down & Time Out of VCCs

If a Control Direct VCC or Control Distribute VCC is ever released, an LE Client must always return to the LECS Connect phase of Initialization (See Section 5.2.). If either BUS VCC is lost while the LE Client is participating in the ELAN, the LE Client may return to the BUS Connect Phase or go to the termination phase(See Section 10) and restart (depending on Cause Code, see Figure 8 and relevant text in Section 5.6.1.1.4).

### 2.3.5 Data Transfer

The types of paths used for data frames in the emulated LAN are:

- Data Direct VCCs between individual LAN emulation clients;
- Multicast Send and Multicast Forward VCCs that connect clients to the BUS.

#### 2.3.5.1 Unicast Frames

When a LAN Emulation client has established via the address resolution mechanism that a certain LAN destination corresponds to a certain ATM address, and when that client knows it has a Data Direct VCC to that ATM address, then a frame addressed to that LAN destination must be forwarded via that Data Direct VCC.

If a LAN Emulation client does not know which Data Direct VCC to use for a given unicast LAN destination, or if that Data Direct VCC has not yet been established, it may elect to transmit the frame over the Multicast Send VCC to the Broadcast and Unknown Server. The Broadcast and Unknown Server, in turn, forwards the frame to at least the client for which it is destined. If the LAN destination is unregistered then the frame must be forwarded to at least all proxy clients and may be forwarded to all clients.

On an emulated LAN, the case can arise where a frame can only reach its destination through an IEEE 802.1D transparent bridge, and that bridge does not know the whereabouts of that destination. The only way such a frame can be assured of reaching its destination is for the frame to be transmitted to all of the IEEE 802.1D transparent bridges via the Broadcast and Unknown Server so that they, in turn, can flood that frame to all of their other bridge ports, or at least the ones enabled by the spanning tree protocol. An LE client that chooses not to forward frames to the Broadcast and Unknown Server, therefore, may not be able to reach destinations via transparent bridges (or perhaps other proxy agents.)

#### 2.3.5.2 Multicast Frames

LAN emulation clients may wish to send frames to a multicast MAC address, and/or they may wish to receive frames addressed to a given multicast MAC address. In order to send frames to a multicast MAC address, an LE Client MUST send the frames to the BUS. The address resolution mechanism is used during the initialization process to provide the ATM address of the BUS for multicast and broadcast traffic, and connection management will provide a point-to-point Multicast Send VCC over which to send such frames.

All that is required in order for the LEC to receive frames addressed to a given multicast MAC address, is for the LE Client to connect to the BUS, after which the BUS will try to set up a return path for all broadcast and multicast traffic. When a client connects to the BUS, the BUS will try to establish a Multicast Forward VCC to that client. It is expected that Multicast Forward VCCs will be unidirectional point-to-multipoint VCCs, but they may be implemented as point-to-point VCCs. This decision is left to the LAN Emulation service, not to the client.

An LE Client will receive all flooded unicast frames and all broadcast and multicast frames over either its Multicast Send VCC or its Multicast Forward VCC. Which VCC the BUS uses to forward frames to the LE Client on is at the discretion of the BUS. An LE Client will not, however, receive duplicate frames.

The LAN Emulation Header of any data frame sent from a client to the BUS must either contain the value 0 or the unique LECID value assigned to that client. The BUS is required to preserve the LAN Emulation Header of a relayed frame. Thus, a client can identify and filter frames which it sent by comparing the LECID field to its own LECID value. Note, a transparent bridge LEC can not reliably use the source MAC address to identify its own BUS traffic.

Token-Ring functional addresses are treated just as any other multicast MAC address.

### 2.3.6 Frame Ordering

There may be two paths for unicast frames between a sending LAN emulation client and a receiving client: one via the Broadcast and Unknown Server and one via a data direct VCC between them. For a given LAN destination, a sending client is expected to use only one path at a time, but the choice of paths may change over time. Switching between those paths introduces the possibility that frames may be delivered to the receiving client out of order. Out-of-order delivery of frames between two LAN endpoints is uncharacteristic of LANs, and undesirable in an ATM emulated LAN. The flush protocol is provided to ensure the correct order of delivery of unicast data frames.

**NOTE**: One class of potential LE Clients includes IEEE 802.1D transparent bridges. These bridges are required by the IEEE 802.1D specification to deliver all frames in order. Even a sequence of frames such as:

1. MAC-A -> MAC-B
2. MAC-A -> multicast
3. MAC-A -> MAC-B

should be delivered in order to B. In actuality, a great many legacy transparent bridges guarantee only that unicast frames will be delivered in order; the multicast frame will likely be delivered after frame 3. ATM LAN Emulation can guarantee only the ordering of unicast frames on a single VCC, and only where the transmitting LE Client implements the Flush protocol. The ordering of a mixed string of unicast and multicast frames on different VCCs under ATM LAN Emulation is explicitly not guaranteed.

Sending clients may choose to use the flush message protocol or not. In particular, if a sending client waits for some period of time for address resolution to work before utilizing the Broadcast and Unknown Server, then out-of-order frames can be minimized at the cost of some delay.

Any client receiving a flush request message must respond by sending a flush response message to the original sender.

### 2.3.7 Source Routing Considerations

Source route (SR) bridging is the predominant bridging technology used within IEEE 802.5 token ring networks. The use of source routing does not preclude transparent bridging in these networks. A token ring end station will typically use a combination of source routed and non source routed frames. The process described in this document allows a LEC to operate with both source routing and transparent bridging.

In addition to the Destination Address (DA) field and Source Address (SA) field, a source routed frame contains a Routing Information (RI) field. The RI field contains a control field and a list of route descriptors (RD) that indicate the frame's path through the network. Therefore the information in the RI field determines which SR bridges will forward the frame. The LE Client determines if the frame is to be forwarded by a SR bridge or if the LAN destination is a station on the ELAN.

The LE Client determines if the frame is to be forwarded by an SR bridge or if the LAN destination is a station on the local emulated LAN by examining the RI field. If the LAN destination is accessible through an SR bridge then the LAN destination is the Next Route Descriptor (Next_RD), otherwise, the LAN destination is the frame's Destination Address.

Frames with specifically routed source routing information (an SRF frame) and Unicast destination MAC addresses are sent down Data Direct VCCs following the usual LE_ARP and VCC setup process. Other source routing frames are sent to the BUS.

# 3. LAN Emulation Service Specification

This section specifies the service interfaces between the LAN Emulation entity and the Higher layer, AAL, Connection Management and Layer Management entities. The services are described in an abstract way and do not imply any particular implementation, or any exposed interface.

## 3.1 LE to Higher Layer Services

These services apply only to the LE Client. The higher layer could be LLC (or equivalent) or a bridging relay function. The services provide the capability to exchange user data frames over the LAN Emulation service. Service definitions are compatible with ISO 10039 [7] service architecture and ISO 10038 [4] MAC Bridging standard.

### 3.1.1 LE_UNITDATA.request

This primitive requests the transfer of data from a local entity to a single peer entity, or multiple peer entities in case of group addresses.

The semantics of this primitive are as follows:

```
LE_UNITDATA.request (
        destination_address,
        source_address,
        routing_information {if applicable},
        frame type {if applicable},
        data,
        priority {if applicable},
        )
```

The destination_address parameter may specify either an individual or a group MAC address. The source_address parameter is an individual MAC address that specifies the source. The routing_information parameter applies only to Token Ring emulation. It will be transmitted by the LAN Emulation entity. The data parameter specifies the data unit to be transmitted by the LAN Emulation entity. It should include enough information to derive the length of the data parameter. The frame type is either IEEE 802.3 or Ethernet (if applicable). The priority parameter indicates the priority requested by the higher layer (if applicable).

This primitive is generated by the Higher layer whenever it has data to be transferred to a peer entity or entities. A mapping of MA_UNITDATA.request for endstations and M_UNITDATA.request for bridges to LE_UNITDATA.request will be needed [4,7].

The receipt of this primitive will cause the LAN Emulation entity to effect transfer of the data to the appropriate peer LAN Emulation entity or entities.

### 3.1.2 LE_UNITDATA.indication

This primitive defines the transfer of data from the LAN Emulation entity to the higher layer.

The semantics of this primitive are as follows:

```
LE_UNITDATA.indication (
        destination_address,
        source_address,
        routing_information {if applicable},
        frame type {if applicable}
        data,
        priority {if applicable},
        service class {if applicable},
        )
```

The destination_address parameter may be either an individual or group address as specified by the incoming frame. The source_address parameter is an individual address as specified by the incoming frame. The routing_information (if applicable) and the data parameters specify the data unit as received by the LAN Emulation entity. The frame type is either IEEE 802.3 or Ethernet (if applicable). The priority (if applicable) parameter is derived from the incoming data unit.

The LE_UNITDATA.indication is passed from the LAN Emulation layer to the Higher layer to indicate the arrival of a frame. Such frames are reported only if they are validly formed and received without error.

The effect of receipt of this primitive is unspecified. A mapping of LE_UNITDATA.indicate to MA_UNITDATA.indicate for endstations and M_UNITDATA.indicate for bridges will be needed [4,7].

## 3.2 LAN Emulation to AAL Services

These services apply to the LE Clients and the LE Service. These services provide the capabilities to transfer frames between peer LAN Emulation layers. This specification assumes a null SSCS layer, i.e., the SSCS provides for the mapping of the equivalent primitives of the AAL and CPCS. The common part of AAL-5 makes use of the services provided by the underlying ATM layer.

A LAN Emulation entity includes the following AAL service interfaces, each identified by a distinct SAP-ID. Each LE Client includes the following SAPs:

(1) One or two control SAPs that handle initialization, registration and address resolution.

(2) Two or more data forwarding SAPs.

(3) Zero or one control SAPs that handle configuration.

### 3.2.1 AAL_UNITDATA.request

This primitive provides the capability to transfer the AAL_SDU from one LAN Emulation layer to a peer LAN Emulation layer, or multiple LAN Emulation layers.

The semantics of this primitive are as follows:

```
AAL_UNITDATA.request(
        SAP_ID
        payload
        )
```

The SAP-ID specifies the point-to-point or point-to-multipoint VCC. The payload parameter specifies the data unit to be transmitted by AAL-5.

This primitive is generated by the LAN Emulation entity whenever it has data to be transferred to a peer entity or entities.

The receipt of this primitive will cause the AAL-5 entity to transmit the payload on the specified connection.

### 3.2.2 AAL_UNITDATA.indication

This primitive provides the capability to transfer data from the AAL-5 entity to the LAN Emulation layer.

The semantics of this primitive are as follows:

```
AAL_UNITDATA.indication(
        SAP_ID
        payload
        )
```

The SAP_ID identifies the VCC on which data was received . The payload parameter specifies the data unit as received and reassembled by the AAL-5 entity.

The AAL_UNITDATA.indication is passed from the AAL-5 layer to the LAN Emulation layer to indicate the arrival of a frame. Such frames are reported only if they are completely reassembled and received without error.

The received payload is processed by the LAN Emulation entity or passed to the higher layer.

## 3.3 Connection Management Services

These services apply to the LE Clients and the LE Service.

The conceptual model assumed by the LAN Emulation layer was illustrated in Figure 1: the Connection Management may use either PVCs or SVCs and provides the primitives described below in sections 3.3.2. through 3.3.2.4 (this is labeled as interface (3) in Figure 1). In particular, the Connection Management moc provides a mapping from { near-end ATM address, far-end ATM address, B_LLI } to VPI/VCI.

### 3.3.1 Permanent Virtual Circuits (PVCs)

For PVCs, connection management selects a connection based upon the assumption that each available connection is identified by a near-end ATM address, a far-end ATM address and a B_LLI value.

The method of determining this per-PVC information is outside the scope of this specification but is presumably accomplished by network management or local configuration.

In PVC or mixed SVC/PVC environments, LE Client hosts needs to be configured to know

- Which PVCs are available for LE Client use.
- Which LE Clients are associated with these PVCs.
- Which local and remote ATM addresses are associated with these PVCs.
- Which B-LLI values are associated with these PVCs.

### 3.3.2 Switched Virtual Circuits (SVCs)

Connection management of SVCs uses the UNI Signaling protocol. Connection management performs the following services for SVCs.

#### 3.3.2.1 SETUP

This service provides initial call establishment. It receives an ATM Address and establishes a virtual connection, identified by a SAP-ID. It includes the following UNI calls:

| | |
|---|---|
| CALL_SETUP invoke | (Causes a SETUP message to be sent) |
| CALL_SETUP signal | (Indicates a SETUP message has been received) |
| CALL_CONNECT invoke | (Causes a CONNECT message to be sent) |
| CALL_CONNECT signal | (Indicates a CONNECT message has been received) |
| CALL_CONNECT_ACKNOWLEDGE invoke | (Causes a CONNECT_ACKNOWLEDGE message to be sent) |
| CALL_CONNECT_ACKNOWLEDGE signal | (Indicates a CONNECT ACKNOWLEDGE message has been received) |

#### 3.3.2.2 RELEASE

This service is used to request the network to clear an end-to-end connection (if any), identified by a SAP-ID. It includes the following UNI calls:

| | |
|---|---|
| CALL_RELEASE invoke | (Causes a RELEASE message to be sent) |

| | |
|---|---|
| CALL_RELEASE signal | (Indicates a RELEASE message has been received) |
| CALL_RELEASE_COMPLETE invoke | (Causes a RELEASE COMPLETE message to be sent) |
| CALL_RELEASE_COMPLETE signal | (Indicates a RELEASE COMPLETE message has been received) |

### 3.3.2.3 ADD PARTY

This service provides the capability to add a party to an existing connection. It includes the following UNI calls:

| | |
|---|---|
| CALL_ADD_PARTY invoke | (Causes an ADD PARTY message to be sent) |
| CALL_ADD_PARTY signal | (Indicates an ADD PARTY message has been received) |
| CALL_ADD_PARTY_ACKNOWLEDGE invoke | (Causes an ADD PARTY ACKNOWLEDGE message to be sent) |
| CALL_ADD_PARTY_ACKNOWLEDGE signal | (Indicates an ADD PARTY ACKNOWLEDGE message has been received) |
| CALL_ADD_PARTY_REJECT invoke | (Causes an ADD PARTY REJECT message to be sent) |
| CALL_ADD_PARTY_REJECT signal | (Indicates an ADD PARTY REJECT message has been received) |

### 3.3.2.4 DROP PARTY

This service is used to drop (clear) a party from an existing point-to-multipoint connection. It includes the following UNI calls:

| | |
|---|---|
| CALL_DROP_PARTY invoke | (Causes a DROP PARTY message to be sent) |
| CALL_DROP_PARTY signal | (Indicates a DROP PARTY message has been received) |
| CALL_DROP_PARTY_ACKNOWLEDGE invoke | (Causes a DROP PARTY ACKNOWLEDGE message to be sent) |
| CALL_DROP_PARTY_ACKNOWLEDGE signal | (Indicates a DROP PARTY ACKNOWLEDGE message has been received) |

### 3.3.2.5 Addressable Components

LAN Emulation is composed of five types of connections. The first carries the LAN Emulation control protocol, and the other four types carry Ethernet/IEEE 802.3 and IEEE 802.5 formatted frames. When an end system receives a SETUP message indicating an incoming call, it needs to have sufficient information within the message in order to "bind" the call to the appropriate application and instance. Sufficient information is required to distinguish between the different types of LAN Emulation VCCs such as Ethernet/IEEE 802.3, IEEE 802.5 and Control VCCs and between other applications using ATM such as RFC 1577 “VC-multiplexed style connections.” This is accomplished by the use of several layers of ATM's addressing hierarchy.

LAN Emulation uses a combination of B-LLI values and ATM addresses to distinguish between VCCs of different types and between different LAN Emulation entities. An LE Client and other LAN Emulation components need to examine the calling and called party numbers and the B-LLI values to identify the type of VCC, to identify the LAN emulation components, and to distinguish between LAN Emulation and other ATM applications. ATM addresses that differ only in the selector (SEL) octet are recognized as different ATM addresses in the context of LAN emulation.

Details are covered in the following sections.

#### 3.3.2.6 Connection Establishment Overview

The call and connection procedures in the UNI (Sections 5.5 and 5.6 of the UNI 3.0 and 3.1) apply. The following discussion details the content of the SETUP, CONNECT, RELEASE and RELEASE COMPLETE messages.

In general, if a required information element is not present, the call SHOULD be rejected.

Note, other information elements described in the UNI specification may be also be present.

#### 3.3.2.7 SETUP Message Contents

The SETUP message is formatted by the calling party. It is received by the called party. All information elements provided MUST be transported by the network within the Private UNI. Signaling does not guarantee that all elements are transported by a Public UNI.

For all information element encodings, in UNI 3.1 the spare bit of octet 2 of the information element has been deleted and the Action Indicator in octet 2 is now expanded to 3 bits and renamed to IE Action Indicator. A UNI 3.0 implementation SHOULD insure that the spare bit of the Action Indicator is set to 0 to insure that future use of information element encodings work as expected under UNI 3.1. Information elements with the IE Action Indicator field equal to binary 100 are reserved in UNI 3.1 and MUST be rejected.

##### 3.3.2.7.1 AAL Parameters

This information element[3] MUST be used in the SETUP message.

**Table 2. AAL Parameters**

[3]Reference: Section 5.4.5.5 of [5, 17]

| Field | Value |
|---|---|
| AAL Type | 5 (for AAL5) |
| Forward Maximum CPCS-SDU Size | Control: 1516 octets for LE Configuration Direct VCCs, LE Control Direct and LE Control Distribute VCCs.<br>Data: one of the standard AAL-5 SDU Max. octet values from Table 26. |
| Backward Maximum CPCS-SDU Size | Same as Forward Maximum CPCS-SDU Size |
| Mode[4] | 1 (Message mode) |
| SSCS Type | 0 (Null SSCS) |

Note: For implementations based on UNI 3.0, the Mode field MUST be present following the Backward Maximum CPCS-SDU Size field. This field MUST NOT present for UNI 3.1. The Mode value MUST be 1 (this is for message mode). UNI 3.0 implementations may have a maximum number of 20 bytes in the AAL Parameters information element. UNI 3.1 implementations may have up to 21 bytes in the AAL Parameters information element.

#### 3.3.2.7.2 ATM User Cell Rate /ATM Traffic Descriptor

The term “ATM User Cell Rate” is used in UNI 3.0, whereas the term “ATM Traffic Descriptor” is used in UNI 3.1.

This information element[5] MUST be present. Only the parameters needed are included. The ones that are not needed are those that specify the peak rate with CLP=0 and the sustainable cell rate and maximum burst size parameters. The parameters shown below are for Best Effort connections. It is possible that the calling end system may wish to create a connection with a different cell rate.[6]

When the called party rejects a call based on User-Cell Rate parameters the following actions MAY be taken:

> If, in response to a SETUP message, a calling endsystem receives a RELEASE COMPLETE message, or a CALL PROCEEDING message followed by a RELEASE message, with Cause code (UNI 3.0 Cause #51, User cell rate unavailable, or UNI 3.1 Cause #37, User Cell Rate not available), it MAY examine the diagnostic field of the Cause IE and re-attempt the call after selecting smaller values for the parameter(s) indicated. If the RELEASE COMPLETE or RELEASE message is received with Cause code (UNI 3.0 or 3.1 Cause #73, Unsupported combination of traffic parameter), it MAY try other combinations from the ATM Forum UNI document (for example Tables 5-7 and 5-8 of UNI 3.0).
>
> The called endsystem SHOULD examine the ATM User Cell Rate /ATM traffic descriptor IE in the SETUP message. If it is unable to process cells at the Forward PCR indicated, it SHOULD clear the call (UNI 3.0 Cause #51, User cell rate unavailable, or UNI 3.1 Cause #37).

---

[4]This field encoding MUST be present for UNI 3.0. The field MUST NOT be present in UNI 3.1.

[5]Reference: Section 5.4.5.6 of [5, 17]

[6]An end system should not reject a connection just because it is not specified at line rate and Best Effort. It may reject it because it can not support the requested connection.

If this information element is not present, the called end system MUST release the call by sending a RELEASE message with cause #96 (mandatory information element is missing) and MUST indicate the Information Element Identifier for ATM traffic descriptor (0x59, binary 01011001).

**Table 3. ATM User Cell Rate/ATM Traffic Descriptor**

| Field | Value |
|---|---|
| Forward Peak Cell Rate (CLP=0+1) | line rate in cells per second |
| Backward Peak Cell Rate (CLP=0+1) | line rate in cells per second |
| Best Effort Indicator | 0xBE |

### 3.3.2.7.3 Broadband Bearer Capability

This information element[7] MUST be used in the SETUP message sent by the calling party. It is used to indicate what kind of network connection is desired. This specification recommends using Service Type X. However as an alternative, Service Type C could be used instead. When the calling party receives an indication from the network that Service Type X is not supported via a RELEASE or RELEASE COMPLETE message with Cause 57 (Bearer Capability not Authorized) or Cause 58 (Bearer Capability Not Presently Available), another Broadband Bearer Capability MAY be tried.

If this information element is not present, the called end system MUST release the call by sending a RELEASE message with cause #96 (mandatory information element is missing) and MUST indicate the Information Element Identifier for Broadband bearer capability (0x5E, binary 01011110).

**Table 4. Broadband Bearer Capability**

| Field | Value |
|---|---|
| Bearer Class | 16 (BCOB-X) |
| Traffic Type | 0 (No Indication) |
| Timing Requirements | 0 (No Indication) |
| Susceptibility to clipping | 0 (Not susceptible to clipping) |
| User plane connection configuration | 0 for point-to-point, 1 for point-to-multipoint |

### 3.3.2.7.4 Broadband Low Layer Information

This information element[8] MUST be used in the SETUP message. It is used to indicate the protocol type carried in the  connection and the purpose of the connection (Client-to-Client, Client-to-LECS, Client-to-LES, Client-to-BUS). LAN Emulation supports three unique frame formats. The two data frame formats are indicated as "ATM Forum LAN Emulation ... Ethernet/IEEE 802.3" and “ATM Forum LAN Emulation ... IEEE 802.5."  The “...” will be either “Data Direct” (for Data Direct VCCs) or “Multicast Send” or “Multicast Forward” (for Multicast

---

[7]Reference: Section 5.4.5.7 of [5, 17]

[8]Reference: Section 5.4.5.9 of [5, 17]

VCCs). The control frame format is indicated as "ATM Forum LAN Emulation Control." These encodings all use the ATM Forum's allocated 24-bit OUI. Note that an “ATM Forum LAN Emulation Ethernet/IEEE 802.3” format VCC may carry both IEEE 802.3 and Ethernet frames.

**Table 5. Broadband Low Layer Information**

| **Field** | **Value** |
|---|---|
| User information layer 3 protocol | 11 (ISO/IEC TR 9577) |
| ISO/IEC TR 9577 Initial Protocol Identifier | 64 (SNAP Identifier - 0x80, spread over 2 octets, left justified) |
| Continued from previous Octet | Continued (Ext bit is set to 1) |
| SNAP ID | 0x80 (indicates SNAP and PID follow) |
| SNAP Organizational Unit Identifier | 00 A0 3E (ATM Forum OUI) |
| PID | **0001** for LE Configuration Direct VCC, Control Direct VCC and Control Distribute VCC<br>**0002** for Ethernet/IEEE 802.3 LE Data Direct VCC<br>**0003** for IEEE 802.5 LE Data Direct VCC<br>**0004** for Ethernet/IEEE 802.3 LE Multicast Send VCC and Multicast Forward VCC<br>**0005** for IEEE 802.5 Multicast Send VCC and Multicast Forward VCC |

#### 3.3.2.7.5 QoS Parameter

This information element[9] MUST be used in the SETUP message sent by the calling party. In relation to the Best Effort requested connection, the Service Class type 0 is specified. An LEC MUST accept Service Class 0 and SHOULD use Class 0.

When the called party rejects a call based on requested QoS parameters, the following action may be taken:

> If, in response to a SETUP message, a calling endsystem receives a RELEASE COMPLETE message, or a CALL PROCEEDING message followed by a RELEASE message, with cause #49, Quality of Service unavailable, it MAY re-attempt the call after selecting another QoS class.

**Table 6.QoS Parameter**

| **Field** | **Value** |
|---|---|
| QoS Class Forward | 0 (for Class 0) |
| QoS Class Backward | 0 (for Class 0) |

[9]Reference: Section 5.4.5.18 of [5, 17]

#### 3.3.2.7.6 Called Party Number

This information element[10] MUST be used in the SETUP message sent by the calling party. All private ATM addresses in ATM LAN Emulation control protocols use the ATM Forum UNI Private Address Format of 20 octets.

The Called Party (and Calling Party) information elements can be more explicitly shown. For example:

**Called Party Number (Private Addresses[11])**

| | |
|---|---|
| Coding Standard | 0 |
| IE Instruction Field | 00 |
| Length of Called Party | 20 octets |
| Type of Number | 0 (for Unknown) |
| Addressing/Numbering Plan | 2 (binary 0010)<br>(for ATM endsystem address) |

Coding of public ATM addresses and mapping from private to public ATM addresses are respectively described in the UNI specification.

#### 3.3.2.7.7 Calling Party Number

This information element[12] MUST be used in the SETUP message sent by the calling party for LAN Emulation and the network will transport it across the Private UNI.

#### 3.3.2.7.8 Connection Identifier

This information element indicates what VPI/VCI values have been assigned by the network for the connection. This information element MUST be present in the SETUP message when received by the called party.

### 3.3.2.8 CONNECT Message Contents

The CONNECT message is formatted by the called party. It is received by the calling party. It is used primarily to confirm the connection and to ensure that the MTU sizes are in agreement with the previously established MTU sizes.

#### 3.3.2.8.1 AAL Parameters

The called party MAY include the AAL Parameter information element in the CONNECT message.

If this information element is returned by the network, it SHOULD be checked to ensure that the parameters are unchanged or values that can be accepted. If not, the call SHOULD be released

#### 3.3.2.8.2 Broadband Low Layer Information

The called party MAY include a B-LLI information element in the CONNECT message.

---

[10]Reference: Section 5.4.5.11 of [5, 17]

[11]For UNI 3.0 addresses, the format is referred to as "OSI NSAP." For UNI 3.1 addresses, the format is referred to as "ATM Endsystem Address."

[12]Reference: Section 5.4.5.13 of [5, 17]

If this information element is returned from the called party, it MUST be unchanged from the coding requested in the SETUP message. If not, the call SHOULD be released.

#### 3.3.2.8.3 Connection Identifier

This information element indicates what VPI/VCI values have been assigned by the network for the connection.

### 3.3.2.9 RELEASE and RELEASE COMPLETE Message Contents

LAN Emulation defines the use of the following Cause Codes to indicate why a VCC is being released. The following encoding is used when explicitly releasing a VCC:

| | | |
|---|---|---|
| Coding Standard | 00 | (ITU-TS) |
| IE Instruction Field | 0 | (not significant) |
| Location | 0 | (User) |
| Cause Value | 16[13]<br>31 | Normal call clearing<br>Normal, Unspecified |

Other Cause Codes as specified in the UNI may be used under appropriate circumstances.

## 3.4 LAN Emulation to Layer Management Services

These services enable initialization and control of the LAN Emulation entities. These services differ between the LE Clients and the LE Service. A complete list of Initial State Parameters for LE Client and LE Server are listed in Section 5.

### 3.4.1 LE Client Layer Management service

#### 3.4.1.1 LM_LEC_INITIALIZE.request

This primitive is used to configure the LE Client and cause it to join the emulated LAN.

The semantics of this primitive are as follows:

**LM_LEC_INITIALIZE.request**(
LEC_ATM_address,
Server_ATM_address,
MAC Address[14]
Configure_mode,
LEC_proxy_class,
Requested_LAN_type,
Requested_max_frame_size,
Requested_ELAN_name,
Join_time-out )

[13]This code applies to UNI 3.1 only.

[14]Need not be present.

The LEC_ATM_address parameter specifies the ATM address of the LE Client. The Server_ATM_address parameter specifies the ATM address of the LE Configuration Server (LECS) or the LE Server (LES) dependent on the Configure_mode parameter and may be unspecified if the Configure_mode parameter is “autoconfigure”. The Configure_mode parameter may be “autoconfigure” or “manual” and it specifies if the LEC will auto configure (i.e., attach to an LECS to discover the LES ATM address), or if it is manually configured in which case the Server_ATM_address specifies the ATM address of the LES. The LEC_proxy_class is specified as either “proxy” or “non-proxy”. The Requested_LAN_type parameter may be “unspecified”, “IEEE802.3/Ethernet”, or “IEEE802.5 Token_ring”. The Requested_max_frame_size parameter may be “unspecified”, “1516 octets”, “4544 octets”, “9234 octets”, or “18190 octets”. The Requested_ELAN_name may be “unspecified” or the name of the ELAN the client is to join. The Join_time-out is the minimum time that the LEC should wait for a response to its Configuration request or its Join request.

This primitive is generated by local management to cause the LEC to join the ELAN.

The receipt of this primitive will cause the LE Client to stop operation (release all VCCs), to reset all its internal information, to update the specified parameters, and enter the INITIAL state. If the Configure_mode parameter is “autoconfigure”, then the LEC will start the Configuration phase (5.3), and if it is “manual” the LEC will start the Join phase (5.4).

### 3.4.1.2 LM_LEC_INITIALIZE.confirm

This primitive is used to confirm the Initialize request and indicate if the LEC’s user service interface is functional.

The semantics of this primitive are as follows:

**LM_LEC_INITIALIZE.confirm** (
Join_status )

The Join_status parameter specifies success or the reason for failure of the Initialize request.

This primitive is generated by the LEC to indicate the completion of the Initialization request. When the join status is “success,” the LEC service user may start sending and receiving user data frames.

### 3.4.1.3 LM_LEC_LD_CONTROL.request

This primitive maintains the LE Client’s database of associated MAC addresses and route designators (i.e., the list of addresses the LEC represents and for which it should receive frames). This primitive associates an action to an element of the database allowing the addition or removal of an element or the clearing of all elements of the same type and class.

The semantics of this primitive are as follows:

**LM_LEC_LD_CONTROL.request**(
LD_action,
LD_type,
LD_proxy_class,
LAN_destination_address )

The LD_action parameter is: add_address, delete_address, or clear_list. The LD_type parameter specifies multicast, unicast, or route_designator. The LD_proxy_class parameter specifies either "proxy" or "local". The LAN_destination_address specifies the address to add or delete. If the action is clear_list, then the LEC deletes all entries in the database whose attributes match the specified LD_type and LD_proxy_class. If the local address is added before initialization, registration will occur.

The LAN_destination_address for a multicast or unicast LD_type is a 48 bit MAC address. The LAN_destination for a route_designator LD_type is a 16 bit route designator consisting of a 12 bit segment_id and a 4 bit bridge_number. Route designators apply to IEEE802.5 Token Ring emulation and are not required for IEEE802.3/Ethernet emulation .

This primitive is generated by Layer Management to modify the LE Client's address database.

Receipt of this primitive will cause the LE Client to add or delete the specified address and register the change with the LES as appropriate.

#### 3.4.1.4 LM_LEC_GET.request

This primitive is used by Layer Management to read the values of operational parameters/MIB variables.

The semantics of this primitive are as follows

**LM_LEC_GET.request**(
Attribute_id )

The Attribute_id parameter specifies the attribute. The attribute may indicate any single variable/ group of variables available to be read by management.

This primitive is generated by Layer Management to read the LE Client's MIB.

Receipt of this primitive will cause the LE Client to generate a LM_LEC_STATUS.indication with the requested information.

#### 3.4.1.5 LM_LEC_SET.request

This primitive controls the operation of the LE Client by modifying its operational parameters.

The semantics of this primitive are as follows:

**LM_LEC_SET.request**(
Attribute_id,
Attribute_value )

The Attribute_id parameter specifies the attribute and the Attribute_value specifies the new value. The Attribute may be any operational parameter or MIB variable that may be modified by management (see 4.1.1).

This primitive is generated by Layer Management to cause the LE Client to change operation.

Receipt of this primitive will cause the LE Client to change the specified attribute's value.

#### 3.4.1.6 LM_LEC_STATUS.indication

This primitive is used by the LE Client to report status to the Layer Management entity.

The semantics of this primitive are as follows:

**LM_LEC_STATUS.indication**(
status_report )

The status_report parameter shall specify appropriate status, including statistics, error counts, etc. This primitive is generated by the LE Client when any of the following events occur: LE Service Up, LE Service Down, LM_LEC_GET.request, or LM_LEC_SET.request.

### 3.4.1.7 LM_LEC_TOPOLOGY.request

This primitive controls the generation of LE Topology Request control frames to the LES and is used to notify other LE Clients of the topology change status. This primitive is invoked, for example, by an IEEE802.1D transparent bridge when it sends a configuration BPDU.

The semantics of this primitive are as follows:

**LM_LEC_TOPOLOGY.request**(
Topology_change_status )

The Topology_change_status parameter is either “topology_change” or “no_topology_change”.

Receipt of this primitive will cause the LE Client to send a Topology Request control frame to the LES.

# 4. LAN Emulation Frame Formats

## 4.1 LE Data Frame

Two data packet formats are used by this specification. One of the two formats MUST be used by an LE Client that conforms to this specification. The first is based on ISO 8802.3/CSMA-CD (IEEE 802.3) and has the format shown in Table 7 below. The minimum LAN Emulation AAL5 SDU length for IEEE 802.3/Ethernet format data frames is 62 octets.

**Table 7. LAN Emulation Data Frame Format for IEEE 802.3/Ethernet Frames**

| | | |
|---|---|---|
| 0 | LE HEADER | DESTINATION ADDR |
| 4 | DESTINATION ADDRESS | |
| 8 | SOURCE ADDRESS | |
| 12 | SOURCE ADDR | TYPE/LENGTH |
| 16 and on | INFO | |

The length of the LAN Emulation header is two octets, and it also contains either the LECID value of the sending LE Client or X'0000'. The LAN frame check sequence, FCS, MUST NOT be included.

The second packet format is based on ISO 8802.5 (IEEE 802.5, or Token Ring) and has the format shown in Table 8 below. The minimum LAN Emulation AAL5 SDU length for IEEE 802.5 format data frames is 16 octets.

**Table 8. LAN Emulation Data Frame Format for IEEE 802.5 Frames**

| | | | | |
|---|---|---|---|---|
| 0 | LE HEADER | | AC PAD | FC |
| 4 | DESTINATION ADDRESS | | | |
| 8 | DESTINATION ADDR | | SOURCE ADDRESS | |
| 12 | SOURCE ADDR | | | |
| 16 UP TO 46 | ROUTING INFORMATION FIELD | | | |
| | INFO | | | |

The length of the LAN Emulation header is two octets, and it also contains either the LECID value of the sending LE Client or X'0000'. The LAN frame check sequence, FCS, MUST NOT be included.

The AC PAD octet is not used in LAN Emulation. This MAY be set to any value on transmit and SHOULD be ignored on receipt.

LAN Emulation only allows LLC frames to be sent, therefore the FC octet MUST be of the form '0100YYY' binary, where 'YYY' is the priority set by the source LLC and MAY be recovered by the receiving LLC. This is described in *IEEE 802.5 - 1992*.

### 4.1.1 Encoding of Ethernet/IEEE 802.3 Type/Length Fields

ATM LAN Emulation supports both IEEE 802.2 (LLC) data frames (see [8]) and DIX Ethernet EtherType frames (see [15]). Because an ATM Emulated LAN allows data frames larger than those allowed on legacy LANs, an LE Client MUST use the following rules to encode the Type/Length fields of data frames:

1. DIX Ethernet EtherType frames MUST be encoded by placing the EtherType field in the TYPE/LENGTH field. Data following the EtherType field follows the TYPE/LENGTH field immediately.
2. LLC data frames whose total length, including the LLC field and data, but not included padding required to meet the minimum data frame length, (and not including the Frame Check Sequence, which is not used in ATM LAN Emulation), is less than 1536 (X"0600") MUST be encoded by placing that length in the TYPE/LENGTH field. The LLC field follows the TYPE/LENGTH field immediately.
3. LLC data frames longer than this maximum MUST be encoded by placing the value 0 in the TYPE/LENGTH field. The LLC data frame follows the TYPE/LENGTH field immediately.

When decoding data frames, the following rules MAY be used by an LE Client:

1. If the TYPE/LENGTH field is 1536 (X"0600") or greater, the frame is DIX Ethernet encoded. The TYPE/LENGTH field is the EtherType, and the data follows. The length of the data may be obtained from the AAL-5 trailer, by subtracting the length of the frame through the TYPE/LENGTH field (16). On short frames, this length has a minimum value (46), so padding octets added to meet the minimum frame size cannot be identified at this level of decoding.

2. If the TYPE/LENGTH field is less than 1536 (X“0600”), the frame is an IEEE 802.2 LLC frame. The LLC field immediately follows the TYPE/LENGTH field.
   a. If the TYPE/LENGTH field is non-zero, it indicates the length of the data, starting with the LLC octets. Since this length does not include padding to meet the minimum data frame length requirement, such padding can be identified.
   b. If the TYPE/LENGTH field is zero, then the length of the LLC data field may be obtained from the AAL-5 trailer, by subtracting the length of the frame through the TYPE/LENGTH field (16)

## 4.2 LE Control Frames

All LAN Emulation control frames, except for READY_IND and READY_QUERY, use the format described in Table 9. See Table 23 for the format of READY_IND and READY_QUERY frames.

**Table 9. Control Frame**

| | | | | |
|---|---|---|---|---|
| 0 | MARKER = X”FF00” | | PROTOCOL = X”01” | VERSION = X”01” |
| 4 | OP-CODE | | STATUS | |
| 8 | TRANSACTION-ID | | | |
| 12 | REQUESTER-LECID | | FLAGS | |
| 16 | SOURCE-LAN-DESTINATION | | | |
| 24 | TARGET-LAN-DESTINATION | | | |
| 32 | SOURCE-ATM-ADDRESS | | | |
| 52 | LAN-TYPE | MAXIMUM-FRAME-SIZE | NUMBER-TLVS | ELAN-NAME-SIZE |
| 56 | TARGET-ATM-ADDRESS | | | |
| 76 | ELAN-NAME | | | |
| 108 | TLVs BEGIN | | | |

The fields common to all frame formats are described in Table 10.

In the two fields in a control frame where a LAN Destination is specified, the 8-octet format described in Table 11 MUST be used. The tag value “not present” MUST only be used where explicitly allowed for individual frame formats and protocols. MAC addresses and Route Designators are in their “natural” bit/octet order, LSB for Ethernet/IEEE 802.3 or MSB for IEEE 802.5, according to the emulated LAN type.

Where the emulated LAN type is undetermined, Ethernet/IEEE 802.3 bit ordering MUST be used. There are only three cases where the LAN type may be undetermined:

1. LE_CONFIGURE_REQUEST, LAN-TYPE=0
2. LE_CONFIGURE_RESPONSE, LAN-TYPE=0
3. LE_JOIN_REQUEST, LAN-TYPE=0

**Table 10. Control Frame Header Format**

| Offset | Size | Name | Function |
|---|---|---|---|
| 0 | 2 | MARKER | Control Frame = X"FF00" |
| 2 | 1 | PROTOCOL | ATM LAN Emulation protocol = X"01" |
| 3 | 1 | VERSION | ATM LAN Emulation protocol version = X"01" |
| 4 | 2 | OP-CODE | Control frame type: See Table 14 for a list of values. |
| 6 | 2 | STATUS | Always X"0000" in requests. See **Table 13** for a list of values. |
| 8 | 4 | TRANSACTION-ID | Arbitrary value supplied by the requester and returned by the responder to allow the receiver to discriminate between different responses. |
| 12 | 2 | REQUESTER-LECID | LECID of LE Client sending the request (X"0000" if unknown). |
| 14 | 2 | FLAGS | Bit flags. See Table 12 for a list of values. |
| 16 | 92 | Meaning of remainder of fields depends on OP-CODE. | |

**Table 11. LAN Destination Field Format**

| Offset | Size | Name | Function |
|---|---|---|---|
| 0 | 2 | TAG | X"0000" = not present<br>X"0001" = MAC address<br>X"0002" = Route Descriptor |
| 2 | 6 | MAC Address | 6-octet MAC address if MAC address specified. |
| 2 | 4 | RESERVED | 0, if Route Descriptor specified. |
| 6 | 2 | Route Descriptor | If Route Descriptor specified. |

The Route Descriptor has the coding form 'LLLLLLLLLLLLBBBB' binary, where 'LLLLLLLLLLLL' is the LAN_ID and 'BBBB' is the Bridge number. This is also described in Section 12.4.2.2.

**Table 12. Control Frame FLAGS Values**

| Hex | Name | Use |
|---|---|---|
| X"0001" | Remote Address | LE_ARP_RESPONSE |

| | | |
|---|---|---|
| X“0080” | Proxy Flag | LE_JOIN_REQUEST |
| X“0100” | Topology Change | LE_TOPOLOGY_REQUEST |

**Table 13. Control Frame STATUS Values**

| Code (dec) | Name | Meaning | Responses |
|---|---|---|---|
| 0 | Success | Successful response | All responses |
| 1 | Version Not Supported | VERSION field of request contains a value higher than that supported by the responder. | All responses |
| 2 | Invalid request parameters | The parameters given are incompatible with the ELAN. | All responses |
| 4 | Duplicate LAN Destination registration | SOURCE-LAN-DESTINATION duplicates a previously-registered LAN Destination | Join or Register |
| 5 | Duplicate ATM address | SOURCE-ATM-ADDRESS duplicates a previously-registered ATM address. | Join or Register |
| 6 | Insufficient resources to grant request | Responder is unable to grant request for reasons such as insufficient table space or ability to establish VCCs. | Configure, Join or Register |
| 7 | Access denied | Request denied for security reasons. | Configure or Join |
| 8 | Invalid REQUESTOR-ID | LECID field is not zero (Configure or Join) or is not LE Client’s LECID (others). | Configure, Join, Register, Unregister, ARP |
| 9 | Invalid LAN Destination | LAN Destination is a multicast address or, on an Ethernet/802.3 ELAN, a Route Descriptor. | Configure, Join, Register, ARP, Flush |
| 10 | Invalid ATM Address | Source or Target ATM Address not in a recognizable format. | Configure, Join, Register, ARP, Flush |
| 20 | No Configuration | LE Client is not recognized. | Configure |
| 21 | LE_CONFIGURE Error | Parameters supplied give conflicting answers. May also be used to refuse service without giving a specific reason. | Configure |
| 22 | Insufficient Information | LE Client has not provided sufficient information to allow the LECS to assign it to a specific ELAN | Configure |

## 4.3 Control Frame Usage

### 4.3.1 Discarding Invalid Frames

256 For the purposes of determining validity, a control frame is a frame which has the value
257 X“FF00” in its MARKER field, X“01” in its PROTOCOL field, X“01” in its VERSION field
258 and a value from Table 14. OP-CODE Summary

A data frame is a frame that has a value between X“0000” and X“FEFF” inclusive in its LE-HEADER field. A LAN Emulation component that receives any frame that is not, by these definitions, either a data frame or a control frame, MUST discard that frame[15], except that

1. An LE Server MAY relay such a frame from a Control Direct VCC to a Control Distribute or Control Direct VCC, and

2. A BUS MAY relay such a frame from a Multicast Send VCC to a Multicast Forward VCC.

### 4.3.2 Request/Response Field Usage

In the tables describing the contents and usage of control frames, some fields are explicitly defined as having different values in request and response frames. All fields whose descriptions are not so differentiated MUST be copied verbatim from the request frame to the response frame by any LAN Emulation component that responds to a request.

### 4.3.3 Sender/Receiver Field Usage

In the tables describing control frame field usage, the terms sender and receiver refer to the component performing the transmission or reception of the AAL-5 SDU in question.

### 4.3.4 Response Frame Routing

An LE Server MUST direct any response frame (control frame with X“01” in the most-significant octet of its OP-CODE field) it receives to at least that LE Client (if any) whose LECID matches the response frame’s REQUESTER-LECID field.

### 4.3.5 Agreement of Response Frame Destinations

An LE Server MAY discard an LE_ARP or FLUSH response frame without re-transmitting it if the SOURCE-ATM-ADDRESS field and the REQUESTER-LECID field indicate two different LE Clients, or indicate no operational LE Client in the emulated LAN.

### 4.3.6 Request Frame Source Identification

An LE Client MUST put one of its ATM addresses in the SOURCE-ATM-ADDRESS field of every request frame (control frame with X“00” in the most-significant octet of its OP-CODE field) that it transmits. It MUST put its LECID in every request frame except for those that require a 0 LECID (i.e. LE_CONFIGURE_REQUEST and LE_JOIN_REQUEST).

### 4.3.7 Response Frame Filtering

An LE Client MUST discard any LE_CONFIGURE_RESPONSE, LE_JOIN_RESPONSE, or LE_FLUSH_RESPONSE frame in which neither the SOURCE-ATM-ADDRESS nor the REQUESTOR-LECID identify that LE Client. It MAY discard any response frame in which

---

[15] Discarding these control frames should not cause an LE Client to generate a response frame.

either of those two fields does not identify the LE Client, except that only the SOURCE-ATM-ADDRESS field can be used on LE_CONFIGURE_RESPONSE and LE_JOIN_RESPONSE frames.

### 4.3.8 Summary of OP-CODEs

**Table 14. OP-CODE Summary**

| OP-CODE Value | OP-CODE Function |
|---|---|
| X"0001" | LE_CONFIGURE_REQUEST |
| X"0101" | LE_CONFIGURE_RESPONSE |
| X"0002" | LE_JOIN_REQUEST |
| X"0102" | LE_JOIN_RESPONSE |
| X"0003" | READY_QUERY |
| X"0103" | READY_IND |
| X"0004" | LE_REGISTER_REQUEST |
| X"0104" | LE_REGISTER_RESPONSE |
| X"0005" | LE_UNREGISTER_REQUEST |
| X"0105" | LE_UNREGISTER_RESPONSE |
| X"0006" | LE_ARP_REQUEST |
| X"0106" | LE_ARP_RESPONSE |
| X"0007" | LE_FLUSH_REQUEST |
| X"0107" | LE_FLUSH_RESPONSE |
| X"0008" | LE_NARP_REQUEST |
| X"0108" | Undefined |
| X"0009" | LE_TOPOLOGY_REQUEST |
| X"0109" | Undefined |

## 4.4 Byte and Bit Order

The convention used to describe LAN Emulation frame formats in this document is to express numbers in decimal and to picture data in "big-endian" order. That is, fields are described left to right, with the most significant octet on the left and the least significant octet on the right.

The order of transmission to the AAL5 layer of the header and data described in this document is resolved to the octet level. Whenever a diagram shows a group of octets, the order of transmission of those octets is the normal order in which they are read in English. For example, in the following diagram the octets are transmitted in the order they are numbered.

| 0<br>0 1 2 3 4 5 6 7 | 1<br>8 9 0 1 2 3 4 5 | 2<br>6 7 8 9 0 1 2 3 | 3<br>4 5 6 7 8 9 0 1 |
|---|---|---|---|
| 1 | 2 | 3 | 4 |
| 5 | 6 | 7 | 8 |
| 9 | 10 | 11 | 12 |

**Transmission Order of Bytes**

Whenever an octet represents a numeric quantity the left most bit in the diagram is the high order or most significant bit. That is, the bit labeled 0 is the most significant bit. For example, the following diagram represents the value 170 (decimal).

| 0 1 2 3 4 5 6 7 |
|---|
| 1 0 1 0 1 0 1 0 |

**Significance of Bits**

Similarly, whenever a multi-octet field represents a numeric quantity the left most bit of the whole field is the most significant bit. When a multi-octet quantity is transmitted the most significant octet is transmitted first. Hexadecimal values are indicated by the notation X"", e.g. X"0101" = decimal 257.

## 4.5 Media Format of MAC Addresses

An IEEE 48 bit MAC address is a 48 bit binary number. The I/G bit indicates if the address is a unicast (I/G=0) or multicast (I/G=1) address. Conventional Ethernet and Token Ring LANs differ in that Ethernet transmits the lsb (least significant bit) of each octet first and Token Ring transmits the msb (most significant bit) of each octet first. Because of this difference, the format of the 48 bit MAC address in memory is different for these two media access methods. The following tables show the formats used for 48 bit MAC addresses in Ethernet and Token Ring LAN Emulation. The format used is determined by the LAN Type parameter. The selected format is used for both LE Data Frames and LE Control Frames.

**Table 15. Formats for IEEE 48 bit MAC Address Based on LAN Type**

| Ethernet: | Data Bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | **msb** | **d6** | **d5** | **d4** | **d3** | **d2** | **d1** | **lsb** |
| **Octet 1** | b7 | b6 | b5 | b4 | b3 | b2 | b1 | I/G |
| **Octet 2** | b15 | b14 | b13 | b12 | b11 | b10 | b9 | b8 |
| **Octet 3** | b23 | b22 | b21 | b20 | b19 | b18 | b17 | b16 |
| **Octet 4** | b31 | b30 | b29 | b28 | b27 | b26 | b25 | b24 |
| **Octet 5** | b39 | b38 | b37 | b36 | b35 | b34 | b33 | b32 |
| **Octet 6** | b47 | b46 | b45 | b44 | b43 | b42 | b41 | b40 |

| Token Ring: | Data Bits | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | **msb** | **d6** | **d5** | **d4** | **d3** | **d2** | **d1** | **lsb** |
| **Octet 1** | I/G | b1 | b2 | b3 | b4 | b5 | b6 | b7 |
| **Octet 2** | b8 | b9 | b10 | b11 | b12 | b13 | b14 | b15 |
| **Octet 3** | b16 | b17 | b18 | b19 | b20 | b21 | b22 | b23 |
| **Octet 4** | b24 | b25 | b26 | b27 | b28 | b29 | b30 | b31 |
| **Octet 5** | b32 | b33 | b34 | b35 | b36 | b37 | b38 | b39 |
| **Octet 6** | b40 | b41 | b42 | b43 | b44 | b45 | b46 | b47 |

# 5. Initialization Protocols, Procedures and Frame Formats

The initialization of an LE Client is divided into an Initial state (Section 5.1) and five phases: LECS Connect phase (5.2), Configuration phase (5.3), Join phase (5.4), Initial Registration phase (5.5), and BUS Connect phase (5.6). These five phases MUST be performed in sequence, starting with the LECS Connect phase. Following the completion of the BUS Connect phase, the initialization procedure is complete, and the LE Client is *operational*. If the initialization phase or the operational state terminates abnormally the LE Client MUST return to the Initial state and inform layer management.

All five phases of the initialization procedure are required for an LE Client to expect to achieve full interoperability. *Note: Provision is made for the LECS Connect and Configuration phases to be empty, so that their functions can be bypassed for certain applications. The Initial Registration phase is also an empty phase if the LE Client can perform all needed address registration during the Join phase.*

## 5.1 Initial State

### 5.1.1 Initial State - LE Client View

The initial state of the LE Client is determined by means outside the scope of this document. The state of the variables defined in this section after the orderly termination of an LE Client, or its abnormal termination from any initialization or operational phase, is an implementation issue not addressed in this specification.

Certain of the parameters, below, contain “minimum”, “maximum”, and/or “default” values. A variable MUST NOT be set to a value smaller than its minimum value or larger than its maximum value. Most ATM Emulated LANs, if composed entirely of LAN Emulation components compliant with this specification, and whose components’ variable values are set to the default values, should operate correctly. The behavior of specific configurations may be optimized by altering the values away from the defaults, but such optimization is beyond the scope of this specification. Values outside the specified minima and maxima are likely to result in an emulated LAN that functions poorly or not at all for many applications.

The following parameters apply to each LE Client:

**C1** **LE Client's ATM Addresses.** The LE Client's own ATM Addresses. The primary ATM address used to connect to the LES and the BUS MUST be known before the Configuration and Join phases can start, and MUST NOT change without restarting the Configuration and Join phases. The Primary ATM address MUST be used to establish the LE Client's Control Direct and Multicast VCCs, and MUST be specified as the SOURCE-ATM-ADDRESS in the client's LE_JOIN_REQUESTs. An LE Client MAY have additional ATM addresses for use with Data Direct VCCs. These addresses do not need to be known at Join time, and can be removed from service without restarting the Join phase.

**C2** **LAN Type.** The type of LAN that the LE Client is, or wishes to become, a member of. This MUST be one of Ethernet/IEEE 802.3, IEEE 802.5, or Unspecified. MUST NOT be Unspecified after a successful Join. This parameter MUST NOT be changed without terminating the LE Client and returning to the Initial state.

**C3** **Maximum Data Frame Size.** The maximum AAL-5 SDU size of a data frame that the LE Client wishes to send on the Multicast Send VCC or to receive on the Multicast Send VCC or Multicast Forward VCC. This parameter also specifies the maximum AAL-5 SDU of all of an LE Client's Data Direct VCCs. This MUST be either 1516, 4544, 9234, or 18190 octets, or Unspecified. MUST NOT be Unspecified after a successful Join. MUST NOT be changed without terminating the LE Client and returning it to the Initial state.

**C4** **Proxy.** This indicates whether the LE Client may have remote unicast MAC addresses in C27. For example, an IEEE 802.1D transparent bridge MUST NOT register with the LE Server the MAC addresses of the endstations on its other LAN segments. MUST be known before the Join phase can start. This parameter MUST NOT change without restarting the Configuration phase.

**C5** **ELAN Name.** The identity of the emulated LAN the LE Client wishes to join, or to which the LE Client last joined. MAY be unspecified before Join. Never unspecified after a successful Join. Formatted as an SNMPv2 DisplayString of length 0-32 octets. MAY have a zero length after a successful Join. The ELAN Name client parameter and ELAN_NAME field in the Join protocol provide a way to configure clients with human-readable strings for network management purposes.

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

**C7** **Control Time-out.** Time out period used for timing out most request/response control frame interactions, as specified elsewhere.
Value: Minimum=10 seconds, Default=120 seconds, Maximum=300 seconds.

**C8** **Route Descriptor(s).** Route Descriptors exist only for source-routed IEEE 802.5 LE Clients that are Source-Route Bridges. All Route Descriptors in any given emulated LAN MUST be unique. An LE Client MAY have zero or more Route Descriptors and these Route Descriptors MAY change during normal operation. In an operational LE Client, every Local Route Descriptor in C8 MUST have been registered with the LE Server. When answering an LE_ARP_REQUEST for any address in this list, the "Remote Address" bit in the FLAGS field of the LE_ARP_RESPONSE MUST be clear.

**C9** **LE Server ATM Address.** The ATM address of the LAN Emulation Server is used to establish the Control Direct VCC. This is obtained in the Configuration phase. This address MUST be known before the Join phase can start.

**C10** **Maximum Unknown Frame Count.**

Value: Minimum=1, Default=1, Maximum=10. (See parameter C11.)

**C11** **Maximum Unknown Frame Time.** Within the period of time defined by the Maximum Unknown Frame Time, an LE Client will send no more than Maximum Unknown Frame Count frames to the BUS for a given unicast LAN Destination, and it must also initiate the address resolution protocol to resolve that LAN Destination.
Value: Minimum=1 second, Default=1 second, Maximum=60 seconds.

**C12** **VCC Time-out Period.** An LE Client SHOULD release any Data Direct VCC that it has not been used to transmit or receive any data frames for the length of the VCC Time-out Period. This parameter is only meaningful for SVC Data Direct VCCs.
Value: Minimum=None specified, Default=20 minutes, Maximum=Unlimited.

**C13** **Maximum Retry Count.** An LE Client MUST not retry an LE_ARP_REQUEST for a given frame's LAN destination more than Maximum Retry Count times, after the first LE_ARP_REQUEST for that same frame's LAN destination.
Value: Minimum=0, Default=1, Maximum=2.

**C14** **LE Client Identifier.** Each LE Client requires an LE Client Identifier (LECID) assigned by the LE Server during the Join phase. The LECID is placed in control requests by the LE Client and MAY be used for echo suppression on multicast data frames sent by that LE Client. This value MUST NOT change without terminating the LE Client and returning to the Initial state. A valid LECID MUST be in the range X"0001" through X"FEFF".

**C15** **LE Client Multicast MAC Address(es).** Each LE Client MAY have a list of multicast MAC addresses which it wishes to receive and pass up to the higher layers. The broadcast address SHOULD be included in this list.

**C16** **LE_ARP Cache**. A table of entries, each of which establishes a relationship between a LAN Destination external to the LE Client and the ATM address to which data frames for that LAN Destination will be sent.

**C17** **Aging Time**. The maximum time that an LE Client will maintain an entry in its LE_ARP cache in the absence of a verification of that relationship.
Value: Minimum=10 seconds, Default=300 seconds, Maximum=300 seconds.

**C18** **Forward Delay Time**. The maximum time that an LE Client will maintain an entry for a non-local MAC address in its LE_ARP cache in the absence of a verification of that relationship, as long as the Topology Change flag C19 is true.
Value[16]: Minimum=4 seconds, Default=15 seconds, Maximum=30 seconds.

**C19** **Topology Change**. Boolean indication that the LE Client is using the Forward Delay Time C18, instead of the Aging Time C17, to age non-local entries in its LE_ARP cache C16.

**C20** **Expected LE_ARP Response Time**. The maximum time that the LEC expects an LE_ARP_REQUEST / LE_ARP_RESPONSE cycle to take. Used for retries and verifies.
Value: Minimum=1 second, Default=1 second, Maximum=30 seconds.

**C21** **Flush Time-out**. Time limit to wait to receive an LE_FLUSH_RESPONSE after the LE_FLUSH_REQUEST has been sent before taking recovery action.
Value: Minimum=1 second, Default=4 seconds, Maximum=4 seconds.

**C22** **Path Switching Delay**. The time since sending a frame to the BUS after which the LE Client may assume that the frame has been either discarded or delivered to the recipient. May be used to bypass the Flush protocol.
Value: Recommend: Minimum=1 second, Default=6 seconds, Maximum=8 seconds.

**C23** **Local Segment ID**. The segment ID of the emulated LAN. This is only required for IEEE 802.5 LE Clients that are Source Routing bridges. This is the source routing Segment ID for the emulated LAN and is used as outlined in the Source Routing Section

[16]This is consistent with IEEE 802.1d, Table 4-3.

2.3.7 (Source Routing Considerations) and Appendix 12.4.2 (Token Ring Source Routing Information and Structures).

**C24** **Multicast Send VCC Type**. Signaling parameter that SHOULD be used by the LE Client when establishing the Multicast Send VCC. This is the method to be used by the LE Client when specifying traffic parameters when it sets up the Multicast Send VCC for this emulated LAN.

**C25** **Multicast Send VCC AvgRate**. Signaling parameter that SHOULD be used by the LE Client when establishing the Multicast Send VCC. Forward and Backward Sustained Cell Rate to be requested by LE Client when setting up Multicast Send VCC, if using Variable bit rate codings.

**C26** **Multicast Send VCC PeakRate**. Signaling parameter that SHOULD be used by the LE Client when establishing the Multicast Send VCC. Forward and Backward Peak Cell Rate to be requested by LE Client when setting up the Multicast Send VCC when using either Variable or Constant bit rate codings.

**C27** **Remote Unicast MAC Address(es)**. The MAC addresses for which this LE Client will answer LE_ARP_REQUESTs, but which are not registered with the LE Server. This list MUST be empty in any operational LE Client that did not join the emulated LAN as a proxy agent (C4). When answering an LE_ARP_REQUEST for any address in this list, the “Remote Address” bit in the FLAGS word of the LE_ARP_RESPONSE MUST be set.

**C28** **Connection Completion Timer**. Optional. In Connection Establishment this is the time period in which data or a READY_IND message is expected from a Calling Party. Value: Minimum=1 second, Default=4 seconds, Maximum=10 seconds.

### 5.1.2 Initial State - LE Service View

The following parameters apply per emulated LAN served by an LE Service:

**S1** **LE Server’s ATM Addresses.** The LE Server MUST know its own ATM Addresses for LE Clients to be able to establish a connection to it. An ATM address cannot be removed from this set as long as any LE Client is connected to the LE Server through it.

**S2** **LAN Type.** The type of this ATM Emulated LAN, either Ethernet/IEEE 802.3 or IEEE 802.5.

**S3** **Maximum Data Frame Size.** The maximum AAL-5 SDU size of a data frame that the LE Service can guarantee not to drop because it is too large. Also the minimum AAL-5 SDU size that every LE Client must be able to receive. MUST be either 1516, 4544, 9234, or 18190 octets.

**S4** **Control Time-out.** Time out period used for timing out most request/response control frame interactions, as specified elsewhere. Once an LE Client establishes a Control Direct VCC to the LE Server, the Join phase must complete within the Join Time-out time. If not, the LE Service SHOULD release any Control VCCs to that LE Client, terminating the Join phase.
Value: Minimum=10 seconds, Default=120 seconds, Maximum=300 seconds.

**S5** **Maximum Frame Age.** The BUS MUST discard a frame if it has not transmitted the frame to all relevant Multicast Send VCCs or Multicast Forward VCCs within the Maximum Frame Age following the BUS’s receipt of the frame over a Multicast Send VCC.
Value: Minimum=1 second, Default=1 second, Maximum=4 seconds.

**S6** **Broadcast and Unknown Server's ATM Address(es).** A Broadcast and Unknown Server MUST know at least one of its own ATM addresses for LE Clients to be able to establish connections to it. A Broadcast and Unknown Server MAY have several ATM addresses. Addresses MAY be added while the BUS is operational, but MAY NOT be removed as long as any LE Client is connected to the BUS through them.

## 5.2 LECS Connect Phase

During the LECS Connect phase, the LE Client establishes its connection with the LE Configuration Server.

If use of the LECS is not required, an LE Client MAY execute a “null” LECS Connect phase by using a preconfigured SVC or PVC to the LES. This VCC is established by some means outside of this specification. If the LE Client does not use the LECS Connect procedures, it MUST NOT use the procedures in Sections 5.2 and 5.3.

### 5.2.1 LECS Connect - LE Client View

The mechanisms used to locate the configuration service are as follows, in the order in which an LE Client MUST attempt them:

1. Get the LECS Address via ILMI
2. Use the well-known LECS Address
3. Use the LECS PVC

These mechanisms are described in the following paragraphs.

#### 5.2.1.1 Get LECS Address via ILMI

The LE Client MUST issue an ILMI Get or GetNext to obtain the ATM address of the LECS for that UNI. If the UNI connection to the obtained address fails, the LE Client MAY issue an ILMI Get or GetNext request to determine if an additional LE Configuration Server ATM address is available, and attempt to establish the Configuration Direct VCC to that ATM address. The ILMI MIB table to be read to obtain the LECS ATM Address is defined in the MIB in Section 12.2.

#### 5.2.1.2 Well-known LECS Address

If the LECS ATM address cannot be obtained from ILMI or if the LE Client is unable to establish a Configuration Direct VCC to that address, then the well-known address X“47007900000000000000000000-00A03E000001-00” MUST be used to open a configuration VCC to the configuration service. 00 A0 3E is the ATM Forum-assigned OUI and 000001 has been allocated by the ATM Forum.

#### 5.2.1.3 LECS PVC

If the LE Client cannot establish a VCC to the well-known ATM address of the LECS, then the well-known PVC of VPI=0, VCI=17 (decimal) MUST be used for the Configuration Direct VCC.

#### 5.2.1.4 Configuration Direct SVC

The LE Client MUST attempt to establish a Configuration Direct VCC using the call parameters for LE Configuration Direct VCCs defined in Table 5. The called party address MUST be the ATM addresses obtained as above. The calling party address may be any valid ATM address belonging to the end station.

### 5.2.2 LECS Connect - LE Service view

#### 5.2.2.1 Finding At Least One LECS

The provider of the LAN Emulation service MUST ensure that an LE Client following the rules of section 5.2.1 will be able to connect to an LE Configuration Server.

#### 5.2.2.2 Finding the Correct LECS

If an LE Client is able to establish a Configuration Direct VCC, then the LECS Connect phase has completed successfully, whether or not the LE Client is able to complete the Configuration phase. Therefore, the LAN Emulation service SHOULD ensure that the state does not persist where an LE Client can connect to a "dead" LE Configuration Server.

## 5.3 Configuration Phase

During the Configuration phase, the LE Client (or other entity) obtains the ATM Address of the LE Server, and may obtain additional configuration parameters. Configuration control frames are of two types:

- **LE_CONFIGURE_REQUEST**: Sent by the LE Client or other interested party to the LECS to obtain configuration information.
- **LE_CONFIGURE_RESPONSE**: Sent by the LE Configuration Server in response to an LE_CONFIGURE_REQUEST.

The configuration phase prepares an LE Client for the Join phase by providing the necessary operating parameters for the emulated LAN which the client will later join.

The Configuration phase may be performed either by using the LE configuration protocol or by static parameters in each LEC.

The LE Configuration protocol allows the assignment of individual LE Clients to different emulated LANs and provides information about the operating parameters of that LAN. Based on its own policies, configuration databases and information provided by clients, an LE Configuration Server assigns any client which requests configuration information to a particular LE Service entity by giving it the LE Server's ATM address and other parameters.

This protocol assumes that the requesting entity already knows how many LE Clients to instantiate. An end-station may request configuration information separately for each LE Client and/or MAC address it represents. The LE Configuration Server returns the address of an LE server which is suitable, given the supplied information.

It is expected that the LAN Emulation Configuration protocol will be used for more purposes than the one of a non-bridge LE Client discovering which ELAN it should join. A typical use for this protocol could include for example:

> A bridge or concentrator supporting multiple LE Clients, and representing multiple legacy LAN endstations, may determine which of its LE Clients (ELANs) should be bound to which of its legacy endstations, based on the endstations' MAC addresses.

If use of the LECS is not required, an LE Client MAY execute a "null" LECS Connect phase and bypass the Configuration Phase by using a preconfigured SVC or PVC to the LES. This VCC is established by some means outside of this specification.

### 5.3.1 Configuration - Requester's View

This section is described in terms of a "requester" which has established a Configuration Direct VCC, and is issuing LE_CONFIGURE_REQUESTs and receiving LE_CONFIGURE_RESPONSEs, and a "prospective LE Client" on whose behalf the requests are being made. In the case of an LE Client configuring itself, these two entities are one and the same. In other cases, they may be different.

#### 5.3.1.1 Configure Request

The requester MUST issue an LE_CONFIGURE_REQUEST to the LE Configuration Server containing at least the primary ATM Address of the prospective LE Client in the SOURCE-ATM-ADDRESS field. Other information MAY be included in the remaining fields of the LE_CONFIGURE_REQUEST as described in Table 16.

#### 5.3.1.2 Unsuccessful Configure Response

If the LE_CONFIGURE_RESPONSE does not contain 0 (Success) in the STATUS field, then the Configuration phase has failed for the prospective LE Client. If this answer is not satisfactory, e.g. in the case of an LE Client configuring itself, the LE Client MUST return to the beginning of the Initialization procedure if it wishes to re-attempt to configure itself.

#### 5.3.1.3 Successful Configure Response

If the LE_CONFIGURE_RESPONSE does contain 0 (Success) in the STATUS field, then the Configuration phase has succeeded for the prospective LE Client. In this case, the LAN-TYPE, MAXIMUM-FRAME-SIZE, TARGET-ATM-ADDRESS, and ELAN-NAME parameters MUST be copied to the prospective LE Client's C2, C3, C9, and C5 variables, respectively.

#### 5.3.1.4 Configure Response TLV Encodings

Table 17 illustrates the LE_CONFIGURE_REQUEST and LE_CONFIGURE_RESPONSE Type-Length-Value (TLV) encodings. A requester MUST recognize the values listed in this table and transfer any values returned in a successful LE_CONFIGURE_RESPONSE to the corresponding variables in the prospective LE Client.

#### 5.3.1.5 Retrying Configure Request

If the LE Configuration Server does not return an LE_CONFIGURE_RESPONSE within the LE Client's Control Time-out C7, the requester MAY repeat the LE_CONFIGURE_REQUEST, with all fields identical to the first request except for the TRANSACTION-ID field. If no LE_CONFIGURE_RESPONSE is obtained after some number of retries, then the Configuration phase has failed. If this failure is not satisfactory, e.g. in the case of an LE Client configuring itself, the LE Client MUST return to the beginning of the Initialization procedure if it wishes to re-attempt to configure itself.

#### 5.3.1.6 Configure Response Ordering

Because the LECS is not required to return LE_CONFIGURE_RESPONSEs in the same order that the corresponding LE_CONFIGURE_REQUESTs were sent, a requester that issues multiple requests SHOULD use the TRANSACTION-ID field to distinguish between responses to different requests.

#### 5.3.1.7 Releasing the Configuration Direct VCC

At any time, the requester MAY release the Configuration Direct VCC. Any number of whole or partial exchanges of LE_CONFIGURE_REQUEST and LE_CONFIGURE_RESPONSE frames, including 0, may occur before the Configuration Direct VCC is released.

### 5.3.2 Configuration - LE Service View

The LE Configuration Server uses the information provided in the LE_CONFIGURE_REQUEST to generate an LE_CONFIGURE_RESPONSE. This response may indicate success or failure, depending on whether the prospective LE Client is to be allowed to attempt to join an LE Server.

#### 5.3.2.1 Requester's ATM Address

The calling party's ATM address used in signaling the Configuration Direct VCC MUST NOT be considered by the LECS in determining which ELAN to direct the requester to, but MAY be used in deciding whether or not to respond and/or release the connection for security reasons.

#### 5.3.2.2 LAN-TYPE Configure Response

If the LAN-TYPE in the LE_CONFIGURE_REQUEST is not Unspecified, then the LAN-TYPE in the LE_CONFIGURE_RESPONSE MUST have the same value as that in the LE_CONFIGURE_REQUEST. If the LAN-TYPE in the LE_CONFIGURE_REQUEST is Unspecified, then the LAN-TYPE in the LE_CONFIGURE_RESPONSE MAY have any value listed in Table 16.

#### 5.3.2.3 MAXIMUM-FRAME-SIZE Configure Response

If the MAXIMUM-FRAME-SIZE in the LE_CONFIGURE_REQUEST is not Unspecified, then the MAXIMUM-FRAME-SIZE in the LE_CONFIGURE_RESPONSE MUST have the same value as or a lower value than that in the LE_CONFIGURE_REQUEST, but MUST NOT be Unspecified. If the MAXIMUM-FRAME-SIZE in the LE_CONFIGURE_REQUEST is Unspecified, then the MAXIMUM-FRAME-SIZE in the LE_CONFIGURE_RESPONSE MAY have any value listed in Table 16.

#### 5.3.2.4 Configure Response Ordering

The LECS MAY issue LE_CONFIGURE_RESPONSEs in a different order than that in which the corresponding LE_CONFIGURE_REQUESTs were received, even for LE_CONFIGURE_REQUESTs received on the same Configure Direct VCC.

#### 5.3.2.5 Configuration vs. Operation

The set up or release of a Configuration Direct VCC by an LE Client, or any exchange of LE_CONFIGURE_REQUEST or LE_CONFIGURE_RESPONSE frames, MUST NOT affect the operational status of any LE component.

### 5.3.3 Configuration Frames

The format of each AAL5 SDU for LE Configuration Request and LE Configuration Response packet is as follows:

**Table 16. Configuration Frame Format**

| Offset | Size | Name | Function |
|---|---|---|---|
| 0 | 2 | MARKER | Control Frame = X"FF00" |
| 2 | 1 | PROTOCOL | ATM LAN Emulation protocol = X"01" |
| 3 | 1 | VERSION | ATM LAN Emulation protocol version = X"01" |
| 4 | 2 | OP-CODE | Type of request:<br>X"0001" LE_CONFIGURE_REQUEST<br>X"0101" LE_CONFIGURE_RESPONSE |
| 6 | 2 | STATUS | Always X"0000" in requests. See Table 13 for a list of values. |
| 8 | 4 | TRANSACTION-ID | Arbitrary value supplied by the requester and returned by the responder. |
| 12 | 2 | REQUESTER-LEC-ID | Always X"0000" in requests, ignored on response. |
| 14 | 2 | FLAGS | Always X"0000" when sent, ignored on receipt. |
| 16 | 8 | SOURCE-LAN-DESTINATION | MAC address or Route Descriptor of prospective LE Client. MAY be encoded as "not present." |
| 24 | 8 | TARGET-LAN-DESTINATION | Always X"0000" when sent, ignored on receipt. |
| 32 | 20 | SOURCE-ATM-ADDRESS | Primary ATM address of prospective LE Client for which information is requested. |
| 52 | 1 | LAN-TYPE | X"00" Unspecified[17]<br>X"01" Ethernet/IEEE 802.3<br>X"02" IEEE 802.5 |

[17] If the LAN-TYPE is "unspecified," then the Ethernet/IEEE 802.3 MAC address format MUST be used.

| | | | |
|---|---|---|---|
| 53 | 1 | MAXIMUM-FRAME-SIZE | X"00" Unspecified<br>X"01" 1516<br>X"02" 4544<br>X"03" 9234<br>X"04" 18190 |
| 54 | 1 | NUMBER-TLVS | Number of Type/Length/Value elements encoded in Request/Response. |
| 55 | 1 | ELAN-NAME-SIZE | Number of octets in ELAN-NAME (may be 0). |
| 56 | 20 | TARGET-ATM-ADDRESS | ATM Address of the LE Server to be used for the LE Client described in the request if Configure Response and STATUS="Success," else X'00'. |
| 76 | 32 | ELAN-NAME | Name of emulated LAN[18]. |
| 108 | 4 | ITEM_1-TYPE | Three octets of OUI, one octet identifier. |
| 112 | 1 | ITEM_1-LENGTH | Length in octets of VALUE field.  Minimum=0. |
| 113 | Variable | ITEM_1-VALUE | |
| | | Etc. | |

Further parameters may be included by either the requester or the configuration server, encoded using Type/Length/Value (TLV) where present, in order to accommodate servers with user-defined and optional extensions. As many parameters may be included as will fit into the maximum SDU size negotiated on the Configuration Direct VCC.

In Type Field, OUI designates the authority responsible for allocating the four octet value of a Type value.  The OUI value 00-A0-3E is used for standard values defined by the ATM Forum specification.  No padding is in the TLV field.  Zero, one, or more TLV fields may be present.

Further vendor-specific extensions may be implemented by servers using the OUI of the vendor defining the extensions.  The first three octets of a TYPE value MUST contain an OUI.  The last octet of a TYPE value should be used to distinguish between different parameters defined by an organization.

### 5.3.4 Configuration Request Frame - Parameter Encodings

There are no ATM Forum specified T/L/V encodings for use in LE_CONFIGURE_REQUEST frames.

### 5.3.5 Configuration Response Frame - Parameter Encodings

The following table describes the T-L-V encodings of further optional operational parameters supplied by a LECS in answer to a configuration request by a LE Client. An LE Client MUST update its operational parameter set with these values if it can parse them. An LE Client MUST ignore any encoded 'Type' values which it does not understand. The Value provided for any

[18]SNMPv2 DisplayString

parameter is in the same units as defined in Section 5.1 and the References are to that same section where appropriate.

**Table 17.Configuration Response Frame - Parameter Encodings**

| Item | Type | Length | Reference/Value/Units |
|---|---|---|---|
| Control Time-out | 00-A0-3E-01 | 2 | C7/in seconds |
| Maximum Unknown Frame Count | 00-A0-3E-02 | 2 | C10 |
| Maximum Unknown Frame Time | 00-A0-3E-03 | 2 | C11/in seconds |
| VCC Time-out Period | 00-A0-3E-04 | 4 | C12/in seconds |
| Maximum Retry Count | 00-A0-3E-05 | 2 | C13 |
| Aging Time | 00-A0-3E-06 | 4 | C17/in seconds |
| Forward Delay Time | 00-A0-3E-07 | 2 | C18/in seconds |
| Expected LE_ARP Response Time | 00-A0-3E-08 | 2 | C20/in seconds |
| Flush Time-out | 00-A0-3E-09 | 2 | C21/in seconds |
| Path Switching Delay | 00-A0-3E-0A | 2 | C22/in seconds |
| Local Segment ID | 00-A0-3E-0B | 2 | C23 |
| Mcast Send VCC Type | 00-A0-3E-0C | 2 | C24:<br>X'0000' Best Effort: LE Client should set the BE flag. Peak Cell Rates should be line rate.<br>X'0001' Variable: LE Client should provide a Sustained Cell Rate (see below).<br>X'0002' Constant: LE Client should provide a Peak and a Sustained Cell Rate (see below). |
| Mcast Send VCC AvgRate | 00-A0-3E-0D | 4 | C25/in cells per second |
| Mcast Send VCC PeakRate | 00-A0-3E-0E | 4 | C26/in cells per second |
| Connection Completion Timer | 00-A0-3E-0F | 2 | C28/in seconds |

where 00-A0-3E is the ATM Forum OUI.

## 5.4 Join Phase

During the Join phase, the LE Client establishes its connection(s) with the LE Server and determines the operating parameters of the emulated LAN. The LE Client also may implicitly register one MAC address with the LE Server as a result of joining the emulated LAN. Join protocol frames are of two types:

- **LE_JOIN_REQUEST:** Sent by the LE Client to the LE Server. Requests that the LE Client be allowed to join an ATM Emulated LAN.
- **LE_JOIN_RESPONSE:** Sent by the LE Server to the LE Client in response to an LE_JOIN_REQUEST frame. Confirms or denies the join request.

These frame formats are described in Table 18.

### 5.4.1 Join Phase - LE Client View

#### 5.4.1.1 Control Direct VCC

The LE Client MUST either initiate and complete UNI signaling procedures to establish a point-to-point bi-directional Control Direct VCC between its LE Client's ATM address C1 and the LE Server ATM address C9, or identify the appropriate Control Direct PVC. If the LE Client cannot establish this connection, then it MUST terminate the Join procedure. SETUP request for this VCC MUST be signaled using the call parameters defined in Section 3.3 for LE Control Direct VCCs. The calling party ATM Address used by the LE Client when setting up the Control Direct VCC MUST be the client's primary ATM Address.

#### 5.4.1.2 Transmitting LE_JOIN_REQUEST

The LE Client MUST send an LE_JOIN_REQUEST over the Control Direct VCC. The LE_JOIN_REQUEST MUST include the LE Client variables C2, C3, C4, and C5, plus the primary ATM address from LE Client variable C1. It MAY also include one MAC address from C6 to be registered as a pair with the SOURCE_ATM_ADDRESS.

#### 5.4.1.3 Accepting Control Distribute VCC

Once the LE Client sends the LE_JOIN_REQUEST, it MUST accept an attempt by the LE Server to establish a Control Distribute VCC, if that attempt is made before the LE_JOIN_RESPONSE is returned by the LE Server. This VCC SETUP Indication will be signaled with the call parameters defined in Section 3.3 for LE Control Distribute VCCs. An LE Client SHOULD refuse any attempt to establish a second or subsequent Control Distribute VCC.

#### 5.4.1.4 Receiving LE_JOIN_RESPONSE

The LE Server returns an LE_JOIN_RESPONSE to the LE Client. The LE_JOIN_RESPONSE may be returned on either the Control Direct VCC or the Control Distribute VCC (if created). The LE Client MUST be prepared to receive it on either VCC. The LE Client MUST be prepared to receive and ignore LE_JOIN_RESPONSE frames or any other control frames on either Control VCC that are not directed to itself.

#### 5.4.1.5 Receiving on Control Distribute VCC

If the LE Server establishes a Control Distribute VCC, then the LE Client MUST be prepared to receive control frames over the Control Distribute VCC. The SETUP request for this VCC MUST be signaled using the call parameters defined in Section 3.3 for LE Control Direct VCCs.

#### 5.4.1.6 Future Control Distribute VCC Establishment

Once the LE Client receives an LE_JOIN_RESPONSE from the LE Server, if the LE Server has not established a Control Distribute VCC, then the LE Client MAY assume that no Control

Distribute VCC will be established by that LE Server, no matter what the status of the LE_JOIN_RESPONSE, and MAY refuse the connection if the LE Server attempts to make it.

#### 5.4.1.7 Validating LE_JOIN_RESPONSE

If the LE_JOIN_RESPONSE indicates a successful join, it MUST contain values for the LE Client's variables C2, C3, C5, and C14. (The value for C5 may be zero length.) The LE Client MUST either update these variables from the values given by the LE Server, or terminate the ELAN membership. If the values are accepted by the LE Client, the LE Client Join phase of initialization is complete and successful.

#### 5.4.1.8 Termination of Join Procedure

If the LE_JOIN_RESPONSE indicates a failure, then the LE Client MUST terminate the ELAN membership.

#### 5.4.1.9 LE_JOIN_RESPONSE Time-out (C7)

If the LE Client receives no LE_JOIN_RESPONSE within the time C7, then it MAY retry the Join procedure, but MUST terminate the ELAN membership after a finite number of retries.

#### 5.4.1.10 Unexpected Control Direct and Control Distribute VCC Clearing

If the LE Client detects that the Control Direct VCC or the Control Distribute VCC are released at any time other that those specified here, then the LE Client MUST terminate the ELAN membership.

#### 5.4.1.11 Ignoring LE_JOIN_REQUEST

An LE Client MUST NOT respond to an LE_JOIN_REQUEST.

### 5.4.2 Join phase - LE Server View

#### 5.4.2.1 Control Direct VCC

An LE Server MAY accept a UNI signaling request to establish a Control Direct VCC to its own ATM address S1. An LE Server MUST NOT accept a Control Direct VCC establishment unless the signaling SETUP Indication contains the call parameters defined in Section 3.3 for LE Control Direct VCCs.

#### 5.4.2.2 Waiting for LE_JOIN_REQUEST (S4)

If no LE_JOIN_REQUEST is received on a new Control Direct VCC within some period of time S4, the LE Server MAY terminate the ELAN membership.

#### 5.4.2.3 Control Distribute VCC

When the LE Server receives an LE_JOIN_REQUEST, it MUST decide whether the Join request is to succeed or fail. If and only if the request is to succeed, the LE Server MAY attempt to establish a Control Distribute VCC to the LE Client using the call parameters defined in Section 3.3 for LE Control Distribute VCCs. This VCC MUST be established before the LE Server can proceed with the Join phase. The LE Server MUST NOT attempt to establish a Control Distribute VCC to an LE Client after sending that LE Client an LE_JOIN_RESPONSE.

#### 5.4.2.4 Validating LE_JOIN_REQUEST

For an LE_JOIN_REQUEST to succeed, the values for LAN type and maximum frame size (C2 and C3) MUST be compatible with that of the LE Server. That is, the LAN type must either be Unspecified or must match the LE Server's LAN type S2 exactly, and the maximum frame size must either be Unspecified or must be greater than or equal to the LE Server's maximum frame size S3. If either condition is not met, the LE Server MUST reject the LE_JOIN_RESPONSE with a STATUS code of "Invalid request parameters."

An LE_JOIN_REQUEST containing a broadcast address, a multicast MAC address or a functional address as the source LAN destination MUST be rejected.

#### 5.4.2.5 Duplicate MAC Address

An LE Server MUST NOT allow two LE Clients to join one ATM Emulated LAN with the same MAC Address. An LE Server MUST accept a duplicate LE_JOIN_REQUEST from an LE Client for a MAC Address or Route Descriptor that duplicates an address or designator already registered by that LE Client.

#### 5.4.2.6 Duplicate ATM Address

An LE Server MUST reject an LE_JOIN_REQUEST if the SOURCE-ATM-ADDRESS field or Control Direct VCC Calling Party Number (which must be the same) duplicates another LE Client's ATM address (STATUS code "Duplicate ATM address"). An LE Server MUST reject a LE_JOIN_REQUEST if the REQUESTER-LECID field is not 0 (STATUS code "Invalid REQUESTER-LECID").

#### 5.4.2.7 Control Distribute VCC Failure

If the LE Client fails to accept the Control Distribute VCC connection then the LE Server MUST terminate the ELAN membership.

#### 5.4.2.8 Sending LE_JOIN_RESPONSE

The LE Server MAY return the LE_JOIN_RESPONSE to the LE Client on either the Control Direct VCC or the Control Distribute VCC (if created).

#### 5.4.2.9 REQUESTER-LECID

If the LE Server returns an LE_JOIN_RESPONSE indicating a successful join, the response MUST include a REQUESTER-LECID for the LE Client that is unique among all LE Clients joined to that same emulated LAN. The REQUESTER-LECID MUST be in the range allowed for LE Clients' LECIDs.

#### 5.4.2.10 Completion of Join Phase

If the LE_JOIN_RESPONSE indicates a successful join, then the LE Client and LE Server have completed the Join phase of initialization.

#### 5.4.2.11 Control Distribute VCC and LE_JOIN_RESPONSE

The LE Server MUST NOT send an LE_JOIN_RESPONSE with any status other than success if it has established a Control Distribute VCC.

#### 5.4.2.12 LE Client Release

If the LE Client does not release the Control Direct VCC after the LE Server sends the LE_JOIN_RESPONSE indicating a failure of the Join procedure, then the LE Server SHOULD terminate the ELAN membership. The release of the Control VCCs by the LE Client SHOULD occur within time S4 from the establishment of the Control Direct VCC to avoid this time-out.

#### 5.4.2.13 LE Server Termination

If the LE Server detects that the Control Direct VCC or the Control Distribute VCC to any given Client has failed, then the LE Server MUST terminate the ELAN membership.

#### 5.4.2.14 Duplicate LE_JOIN_REQUESTS

If the LE Server receives an LE_JOIN_REQUEST on a Control Direct VCC after it has returned any LE_JOIN_RESPONSE on that same Control Direct VCC, and if that second LE_JOIN_REQUEST differs from the first in any parameter except the TRANSACTION-ID, then the LE Server MUST terminate the ELAN membership. On receipt of a duplicate LE_JOIN_REQUEST (i.e. one that is identical to a previous LE_JOIN_REQUEST in all fields except TRANSACTION-ID, which may be different), the LE Server returns the same information it provided in response to the original LE_JOIN_REQUEST. This flexibility is designed to cover the case where an LE Client did not receive an LE_JOIN_RESPONSE.

#### 5.4.2.15 Duplicate Registered LAN-DESTINATION

An LE Server MUST reject an LE_JOIN_REQUEST if the SOURCE-LAN-DESTINATION field duplicates another LE Client's registered LAN-DESTINATION, whether that previous registration was made via the registration protocol or via implicit registration in an LE_JOIN_REQUEST (STATUS code "Duplicate LAN Destination registration").

### 5.4.3 Join Frames

The Join frames are used in the Join phase of LAN emulation initialization and are described in Table 18.

**Table 18. Join Frame Format**

| Offset | Size | Name | Function |
|---|---|---|---|
| 0 | 2 | MARKER | Control Frame = X"FF00" |
| 2 | 1 | PROTOCOL | ATM LAN Emulation protocol = X"01" |
| 3 | 1 | VERSION | ATM LAN Emulation protocol version = X"01" |
| 4 | 2 | OP-CODE | Type of request:<br>X"0002" LE_JOIN_REQUEST<br>X"0102" LE_JOIN_RESPONSE |
| 6 | 2 | STATUS | Always X"0000" in requests. In Responses: See Table 13 for a list of values. |
| 8 | 4 | TRANSACTION-ID | Arbitrary value supplied by the requester and returned by the responder. |
| 12 | 2 | REQUESTER-LECID | Assigned LECID of joining client if join response and STATUS = "Success", else X"0000". |
| 14 | 2 | FLAGS | Each bit of the FLAGS field has a separate meaning if set:<br>X"0080" Proxy Flag: LE Client serves non-registered MAC addresses and therefore wishes to receive LE_ARP requests for non-registered LAN destinations. |
| 16 | 8 | SOURCE-LAN-DESTINATION | Optional MAC address to register as a pair with the SOURCE_ATM_ADDRESS. |
| 24 | 8 | TARGET-LAN-DESTINATION | Always X"00" when sent, ignored on receipt. |
| 32 | 20 | SOURCE-ATM-ADDRESS | Primary ATM address of LE Client issuing join request. |
| 52 | 1 | LAN-TYPE | X"00" Unspecified<br>X"01" Ethernet/IEEE 802.3<br>X"02" IEEE 802.5 |
| 53 | 1 | MAXIMUM-FRAME-SIZE | X"00" Unspecified<br>X"01" 1516<br>X"02" 4544<br>X"03" 9234<br>X"04" 18190 |

| | | | |
|---|---|---|---|
| 54 | 1 | NUMBER-TLVS | Always X"00" when sent, ignored on receipt. |
| 55 | 1 | ELAN-NAME-SIZE | Number of octets in ELAN-NAME.<br>X"00" indicates empty ELAN-NAME. |
| 56 | 20 | TARGET-ATM-ADDRESS | Always X"00" when sent, ignored on receipt. |
| 76 | 32 | ELAN-NAME | Name of emulated LAN. Expresses LE Client's preference in LE_JOIN_REQUEST, specifies name of LAN joined in successful LE_JOIN_RESPONSE, else not used. Format is SNMPv2 DisplayString. |

## 5.5 Registration During Initialization

After completing the Join phase, an LE Client MAY attempt to register additional LAN Destinations from its C6 and C8 variables according to the procedures described in Section 6, Registration. Registration at this time is provided so that an LE Client can register those LAN Destinations which are required for its normal operation. An LE Client MAY (but is not required to) terminate all connections and restart Initialization if it fails to register any LAN Destinations at this time, without ever becoming operational.

## 5.6 Connecting to the BUS Protocol and Procedures

The Protocol and Procedures for an LE Client connecting to the BUS follows.

### 5.6.1 Protocol

#### 5.6.1.1 Connecting to the BUS - LE Client View

##### 5.6.1.1.1 Determining BUS ATM Address

A client MUST connect to the BUS. The address of the BUS is determined by using the address resolution procedure (LE_ARP_REQUEST) to resolve the all 1's broadcast MAC address.

##### 5.6.1.1.2 Multicast Send VCC

The LE Service will respond to the LE_ARP_REQUEST with an LE_ARP_RESPONSE containing an ATM address of the BUS (S6), which the LE Client MUST use as the called party in establishing a bi-directional Multicast Send VCC. The calling party ATM address used by the LE Client MUST be the LE Client's primary ATM address, as defined in Section 5.1.1. This VCC is signaled using the call parameters defined in Section 3.3 for the LE Ethernet/IEEE 802.3 Multicast Send VCC or the LE IEEE 802.5 Multicast Send VCC. All multicast destination packets MUST be transmitted to the BUS over this VCC.

##### 5.6.1.1.3 LE Client Accepts Multicast Forward VCC

In order to receive traffic sent to a multicast address on an emulated LAN, a client must first have a connection from the BUS. The act of opening the Multicast Send VCC described above will automatically cause the BUS to connect back to the client to establish a Multicast Forward VCC. This connection will come either in the form of a point-to-point or point-to-multipoint connection indication, at the discretion of the BUS.

The client MUST accept any such connection indications if it wishes to receive all of the broadcast and multicast frames (and potentially, unknown unicast frames) on the emulated LAN. The calling party identifier of the connection setup may or may not contain the ATM address which was returned to the client when it got the LE_ARP_RESPONSE described above. The called party identifier will contain the ATM address which the client used as the calling party identifier when it opened the Multicast Send VCC. The call parameters signaled by the BUS are those defined in Section 3.3 for LE Multicast Forward VCCs. An LE Client SHOULD refuse any attempt to establish a second or subsequent Multicast Forward VCC.

##### 5.6.1.1.4 Losing BUS Connections

If the LE Client detects release of the Multicast Send VCC for any reason, it MUST also release its Multicast Forward VCC (UNI Reason Code "Normal, Unspecified" - Cause 31). The LE Client MAY re-attempt connection to the BUS by re-executing the procedures of Section 5.6.1.1 for some finite number of retries. If unable to regain its connection to the BUS, the LEC MUST terminate its ELAN membership and return to the Initialization phase.

If the LE Client detects deliberate release of the Multicast Forward VCC (UNI Reason Code "Normal, Unspecified" - Cause 31) it MUST NOT attempt any recovery process for the VCC. The LE Client MUST terminate its ELAN membership and return to the Initialization phase.

If the LE Client detects accidental release of the Multicast Forward VCC (UNI Reason Code other than "Normal, Unspecified" - Cause 31) it MAY re-attempt connection to the BUS by re-executing the procedures of Section 5.6.1.1, after having first ensured that its Multicast Send VCC, if any, is deliberately released (UNI Reason Code "Normal, Unspecified" - Cause 31).

#### 5.6.1.2 Connecting to the BUS - BUS View

##### 5.6.1.2.1 BUS Opens Multicast Forward VCC

When an LE Client successfully joins an emulated LAN and after the client successfully establishes a bi-directional Multicast Send VCC to the BUS (see above), the BUS MUST attempt to open a Multicast Forward VCC back to the client.

This MAY be performed by opening a new point-to-point VCC to the client or by adding the client as a leaf to a point-to-multipoint call at the choice of the BUS implementation.

#### 5.6.1.2.2 Called Party Number Provided

The calling party number which was used by the LE Client to signal the Multicast Send VCC MUST be used by the BUS as the called party number (ATM address) for the Multicast Forward VCC.

#### 5.6.1.2.3 B-LLI Codepoint

The call parameters signaled by the BUS are those defined in Section 3.3 for LE Multicast Forward VCCs.

#### 5.6.1.2.4 Failure to Establish Multicast Forward VCC

If the BUS is unable to set up the Multicast Forward VCC, the LE Service MAY terminate that Client.

#### 5.6.1.2.5 BUS Action on Release of Multicast Send VCC

Failure of Multicast VCCs: If the BUS detects release of a LE Client's Multicast Send VCC for any reason, it MUST also release that LE Client's Multicast Forward VCC (UNI Reason Code "Normal, Unspecified" - Cause 31). The BUS MUST NOT attempt to re-establish the Multicast Send VCC.

#### 5.6.1.2.6 BUS Action on Release of Multicast Forward VCC

If the BUS detects release of a LE Client's Multicast Forward VCC for any reason, it MUST also release that LE Client's Multicast Send VCC (UNI Reason Code "Normal, Unspecified" - Cause 31). The BUS MUST NOT attempt to re-establish the Multicast Forward VCC.

# 6. Registration Protocol, Procedures and Frame Formats

Registration is the procedure whereby the LE Client establishes with the LE Server any additional (LAN destination, ATM Address) pairs not registered in the Join procedure. Registration and Unregistration of LAN destinations may occur at any time after successfully joining an emulated LAN.

Registration protocol frames are of four types:

•**LE_REGISTER_REQUEST:** Sent by the LE Client to the LE Server. Requests that the LE Server register one LAN destination - ATM address pair for the LE Client.

•**LE_REGISTER_RESPONSE:** Sent by the LE Server to the LE Client in response to an LE_REGISTER_REQUEST frame. Confirms or denies the registration request.

•**LE_UNREGISTER_REQUEST:** Sent by the LE Client to the LE Server. Requests that the LE Server remove the registration of one LAN destination - ATM address pair for the LE Client.

•**LE_UNREGISTER_RESPONSE:** Sent by the LE Server to the LE Client in response to an LE_UNREGISTER_REQUEST frame. Confirms or denies the unregistration request.

These frame formats are described in Table 19.

In the following sections describing registration, the terms “accept” and “deny” are used with regard to LE_REGISTER_REQUEST. The LE Server “accepts” an LE_REGISTER_REQUEST by returning a successful LE_REGISTER_RESPONSE, and “denies” it by returning a failure code in the LE_REGISTER_RESPONSE.

An LE Client with only one unicast MAC address need not use the registration protocol, since every LE Client may implicitly register one MAC address during the Join phase. Also, it should be noted that a Join with a MAC address is functionally equivalent to a Join without a MAC address, followed by a Register with a MAC address.

### 6.1.1 Registration - LE Client View

#### 6.1.1.1 Registration Protocol VCC

Registration protocol frames MUST only be sent by an LE Client over the Control Direct VCC.

#### 6.1.1.2 Registration Requirements

LECs on a Token Ring emulated LAN MUST register local LAN destinations (Local Unicast MAC Addresses, C6 and/or Route Descriptors, C8). Other LECs MUST register local LAN destinations (Local Unicast MAC Addresses, C6). Once an LE Client has reached the operational state, it MUST NOT have in its variables C6 or C8 any LAN Destination which has not successfully been registered with the LE Server. LE Clients MUST NOT register non-local LAN destinations (C27).

#### 6.1.1.3 Ignoring Registration Protocol Frames

An LE Client MUST ignore any registration protocol frames received over any VCC other than the Control Direct VCC or the Control Distribute VCC.

#### 6.1.1.4 Registering Broadcast, Multicast and Functional Addresses

An LE Client MUST NOT register for a broadcast address, a multicast group MAC address or functional address.

#### 6.1.1.5 Duplicate LE_REGISTER_REQUEST or LE_UNREGISTER_REQUESTs

An LE Client MAY repeat an LE_REGISTER_REQUEST or LE_UNREGISTER_REQUEST if no LE_REGISTER_RESPONSE or LE_UNREGISTER_RESPONSE has been received from the LE Server for a period of C7.

#### 6.1.1.6 Duplicate Registration Frame Frequency

An LE Client MUST NOT repeat an LE_REGISTER_REQUEST or LE_UNREGISTER_REQUEST for the same MAC address or Route Descriptor more often than once per second.

#### 6.1.1.7 Optional Termination

If an LE Client is unable to obtain a registration response from the LE Server after some number of attempts to register or unregister a LAN destination, it MAY terminate the ELAN membership.

#### 6.1.1.8 Ignoring Request Frames

An LE Client MUST NOT respond to an LE_REGISTER_REQUEST or an LE_UNREGISTER_REQUEST.

### 6.1.2 Registration - LE Server View

#### 6.1.2.1 Registration Frames

An LE Server MUST send all registration protocol frames over the Control Direct VCC or the Control Distribute VCC (if it exists) to an LE Client.

#### 6.1.2.2 Ignoring Registration Frames

An LE Server MUST ignore all registration protocol frames received over any VCC other than a Control Direct VCC.

#### 6.1.2.3 Multiple MAC Registrations

An LE Server MUST NOT allow two or more different LE Clients to register the same MAC Address or Route Descriptor.

#### 6.1.2.4 Multiple ATM Registrations

An LE Server MUST NOT allow two or more different LE Clients to register LAN destinations using the same ATM address in the SOURCE-ATM-ADDRESS field.

#### 6.1.2.5 One Registered ATM address per LAN Destination

An LE Server MUST NOT allow an LE Client to register more than one ATM address for any given LAN destination.

#### 6.1.2.6 Denying LE_REGISTER_REQUEST

An LE Server MUST deny an LE_REGISTER_REQUEST with the STATUS code "Invalid LAN Destination" for a broadcast address, a multicast MAC Address or functional address.

#### 6.1.2.7 Duplicate LE_REGISTER_REQUESTs for Same LE Client

An LE Server MUST positively acknowledge an otherwise valid LE_REGISTER_REQUEST for a (LAN destination, ATM address) pair that duplicates a (LAN destination, ATM address) pair already registered by that same LE Client.

#### 6.1.2.8 LE_UNREGISTER_REQUEST for Unregistered LAN Destination

An LE Server MUST return a successful response to an otherwise valid LE_UNREGISTER_REQUEST for a LAN destination the requester had not registered. It MUST NOT, however, actually unregister a LAN destination registered by another client.

#### 6.1.2.9 First Registration Wins

If an LE_REGISTER_REQUEST (or implicit registration in an LE_JOIN_REQUEST) conflicts with a previous registration, thus violating the above restrictions on multiple registration of a LAN Destination or ATM Address, then the LE_REGISTER_RESPONSE MUST indicate a failure (STATUS code "Duplicate LAN Destination registration" or "Duplicate ATM Address").

### 6.1.3 Registration Frames

Table 19 describes Registration frames, which are used to register MAC addresses and Route Descriptors with the LE Server.

**Table 19. Registration Frame Format**

| Offset | Size | Name | Function |
|---|---|---|---|
| 0 | 2 | MARKER | Control Frame = X"FF00" |
| 2 | 1 | PROTOCOL | ATM LAN Emulation protocol = X"01" |
| 3 | 1 | VERSION | ATM LAN Emulation protocol version = X"01" |
| 4 | 2 | OP-CODE | Type of request:<br>X"0004" LE_REGISTER_REQUEST<br>X"0104" LE_REGISTER_RESPONSE<br>X"0005" LE_UNREGISTER_REQUEST<br>X"0105" LE_UNREGISTER_RESPONSE |
| 6 | 2 | STATUS | Always X"0000" in requests. In Responses: See Table 13 for a list of values. |
| 8 | 4 | TRANSACTION-ID | Arbitrary value supplied by the requester and returned by the responder. |
| 12 | 2 | REQUESTER-LECID | LECID of LE Client issuing the register or unregister request and returned by the responder. |
| 14 | 2 | FLAGS | Always X"00" when sent, ignored on receipt. |
| 16 | 8 | SOURCE-LAN-DESTINATION | Unicast MAC address or Route Descriptor LE Client is attempting to register. |
| 24 | 8 | TARGET-LAN-DESTINATION | Always X"00" when sent, ignored on receipt. |
| 32 | 20 | SOURCE-ATM-ADDRESS | An ATM address of LE Client issuing the register or unregister request. |
| 52 | 56 | RESERVED | Always X"00" when sent, ignored on receipt. |

# 7. Address Resolution Protocol, Procedures and Frame Formats

The basic flow of the address resolution protocol and Data Direct VCC management has been described in Section 2. There are four types of frames associated with address resolution protocol:

- **LE_ARP_REQUEST:** Sent by an LE Client to determine the ATM address associated with a given MAC address or Route Descriptor.
- **LE_ARP_RESPONSE:** Sent by the LE Server or an LE Client in response to an LE_ARP_REQUEST to provide the information requested.
- **LE_NARP_REQUEST:** Sent by an LE Client to advertise changes in Remote address bindings.
- **LE_TOPOLOGY_REQUEST:** Sent by LE Client or LE Server to indicate whether network topology change is in progress.

These frame formats are described in Table 20, 21 and 22. The specific procedures for address resolution and VCC connection management are presented in separate sections, one for LE Clients, and one for the LE Server.

## 7.1 Address Resolution - LE Client View

### 7.1.1 Address Resolution Over Control Direct VCC

An LE Client MUST send all LE_ARP_REQUESTs and LE_ARP_RESPONSEs over the Control Direct VCC.

### 7.1.2 Accepting Address Resolution Frames

An LE Client MUST accept LE_ARP_REQUESTs and LE_ARP_RESPONSEs arriving on either the Control Direct VCC or the Control Distribute VCC.

### 7.1.3 Ignoring Frames Prior to Joining

An LE Client MUST NOT respond to any LE_ARP_REQUEST if it has not completed the Join procedure regardless of the other provisions in Section 7.1.

### 7.1.4 Responding to LE_ARP_REQUESTs for LAN Destinations

An operational LE Client MUST respond to any LE_ARP_REQUEST it receives that requests information for any LAN Destination in that client's variables C6 (Local Unicast MAC Addresses), C8 (Route Descriptors), and C27 (Remote Unicast MAC Addresses). It MUST NOT respond to any LE_ARP_REQUEST for a LAN Destination not in those three variables.

### 7.1.5 Local LAN Destinations

An LE Client MUST clear the Remote Address FLAGS bit in any LE_ARP_RESPONSE sent for a LAN Destination in that Client's variables C6 (Local Unicast MAC Addresses) or C8 (Route Descriptors).

### 7.1.6 Remote MAC Addresses

An LE Client MUST set the Remote Address FLAGS bit in any LE_ARP_RESPONSE sent for a LAN Destination in that Client's variable C27 (Remote Unicast MAC Addresses).

If a LAN-attached endsystem is moved from one proxy agent to another, it is possible for a period of time that both proxy LE Clients will reply to an LE_ARP_REQUEST for the same LAN-attached endsystem.

### 7.1.7 Obtaining BUS ATM Address

An LE Client MUST obtain the address of the Broadcast and Unknown Server by sending an LE_ARP_REQUEST for the broadcast group address.

### 7.1.8 Transmitting to Unresolved LAN Destination

For each C10 data frames for the same unresolved unicast LAN destination sent by an LE Client to the BUS within a time period C11 seconds, that LE Client MUST send at least one LE_ARP_REQUEST to the LE Server to resolve that LAN destination.

### 7.1.9 Establishing Connections

Except in a PVC-only environment, upon receiving an LE_ARP_RESPONSE containing an ATM address for which it has an unresolved LE-ARP cache entry pending and no existing Data Direct VCC, an LE Client which supports SVCs MUST either issue a connection SETUP for a direct VCC to that address if its available resources permit, or else a connection RELEASE to some other VCC to free up the resources to make that connection.

### 7.1.10 LE_ARP_REQUEST Frequency

An LE Client MUST NOT transmit an LE_ARP_REQUEST for the same LAN destination more often than once every second.

### 7.1.11 Multicast MAC Address

An LE Client MUST NOT send an LE_ARP_REQUEST to resolve a multicast (or broadcast, except all 1's broadcast) MAC address.

### 7.1.12 LE_ARP Cache (C16)

An LE Client MUST cache in C16 the LAN Destination / ATM Address mapping information received in at least those LE_ARP_RESPONSEs corresponding to LE_ARP_REQUESTs sent by that LE Client.

### 7.1.13 Verifying LE_ARP Cache Entries

An LE Client MUST attempt to verify an entry in its LE_ARP cache before deleting it due to a time out. An LE Client MAY verify an entry in its LE_ARP cache (C16) by one or more of the following methods: by receiving an LE_ARP RESPONSE or LE_NARP_REQUEST that applies to the entry; by learning the MAC source address and/or Routing Information Field from an incoming data frame on a Data Direct VCC.

### 7.1.14 Elimination of LE_ARP cache entries

In order to avoid unnecessary LE_ARP activity, an LE Client SHOULD remove any LE_ARP cache entry which has not been used to forward a data frame for a long period of time, whether or not that entry has been recently verified.

### 7.1.15 Aging Time

An LE Client MUST NOT use any entry in its LE_ARP cache (C16) for a Route Descriptor or unicast MAC address after the Aging Time (C17) has elapsed since the last verification of that entry. (Creation of an LE_ARP cache entry via LE_ARP is a form of verification.)

### 7.1.16 Topology Change Flag (C19)

An LE Client whose Topology Change flag (C19) is true SHOULD NOT use any non-local entry in its LE_ARP cache C16 for a unicast MAC address after the Forward Delay Time C18 has elapsed since the last verification of that entry.

### 7.1.17 Updating Topology Change Flag (C19)

Whenever an LE Client receives an LE_TOPOLOGY_REQUEST frame from the LE Server with the Topology Change bit set in its FLAG field, the LE Client MUST set its Topology Change flag (C19) to “true”.An LE Client MUST set its Topology Change flag (C19) to “false” whenever it receives an LE_TOPOLOGY_REQUEST frame from the LE Server whose FLAG field has the Topology Change bit clear.

### 7.1.18 LE_ARP_REQUEST Frequency

If an LE Client issues an LE_ARP_REQUEST for an LE_ARP Cache entry that is soon to expire, it MUST NOT issue more than C13 LE_ARP_REQUESTs before the entry expires.

### 7.1.19 Generation of LE_TOPOLOGY_REQUESTs

An LE Client that is acting as an IEEE 802.1D transparent bridge MUST send one LE_TOPOLOGY_REQUEST to its LE Server for every Configuration BPDU it sends to the BUS. The Topology Change bit in the FLAGS field in the LE_TOPOLOGY_REQUEST MUST be set to the same value as the Topology Change bit in the Configuration BPDU.

### 7.1.20 Local Management Directives

A LE Client MAY send LE_TOPOLOGY_REQUESTs to its LE Server in response to local management directives or network management directives. However, such LE_TOPOLOGY_REQUESTs MAY be sent while bridge clients are still active, and MAY have either value for the Topology Change bit. Note that adverse operation may result in all LE Clients when more than one LE Client generates LE_TOPOLOGY_REQUESTs and the topology change status values do not agree.

### 7.1.21 Spanning Tree Configuration BPDUs

An LE Client that is acting as an IEEE 802.1D bridge MAY choose to base its LAN Emulation Topology Change state on Spanning Tree configuration BPDUs, rather than on received LE_TOPOLOGY_REQUESTs. This rule[19] applies only to IEEE 802.1D bridge clients and takes priority over rules 7.1.16 and 7.1.17.

### 7.1.22 Unresolved LE_ARP_REQUESTs

An LE Client MUST NOT retry an LE_ARP_REQUEST for a given frame's LAN destination more than C13 times after the first LE_ARP_REQUEST for that same frame's LAN destination. By the time the last-allowed LE_ARP_REQUEST has timed out the LE Client SHOULD have transmitted the frame to the BUS. This does not preclude sending that frame to the BUS before the failure of the address resolution protocol. However, if this transmission would violate the restrictions imposed by the Maximum Unknown Frame Count C10 and Maximum Unknown Frame Time C11, the data frame MUST be discarded.

### 7.1.23 Eavesdropping

An LE Client MAY incorporate information from any LE_ARP_REQUEST or LE_ARP_RESPONSE, even those not directed to that LE Client, into its LE_ARP Cache C16.

### 7.1.24 Generating an Unsolicited LE_NARP_REQUEST

An LE Client MAY generate unsolicited LE_NARP_REQUESTs to allow other clients to learn about a changed Remote LAN-ATM address binding. A LE_NARP_REQUEST advertises that the generating LEC believes that an old binding between TARGET-LAN-DESTINATION and TARGET-ATM-ADDRESS is no longer valid because it is now representing the LAN destination at SOURCE-ATM-ADDRESS.

### 7.1.25 LE_NARP Time Limits

LE_NARP_REQUEST generation is subject to the same repeat/time limits described above for LE_ARP_REQUESTs.

### 7.1.26 LE_NARP_REQUESTs and Topology Changes

An LE Client MUST NOT generate any LE_NARP_REQUESTs when its Topology Change flag (C19) is set i.e. when there are more general topology changes occurring in the network.

### 7.1.27 LE_NARP_REQUESTs not for Local Bindings

An LE Client MUST NOT generate LE_NARP_REQUESTs for Local bindings, i.e. for bindings for which it uses the LE_REGISTER/UNREGISTER mechanism.

---

[19]The reason for this rule is as follows. An LE Client that is acting as an IEEE 802.1D bridge might maintain a combined bridge forwarding database and LE_ARP cache, aging out MAC addresses as appropriate for the Spanning Tree Protocol. At best, received LE_TOPOLOGY_REQUESTs will duplicate information which should be available in configuration BPDUs. At worst, they might tell the bridge to change aging times in some way that is inconsistent with the Spanning Tree protocol. For instance, a network manager might try to clear the Topology Change mode while the IEEE 802.1D bridge client is still participating in the emulated LAN. To allow the implementation of combined bridge forwarding databases and LE_ARP caches without requiring bridge clients to violate the IEEE 802.1D bridging specification, it is necessary to give Spanning Tree precedence in the event of such conflicts.

### 7.1.28 Previously Represented LAN Destination

An LE Client generating an LE_NARP_REQUEST MUST include the ATM address of the LEC which was previously representing the LAN destination.

### 7.1.29 Remote Binding Advertised

An LE Client which generates a LE_NARP_REQUEST implicitly advertises the fact that it is now representing the TARGET-LAN-DESTINATION with SOURCE-ATM-ADDRESS as a Remote binding.

### 7.1.30 Updating the LE_ARP Cache

LE Clients MAY update their LE_ARP cache from the binding information in received LE_NARP_REQUESTs.[20][21]

## 7.2 Address Resolution - LE Service View

### 7.2.1 Address Resolution VCC

An LE Server MUST ignore any LE_ARP_REQUEST or LE_ARP_RESPONSE received on any VCC other than a Control Direct VCC.

### 7.2.2 Transmitting LE_ARP_REQUESTs and LE_ARP_RESPONSEs

If an LE Server sends an LE_ARP_REQUEST or LE_ARP_RESPONSE to an LE Client, it must send it on the Control Direct VCC (to that client) or the Control Distribute VCC (to that client and/or other LE Clients).

### 7.2.3 Responding to Known LE_ARP_REQUESTs

An LE Server MAY respond to an LE_ARP_REQUEST for a registered LAN destination with the information obtained when that LAN destination was registered. The Remote Address bit Flag in this LE_ARP_RESPONSE must be cleared.

### 7.2.4 Not Responding to Unknown LE_ARP_REQUESTs

An LE Server MUST NOT respond to an LE_ARP_REQUEST for a LAN destination that has not been registered.

### 7.2.5 Not Forwarding Known LE_ARP_REQUESTs

If an LE Server responds to an LE_ARP_REQUEST, it MUST NOT also forward that LE_ARP_REQUEST to any LE Client.

---

[20] IEEE 802.1d bridges which learn from packet source addresses will probably not gain very much from either generating or learning from LE_NARP_REQUESTs. They achieve a better effect by performing source-address learning.

[21] Use of LE_NARP_REQUESTs does not resolve the problem of having to deal with LECs using long timeouts and not listening to LE_NARP_REQUESTs: LECs may still receive more than one LE_NARP_REQUEST with contradictory information at regular intervals until the responders resolve who is really representing the LAN Destination, just as they may also receive more than one LE_ARP_RESPONSE for a single LE_ARP_REQUEST for a destination that multiple LECs believe is theirs to represent.

### 7.2.6 Forwarding Unknown LE_ARP_REQUESTs

If an LE Server does not respond to an LE_ARP_REQUEST, it MUST forward that LE_ARP_REQUEST.

### 7.2.7 Forwarding LE_ARP_REQUESTs

If the LE Server does not respond to an LE_ARP_REQUEST for a registered LAN destination, then it MUST forward the LE_ARP_REQUEST to the LE Client that registered that LAN destination. The LE Server MAY also forward that LE_ARP_REQUEST to other LE Clients, as well.

### 7.2.8 Forwarding LE_ARP_REQUESTs to Proxy LE Clients

An LE Server MUST forward any LE_ARP_REQUEST for an unregistered LAN destination to all LE Clients that successfully joined as proxy agents (Proxy Flag in the LE_JOIN_REQUEST). The LE Server MAY also forward that LE_ARP_REQUEST to other LE Clients, as well.

### 7.2.9 Responding to LE_ARP_REQUESTs for Broadcast MAC Address

The LE Service MUST respond to every LE_ARP request for the broadcast MAC address with the ATM address of the BUS.

### 7.2.10 Forwarding LE_TOPOLOGY_REQUESTs

The LE Server MUST send one LE_TOPOLOGY_REQUEST to all LE Clients for each valid LE_TOPOLOGY_REQUEST received from any client. The Topology Change bit in the FLAGS field in the LE_TOPOLOGY_REQUEST SHOULD match that of the LE_TOPOLOGY_REQUEST last received by the LE Server.

### 7.2.11 Initiating LE_TOPOLOGY_REQUESTs

An LE Server MAY issue LE_TOPOLOGY_REQUESTs without receiving any LE_TOPOLOGY_REQUESTs. For example, an LE Server may wish to clear the LE Clients' Topology Change flags if all its transparent bridge LE Clients cease operations while the other LE Clients' Topology Change flags are set.

## 7.3 Address Resolution Frames

The LE_ARP frames used to resolve LAN destinations to ATM addresses, and thus VCCs, are described in Table 20.

**Table 20. LE_ARP Frame Format**

| Offset | Size | Name | Function |
|---|---|---|---|
| 0 | 2 | MARKER | Control Frame = X"FF00" |
| 2 | 1 | PROTOCOL | ATM LAN Emulation protocol = X"01" |
| 3 | 1 | VERSION | ATM LAN Emulation protocol version = X"01" |
| 4 | 2 | OP-CODE | Type of request:<br>X"0006" LE_ARP_REQUEST<br>X"0106" LE_ARP_RESPONSE |
| 6 | 2 | STATUS | Always X"0000" in requests. In Responses: See Table 13 for a list of values. |
| 8 | 4 | TRANSACTION-ID | Arbitrary value supplied by the requester. |
| 12 | 2 | REQUESTER-LECID | LECID of LE Client issuing the LE_ARP request. |
| 14 | 2 | FLAGS | Each bit of the FLAGS field has a separate meaning if set:<br>X"0001" Remote Address. The TARGET-LAN-DESTINATION is not registered with the LE Server. |
| 16 | 8 | SOURCE-LAN-DESTINATION | Source MAC address from data frame that triggered this LE_ARP sequence. May be encoded with "not present" LAN Destination tag. |
| 24 | 8 | TARGET-LAN-DESTINATION | Destination unicast MAC address or next Route Descriptor for which an ATM address is being sought. |
| 32 | 20 | SOURCE-ATM-ADDRESS | ATM address of originator of LE_ARP request. |
| 52 | 4 | RESERVED | Always X"00" when sent, ignored on receipt. |
| 56 | 20 | TARGET-ATM-ADDRESS | X"00" in LE_ARP request. ATM address of LE Client responsible for target LAN destination in LE_ARP response. |
| 76 | 32 | RESERVED | Always X"00" when sent, ignored on receipt. |

## 7.4 LE_NARP Frame Format

**Table 21. LE_NARP Frame Format**

| Offset | Size | Name | Function |
|---|---|---|---|
| 0 | 2 | MARKER | Control Frame = X”FF00” |
| 2 | 1 | PROTOCOL | ATM LAN Emulation protocol = X”01” |
| 3 | 1 | VERSION | ATM LAN Emulation protocol version = X”01” |
| 4 | 2 | OP-CODE | Type of request:<br>X”0008” LE_NARP_REQUEST |
| 6 | 2 | STATUS | Always X”0000”. |
| 8 | 4 | TRANSACTION-ID | Arbitrary value supplied by the requester. |
| 12 | 2 | REQUESTER-LECID | LECID of LE Client issuing the LE_NARP request. |
| 14 | 2 | FLAGS | Always X’00’’. |
| 16 | 8 | SOURCE-LAN-DESTINATION | Not used. Encoded as X’00’. |
| 24 | 8 | TARGET-LAN-DESTINATION | Destination unicast MAC address or next Route Descriptor for which the target ATM address no longer applies. |
| 32 | 20 | SOURCE-ATM-ADDRESS | ATM address of originator of LE_NARP request. |
| 52 | 4 | RESERVED | Always X”00” when sent, ignored on receipt |
| 56 | 20 | TARGET-ATM-ADDRESS | ATM address of LE Client which was previously representing the target LAN destination. |
| 76 | 32 | RESERVED | Always X”00” when sent, ignored on receipt. |

## 7.5 Topology Change Frames

Topology Change frames are used for notifying all LE Clients that their cached LE_ARP information for remote LAN destinations may be incorrect. They are described in Table 22.

**Table 22. Topology Change Frame Format**

| Offset | Size | Name | Function |
|---|---|---|---|
| 0 | 2 | MARKER | Control Frame = X"FF00" |
| 2 | 1 | PROTOCOL | ATM LAN Emulation protocol = X"01" |
| 3 | 1 | VERSION | ATM LAN Emulation protocol version = X"01" |
| 4 | 2 | OP-CODE | Type of request:<br>X"0009" LE_TOPOLOGY_REQUEST |
| 6 | 2 | STATUS | Always X"0000" |
| 8 | 4 | TRANSACTION-ID | Arbitrary value supplied by the requester. |
| 12 | 2 | REQUESTER-LECID | LECID of LE Client issuing the Topology Change request. |
| 14 | 2 | FLAGS | Each bit of the FLAGS field has a separate meaning if set:<br>X"0100" Topology Change Flag. A network topology change is in progress. |
| 16 | 92 | RESERVED | Always X"00" when sent, ignored on receipt. |

# 8. Data Transfer Protocol and Procedures

## 8.1 Data Transfer Phase - LE Client Protocol Mechanisms

### 8.1.1 Data Frame VCC

An LE Client MUST NOT send data frames over any VCCs except Data Direct VCCs and Multicast Send VCCs.

### 8.1.2 Data Frame Frequency

An LE Client MUST send no more than C10 frames within a time period of C11 seconds to a given unicast LAN destination to the Broadcast and Unknown Server.

### 8.1.3 Use of LECID

An LE Client MUST either insert its own LECID (parameter C14) or X'0000' as the first two octets of any data frame it sends on a data connection.

### 8.1.4 Discarding Frames Based on LECID

An LE Client MAY examine all frames received on any Data VCC, and discard any frames having its own LECID.

### 8.1.5 Maximum Frame Sizes

An LE Client MUST NOT send any frame that is larger than its Maximum Frame Size parameter C3 on any data VCC.

### 8.1.6 Forwarding Unicast Data Frames

An LE Client MUST NOT forward any data frame received on a Data Direct VCC or Multicast Forward VCC to another Data Direct VCC or a Multicast Send VCC for the same emulated LAN. This restriction applies to LE Clients in bridges, as well as non-bridges. This is a restriction on the LE Client itself, not the higher-level entities making use of the LE Client's facilities.

### 8.1.7 Filtering Data Frames

Each client is responsible for filtering out only those destinations which it needs to deliver up to higher layers (See service definition interface, as well as parameter C15).

### 8.1.8 Minimum IEEE 802.3 Data Frame Size

An LE Client MUST NOT transmit an IEEE 802.3/Ethernet data frame with an AAL5 SDU size of less than 62 octets.

### 8.1.9 Minimum IEEE 802.5 Data Frame Size

An LE Client MUST NOT transmit an IEEE 802.5 data frame with an AAL5 SDU size of less than 16 octets.

### 8.1.10 Maximum Data Frame Size (C3)

An LE Client using SVCs MUST signal the Maximum Data Frame Size C3 as both the Forward Maximum CPCS-SDU size and the Backward Maximum CPCS-SDU size for establishing Data Direct, Multicast Send, and Multicast Forward VCCs. An LE Client MUST refuse to establish such VCCs with any other Forward Maximum CPCS-SDU size or Backward Maximum CPCS-SDU size.

### 8.1.11 Duplicate Incoming Data Direct VCC

When a client receives an incoming Data Direct VCC connection request from an ATM address to which it already has a connection, it SHOULD accept that request. When two Data Direct VCCs are established via signaling between the same two ATM addresses, an LE Client MUST send frames only on the connections for which the SETUP message was initiated by the numerically lower ATM address. This may cause the duplicate VCC to be aged out due to inactivity without the possibility of a race condition removing both VCCs.

### 8.1.12 Data Direct VCC Termination

An LE Client SHOULD terminate Data Direct VCCs over which no traffic has been sent or received for a minimum length of time C12. It MUST indicate the UNI Cause Code (for example UNI 3.0 Cause “Normal, Unspecified” Cause 31) so that the other end can distinguish between accidental failures and deliberate dropping of the VCC.

### 8.1.13 Terminating ELAN Membership

If the LE Client detects that the Control Direct VCC or the Control Distribute VCC is released at any time other than those specified for the “Join” phase above, then the LE Client MUST terminate its ELAN membership.

## 8.2 Data Direct Call Establishment

### 8.2.1 Called Client Ready to Receive Frames

A called LE Client accepting a Data Direct VCC MUST NOT issue a Connect signaling message until it is ready to receive frames on that VCC. It then MAY establish a time-out C28.

### 8.2.2 Calling Client Action on Receiving CONNECT Message

An LE Client that receives a Connect message for a requested Data Direct VCC MUST issue a READY_IND message as soon as it is ready to accept frames on that VCC. It then MAY also send data or flush frames. It MUST NOT send any frames before it is ready to accept frames. Data or Flush frames MAY be sent before the READY_IND message.

### 8.2.3 Called Client Terminates C28

If the called LE Client implements time-out C28, then it MUST terminate time-out C28 upon receipt of a READY_IND frame. The called LE Client MAY also terminate time-out C28 upon receipt of any valid frame on the VCC. If the called LE Client transmits any data or control frames before receiving the READY_IND or valid data, then this data may be discarded by the network as described in Section 2.3.4.1.

### 8.2.4 Called Client Timer C28 Expires

If time-out C28 expires, the LE Client MAY transmit data or a READY_QUERY frame. It MAY re-establish time-out C28, and repeat this as desired.

### 8.2.5 READY_IND Response to READY_QUERY

An LE Client receiving a READY_QUERY frame on any operational Data Direct VCC MUST reply with a READY_IND frame.

### 8.2.6 Ready Frame Format

**Table 23. Ready Frame Format**

| Offset | Size | Name | Function |
|---:|---:|---:|---|
| 0 | 2 | MARKER | Control Frame = X"FF00" |
| 2 | 1 | PROTOCOL | ATM LAN Emulation Protocol=X"01" |
| 3 | 1 | VERSION | ATM LAN Emulation Protocol=X"01" |
| 4 | 2 | OP-CODE | X"0003": READY_QUERY<br>X"0103": READY_IND |

## 8.3 Data Transfer Phase - BUS Protocol Mechanisms

### 8.3.1 Maximum Frame Age S5

The BUS MUST discard any frame which has been held in the BUS for longer than the Maximum Frame Age S5, regardless of other provisions in Section 8.3.

### 8.3.2 BUS Forwarding Data Frames

The BUS SHOULD transmit any valid data frame received from a Multicast Send VCC on either a Multicast Send VCC or all Multicast Forward VCCs.

### 8.3.3 BUS Forwards on Multicast Forward VCC

All broadcast and multicast traffic received by a BUS MUST be forwarded onto the Multicast Forward VCC to each client.

## 8.4 Data Transfer Phase - LES Protocol Mechanisms

### 8.4.1 LE Server Detects VCC Release and Terminates ELAN Membership

If the LE Server detects that the Control Direct VCC or the Control Distribute VCC to/from a client are released, then the LE Server MUST terminate the client's emulated ELAN membership.

### 8.4.2 LE Server Terminates ELAN Membership

At any time during the data transfer phase, the LE Server may close the Control Direct or the Control Distribute VCC to a client with Reason Code 31 in order to terminate the client's emulated ELAN membership.

## 8.5 Delivery of Token Ring Frames

When the LEC has a frame to send it examines both the frame's routing information field and the frame's destination MAC address field to determine where to send the frame and if it should LE_ARP. An LE Client emulating Token Ring MUST support address resolution of route descriptors. The Appendix provides information on determining if a frame is a source routed frame (contains an RI field), determining the RI type, and determining the next RD. A frame is classified as being a Non Source Routed (NSR), All Routes Explorer (ARE), Spanning Tree Explorer (STE), or a Specifically Routed Frame (SRF).

The following requirements are based on NSR traffic being sent on a VCC determined by the frame's DA and source route traffic being sent on a VCC determined by the frame's RI field and DA field. The resulting actions and operations are summarized in Table 24 and Table 25.

### 8.5.1 Bridge Response to LE_ARP

A source routing bridge MUST respond to an LE_ARP request for one of its RDs.

If an LE Client is emulating Token Ring then it MUST function as follows when it sends a data frame.

### 8.5.2 Initial Frame Handling

The LE Client sends a NSR frame on the data direct VCC for the frame's DA. If the VCC is unknown, then the LE Client MUST generate an LE_ARP request for the DA and the frame may be sent to the BUS.

### 8.5.3 Multicast Frames

The LE Client sends a multicast frame to the BUS.

### 8.5.4 ARE, STE Frames

The LE Client sends an ARE or a STE frame to the BUS. The appropriate Route Descriptor needs to be added to the RIF and the LTH needs to be recalculated.

### 8.5.5 RI Field Length

The LE Client that is not part of a bridge may assume that if the RI field of an SRF frame has a length of 6 or more, then the local SEGMENT ID is the first RD in the list and the "next RD" is

SEGMENT ID of the second RD in the list (order depending on the RI direction bit) and the Bridge number located between those two SEGMENT IDs (refer to Appendix 12.3.2.3 for information on determining the next RD). This implies that the last hop is not this ELAN.

### 8.5.6 Invalid RI Field

The LE Client MAY discard a frame with an invalid RI field. This includes an RI field with an odd length and a SRF frame with an RI field that does not contain an RD matching the emulated LAN's SEG_ID.

### 8.5.7 No Hops

An SRF frame with a RI length less than 6 contains no hops. The LE Client MUST send a SRF frame with no hops on the direct VCC for the frame's DA. If the VCC is unknown, then the LE Client MUST generate an LE_ARP request for the DA and the frame may be sent to the BUS.

### 8.5.8 Last Hop on Same ELAN

If the last hop of a SRF frame is this ELAN, then the LE Client MUST send the frame on the direct VCC for the frame's DA. If the VCC is unknown, then the LE Client MUST generate an LE_ARP request for the DA and the frame may be sent to the BUS.

### 8.5.9 Next RD

If the length of a SRF frame's RI field is 6 or more and this ELAN is not the last hop, then the LE Client MUST send the frame on the direct VCC for the "next_RD". If the VCC for the "next_RD" is unknown, then the LE Client MUST generate an LE_ARP request for the "next_RD" and the frame may be sent to the BUS.

**Table 24. LAN Destination of Token Ring Frames**

| Frame Type | Destination Address: | |
|---|---|---|
| | **Unicast:** | **Multicast:** |
| Non source Routed (NSR) | VCC(DA) | BUS |
| Explorer (ARE or STE) | BUS | BUS |
| Specifically Routed (SRF) | Table 25 | BUS |

**VCC(DA):** Use the frame's DA as the basis for delivering the frame. If the VCC is not known, LE_ARP with the DA as the target address.

**BUS:** Send to the BUS.

**Table 25**: If this is the last hop then use the DA otherwise use the RD as summarized in Table 25 (which follows).

**Table 25. LAN Destination for Unicast SRF Token Ring Frames**

| Routing Information | Destination: |
|---|---|
| No hops | VCC(DA) |
| Last hop | VCC(DA) |
| Not last hop | VCC(next_RD) |

**VCC(DA):** Use the frame's DA as the basis for delivering the frame. If the VCC is not known, LE_ARP with the DA as the target address.

**VCC(next_RD):** Use the next_RD as the basis for delivering the frame. If the VCC is not known, LE_ARP with the next_RD as the target address.

## 8.6 Maximum Data Frame Size.

As stated in Section 1, the main objective of the LAN emulation service is to enable existing applications to access an ATM network as if they were running over traditional LANs. An emulated LAN corresponds more or less to a single LAN segment, and an LE Client corresponds more or less to a NIC. In a traditional LAN, MTU is the same for all adapters on a given LAN segment regardless of destination address. In order to support conventional LAN services where the maximum frame size is the same for all adapters on a given LAN segment, all LE clients that belong to a given emulated LAN use the same maximum AAL-5 SDU size for all their connections.

The LAN Emulation components MUST use one of the maximum AAL-5 SDU sizes of 1516, 4544, 9234, and 18190 octets for data frames - and for establishing data SVCs. These sizes have been selected to support bridging to existing LANs. The basis for these sizes follow and is summarized in 12.4.1 in the Appendix.

**Table 26. Maximum Data Frame Size**

| AAL-5 SDU Max. octets | AAL-5 PDU Max. octets | Basis for SDU Size |
|---|---|---|
| 1516 | 1536 (32 cells) | IEEE 802.3/Ethernet |
| 4544 | 4560 (95 cells) | IEEE 802.5 Token Ring 4Mbps |
| 9234 | 9264 (193 cells) | RFC 1626 |
| 18190 | 18240 (380 cells) | IEEE 802/5 Token Ring 16Mbps |

# 9. Flush Message Protocol, Procedures and Frame Formats

A LAN emulation client may send unicast frames to the same destination LAN address via the Broadcast and Unknown Server and via a data direct VCC at different times. The flush message protocol allows the sender to avoid the possibility of delivering frames out of order caused by having two paths. When switching from the “old” path to the “new” path, the sender first transmits a flush message down the old path, then sets appropriate table entries so that any further frames for the given LAN destination will be held (or discarded) at the sender and not transmitted. The flush message is a special frame distinguishable from a data frame by having a reserved value (X’FF00’, marker for control frame) in the LAN emulation data frame header in place of the LECID of the sender. The flush message must be returned to the sender by the receiving client via control VCCs. Once the sender receives the returned flush message, it knows that the old path is clear of data for that LAN destination, and it can start using the new path.

Note that Flush messages may be exchanged between LE Clients, but are never forwarded by a bridge to a legacy LAN.

## 9.1 Flush Message Protocol and Procedures

The Flush Message Protocol ensures that data frames are delivered in the same order that they were transmitted. Flush Protocol messages are of two types:

- **LE_FLUSH_REQUEST:** Flush requests are sent by an LE Client down a Data Direct VCC or Multicast Send VCC to ensure that all data frames in transit on that path have reached their destination LE Client.
- **LE_FLUSH_RESPONSE:** Flush responses are returned by LE Clients via Control Direct VCCs and Control Distribute VCCs in response to received LE_FLUSH_REQUESTs.

These frame formats are described in Table 27.

### 9.1.1 Mandatory Protocol

The following requirements apply to any LE Client or LE Service Component, whether or not the LE Client chooses to implement the flush message protocol for ensuring the order of delivery for data frames.

#### 9.1.1.1 LE_FLUSH_REQUEST VCC

An LE Client MAY send an LE_FLUSH_REQUEST over any Data Direct VCC or Multicast Send VCC.

#### 9.1.1.2 Ignoring LE_FLUSH_REQUEST Prior to Join

The points below notwithstanding, an LE Client MUST NOT respond to an LE_FLUSH_REQUEST arriving on a VCC associated with an ATM Emulated LAN for which the client has not successfully completed the Join procedure.

#### 9.1.1.3 Forwarding LE_FLUSH_REQUESTs

The BUS MUST distribute any LE_FLUSH_REQUEST received via its Multicast Send VCC to at least the LE Client specified in that frame's target ATM address via its Multicast Send VCC or its Multicast Forward VCC(s), if that LE_FLUSH_REQUEST does not specify the BUS's ATM address as its target. The BUS MAY distribute requests targeted to its own ATM addresses, as well.

#### 9.1.1.4 Responding to LE_FLUSH_REQUEST on Control Direct VCC

An LE Service Component MAY respond to an LE_FLUSH_REQUEST directed to that LE Service component. If it does respond, it MUST respond to an LE_FLUSH_REQUEST specifying its own ATM address as the target in such a manner that the LE_FLUSH_RESPONSE will be returned to the requester on a Control VCC.

#### 9.1.1.5 Responding to LE_FLUSH_REQUEST

An LE Client MUST respond to an LE_FLUSH_REQUEST it receives on any VCC if that LE_FLUSH_REQUEST specifies that LE Client's ATM address as its target, by returning an LE_FLUSH_RESPONSE to the sender of the LE_FLUSH_REQUEST. The responding LE Client MUST use its Control Direct VCC to send the response.

#### 9.1.1.6 Ignoring LE_FLUSH_REQUEST

An LE Client MUST NOT respond to an LE_FLUSH_REQUEST with a TARGET-ATM-ADDRESS that is not its own.

#### 9.1.1.7 Forwarding LE_FLUSH_RESPONSE

The LE Server MUST relay LE_FLUSH_RESPONSEs whose SOURCE-ATM-ADDRESS field matches the information in the REQUESTER-LECID field to at least the LE Client specified in the REQUESTER-LECID field.

### 9.1.2 Optional Protocol

An LE Client MAY have two paths available to reach any given LAN destination, either a Data Direct VCC, or the Multicast Send VCC to the Broadcast and Unknown Server. In order to ensure that data frames sent to a given LAN destination are delivered in the same order sent, an LE Client SHOULD perform the following flush message protocol whenever it switches data paths to reach that LAN destination.

- NOTE: In this section, the words SHOULD, MUST, SHOULD NOT, and MUST NOT apply only to LE Clients that choose to implement the flush message protocol.

#### 9.1.2.1 Transaction ID in LE_FLUSH_REQUEST

The LE Client MUST send an LE_FLUSH_REQUEST on the old data VCC (either a Data Direct VCC or a Multicast Send VCC). This request MUST contain a Transaction ID not currently in use by the LE Client. (The Transaction ID MAY, for example, be a counter that is incremented each time a new Transaction ID is needed.) It MUST also contain the requester's ATM address and the ATM address of the target LE Client to which the path is being switched.

#### 9.1.2.2 Changing Data Paths

Until the LE Client originating the LE_FLUSH_REQUEST receives the LE_FLUSH_RESPONSE with a matching Transaction ID, or until it times out (time C21) waiting for that response, it MUST NOT send any data frames over the old data path destined for the same LAN destination that prompted sending the LE_FLUSH_REQUEST.

#### 9.1.2.3 Discarding or Holding Data Frames

Data frames destined for a LAN destination that is awaiting an LE_FLUSH_RESPONSE MAY be discarded, or MAY be held by the LE Client until the LE_FLUSH_RESPONSE is received or timed out.

#### 9.1.2.4 Data Frame Handling

If an LE Client times out an LE_FLUSH_REQUEST, and it is holding data frames awaiting the request's LE_FLUSH_RESPONSE, those held frames MUST either be discarded or sent down the old data path. The LE Client MAY then send another LE_FLUSH_REQUEST with a new Transaction ID. If it does so, it MUST ignore any LE_FLUSH_RESPONSE received with the old Transaction ID. Note that limitations on the frequency of sending unicast frames to the BUS may require some held frames to be discarded.

#### 9.1.2.5 Transmitting Held Frames

Once the LE_FLUSH_RESPONSE is received, the LE Client MUST transmit any held data frames on the new data path before transmitting any further frames on the new path.

#### 9.1.2.6 Switching Over Paths Without Flush

Regardless of the provisions of the rest of Section 9.1, when switching from the old path to a new path, if an LE Client has not transmitted a data frame to a particular LAN destination via the old path for a period of time greater than or equal to the Path Switching Delay C22, then it MAY start using the new path without employing the Flush protocol.

## 9.2 Flush Frames

Flush messages are used for ensuring the in-order delivery of data frames. They are described in Table 27.

**Table 27. Flush Frame Format**

| Offset | Size | Name | Function |
|---|---|---|---|
| 0 | 2 | MARKER | Control Frame = X"FF00" |
| 2 | 1 | PROTOCOL | ATM LAN Emulation protocol = X"01" |
| 3 | 1 | VERSION | ATM LAN Emulation protocol version = X"01" |
| 4 | 2 | OP-CODE | Type of request:<br>X"0007" LE_FLUSH_REQUEST<br>X"0107" LE_FLUSH_RESPONSE |
| 6 | 2 | STATUS | Always X"0000" in requests. In Responses: See Table 13 for a list of values. |
| 8 | 4 | TRANSACTION-ID | Arbitrary value supplied by the requester and returned by the responder. |
| 12 | 2 | REQUESTER-LECID | LECID of LE Client issuing the flush request. |
| 14 | 2 | FLAGS | Always 0 sending, ignored on receipt. |
| 16 | 16 | SOURCE-LAN-DESTINATION | Always X"00" when sent, ignored on receipt. |
| 32 | 20 | SOURCE-ATM-ADDRESS | ATM address of originator of flush request. |
| 52 | 4 | RESERVED | Always X"00" when sent, ignored on receipt. |
| 56 | 20 | TARGET-ATM-ADDRESS | ATM address of LE Client to which flush request is directed. |
| 76 | 32 | RESERVED | Always X"00" when sent, ignored on receipt. |

# 10. Termination Protocol and Procedures

Procedures to be used by an LE Client and an LE Server when terminating an LE Client's emulated LAN membership are defined to ensure that clients may be effectively managed by the LAN Emulation service.

## 10.1 Termination - LE Client View

### 10.1.1 Releasing SVCs

Whenever the preceding sections on Protocols and Procedures indicate that an LE Client terminate a procedure or an LE Client's connection to the emulated LAN, with the exceptions defined below, all SVCs associated with this LE Client, including all Control VCCs, all Data Direct VCCs and any Data VCCs to/from the BUS, MUST be released and any PVCs MUST no longer be used. These call release operations are supported by UNI signaling protocols. The client MUST also flush any dynamically-learned entries from its LE_ARP cache.

### 10.1.2 LE Client Returns to Configuration Phase

If a client's membership is terminated by an explicit server release of Control Direct or Control Distribute VCCs signaled with UNI Reason Code (for example UNI 3.0/3.1 "Normal, Unspecified" - Cause 31, or UNI 3.1 "Normal Call Clearing, Cause 16"), the client MUST perform the functions described in Section 10.1.1 and return to the Initial State.

## 10.2 Termination - LE Service View (LES or BUS)

### 10.2.1 LE Server and BUS Actions

Whenever the preceding sections on Protocols and Procedures indicate that an LE Server or BUS terminate a procedure or an LE Client's connection to the emulated LAN, with the exceptions defined below, all SVCs associated with this LE Client MUST be released with the UNI Reason Code (for example UNI 3.0/3.1 "Normal, Unspecified" - Cause 31, or UNI 3.1 "Normal Call Clearing, Cause 16"), and the PVCs are no longer used. The LE Server MUST clear any registered LAN destinations and cached address bindings associated with that LE Client.

### 10.2.2 SVC Re-establishment

An LE Server or BUS MUST NOT attempt to re-establish the SVCs to any LE Client after they have been released for any reason. It is the LE Client's responsibility to reinitiate its connection to the emulated LAN.

### 10.2.3 LE Client May Have Service Refused or Terminated

The LAN Emulation Service MAY terminate and/or refuse service to any LAN Emulation Client whose behavior violates the normative rules in the LAN Emulation specification.

# 11. Usage of ATM Addresses

In order to support SVC-only and mixed PVC/SVC environments, ATM LAN Emulation components must be able to reliably associate incoming LAN Emulation SVCs with particular emulated LANs and functions within each emulated LAN.

In general, there are two ways of identifying such associations.

1. Explicit identification. The SVC Setup message, or some protocol sent over the new VCC, identifies the emulated LAN and the usage of a VCC.
2. Implicit identification. ATM addresses of LE components are allocated in such a way that they help to identify emulated LANs and LAN functions.

LAN Emulation uses a combination of these techniques. Call setup using UNI signaling procedures MUST indicate whether LAN Emulation SVCs carry LE Control traffic, LE Multicast Ethernet/IEEE 802.3 traffic, LE Multicast IEEE 802.5 traffic, LE Data Direct Ethernet/IEEE 802.3 traffic, or LE Data Direct IEEE 802.5 traffic (Section 3.3.2.7.4). In addition, LE Clients, LE Servers, and Broadcast and Unknown Servers rely upon implicit identification to direct SVCs to the proper LAN Emulation component.

For the purposes of the normative statements below, a LAN Emulation component is defined to be any of the following: a LAN Emulation Client, a LAN Emulation Server, a Broadcast and Unknown Server, or a LAN Emulation Configuration Server.

## 11.1 ATM Address Uses Summary

LAN Emulation components MUST adhere to Table 28 when assigning and utilizing ATM addresses. In Table 28, there are four ATM addresses, **X**, **C**, **B**, and **S**, plus other addresses marked “any”. Except for addresses B and S (which may or may not be equal to each other), each of the four named ATM addresses MUST be distinct. Each of the four MUST be unique among all LE Clients’ ATM addresses.

**Table 28**. **ATM Address Uses**

| Usage | Allowed ATM Address |
|---|---|
| A particular LE Client’s primary ATM address | **X** |

| | |
|---|---|
| LECS's ATM address as seen by LE Client X | **C** |
| BUS's ATM address as seen by LE Client X (maybe == **S**) | **B** |
| LES's ATM address as seen by LE Client X (maybe == **B**) | **S** |
| Data Direct VCC calling party | any* |
| Data Direct VCC called party | any |
| Configuration Direct VCC calling party | any |
| Configuration Direct VCC called party | **C** |
| Control Direct VCC calling party | **X** |
| Control Direct VCC called party | **S** |
| Multicast Send VCC calling party | **X** |
| Multicast Send VCC called party | **B** |
| Multicast Forward VCC calling party | any* |
| Multicast Forward VCC called party | **X** |
| Control Distribute VCC calling party | any* |
| Control Distribute VCC called party | **X** |
| Configure Request SOURCE-ATM-ADDRESS | **X** |
| Configure Response TARGET-ATM-ADDRESS | **S** |
| Join Request SOURCE-ATM-ADDRESS | **X** |
| LE_ARP Response for the BUS: TARGET-ATM-ADDRESS | **B** |

* any of the ATM addresses belonging to the entity initiating the connection.

## 11.2 LE Client ATM Addresses

### 11.2.1 Primary ATM Address

An operational LE Client MUST have a primary ATM address, and MAY have other ATM addresses.

### 11.2.2 Use of Primary Address

An LE Client MUST use its primary ATM address for establishing the Control Direct VCC, for establishing the Multicast Send VCC, and as the SOURCE-ATM-ADDRESS in its LE_JOIN_REQUEST control frames. An LE Client MUST NOT change its primary ATM address without terminating its connection (if any) to the emulated LAN.[22]

### 11.2.3 Unique ATM Address

An operational LE Client MUST have unique ATM addresses. None of an operational client's ATM addresses may be shared with any other LAN Emulation component, even if two LAN Emulation components are co-located and share the use of a UNI. ATM hosts can therefore associate an incoming LAN Emulation SVC with the proper LE Client on the basis of the Called Party Number.

## 11.3 LE Server ATM Addresses

### 11.3.1 ATM Address Required

An operational LE Server MUST have at least one ATM address, and MAY have more.

### 11.3.2 Sharing ATM Address

An LE Server MAY share an ATM address with a Broadcast and Unknown Server on the same emulated LAN.

### 11.3.3 When Sharing ATM Address not valid

An operational LE Server MUST NOT share an ATM address with any LAN Emulation component other than a BUS, even if two LAN Emulation components are co-located and share the use of a UNI. In particular, two LE Servers for different emulated LANs MUST NOT share an ATM address.

## 11.4 Broadcast and Unknown Server ATM Addresses

### 11.4.1 BUS ATM Address Required

An operational Broadcast and Unknown Server MUST have at least one ATM address, and MAY have more.

### 11.4.2 Sharing an ATM Address

A Broadcast and Unknown Server MAY share an ATM address with a LE Server on the same emulated LAN.

[22] After a client leaves an emulated LAN, other clients CAN remember its ATM addresses in their LE_ARP caches until their cache entries time out (which might take several minutes). Hosts that want to reassign ATM addresses to a different LAN Emulation component ought to take precautions to prevent the establishment of Data Direct VCCs which cross two emulated LANs.

### 11.4.3 When Sharing ATM Address not valid

An operational Broadcast and Unknown Server MUST NOT share an ATM address with any LAN Emulation component other than a LES, even if two LAN Emulation components are co-located and share the use of a UNI. In particular, two Broadcast and Unknown Servers for different emulated LANs MUST NOT share an ATM address.

## 11.5 LE Configuration Server ATM Addresses

### 11.5.1 ATM Address Required

An operational LE Configuration Server MUST have at least one ATM address, and MAY have more.

### 11.5.2 Sharing an ATM Address

Several operational LE Configuration Servers MAY share a “well-known” ATM address, but any one LE Client is only aware of a single instance: this is ensured by the assumed definition of a "well-known" address. Note that it is highly desirable for the LE Configuration Service to be set up in such a way that clients who tear down and re-establish Configuration Direct SVCs see consistent information. How this is accomplished is beyond the scope of the LUNI.

### 11.5.3 When Sharing ATM Address not valid

An operational LE Configuration Server MUST NOT share an ATM address with a LE Client, LE Server, or Broadcast and Unknown Server.

# 12. Appendix

## 12.1 Network Management Considerations

Network management of a LAN Emulation Client is defined using SNMP MIBs. Several MIBs are relevant, including

- RFC 1573 - Evolution of the Interfaces Group of MIB-II.
- RFC 1695 - Definitions of Managed Objects for ATM Management, otherwise known as the AToM MIB.
- The LAN Emulation Client MIB.

## 12.2 ILMI Extensions for LAN Emulation (ILMI MIB)

UNI interface management is defined to use ILMI. This specification extends the ILMI MIB to provide access to the LECS address.[23]

```
ATM-FORUM-SRVC-REG DEFINITIONS ::= BEGIN

IMPORTS
        atmForumUni, atmForumAdmin    FROM  ATM-FORUM-MIB
        OBJECT-TYPE                         FROM RFC-1212;

-- Textual Convention
--
-- Representations of this MIB Module of an ATM address
-- use the data type:

AtmAddress ::= OCTET STRING (SIZE (8 | 20))

-- New MIB Groups
```

[23]See Section 5.2.1.1

```
atmfSrvcRegistryGroup    OBJECT IDENTIFIER ::= { atmForumUni 8 }


-- Object Identifier definitions
--
-- The following values are define dfor use as possible values
-- of the atmfSrvcRegServiceID object.

atmfSrvcRegTypes         OBJECT IDENTIFIER ::= { atmForumAdmin 5 }

-- LAN Emulation Configuration Server (LECS)
atmfSrvcRegLecsOBJECT IDENTIFIER ::= { atmfSrvcRegTypes 1 }


--      The Service Registry Table
--
-- The Service Registry Table is implemented by the network side
-- of the ATM UNI port

atmfSrvcRegTable OBJECT-TYPE
        SYNTAX          SEQUENCE OF AtmfSrvcRegEntry
        ACCESS          not-accessible
        STATUS          mandatory
        DESCRIPTION
                "The table implemented by the UNI Management Entity on the
                network side of the ATM UNI port contains all of the
                services that are available to the user-side of the UNI
                indexed by service identifier."
        ::= { atmfSrvcRegistryGroup 1 }

atmfSrvcRegEntry OBJECT-TYPE
        SYNTAX          AtmfSrvcRegEntry
        ACCESS          not-accessible
        STATUS          mandatory
        DESCRIPTION
                "Information about a single service provider that is
                available to the user-side of the ATM UNI port."
        INDEX  { atmfSrvcRegPort, atmfSrvcRegServiceID,
          atmfSrvcAddressIndex }
        ::= { atmfSrvcRegTable 1 }

AtmfSrvcRegEntry ::=
        SEQUENCE        {
                atmfSrvcRegPort             INTEGER,
                atmfSrvcRegServiceID        OBJECT IDENTIFIER,
                atmfSrvcRegATMAddressAtmAddress,
                atmfSrvcRegAddressIndex             INTEGER
        }

atmfSrvcRegPort OBJECT-TYPE
        SYNTAX          INTEGER (0..2147483647)
        ACCESS          not-accessible
        STATUS          mandatory
        DESCRIPTION
```

```
                              "A unique value which identifies the UNI port for
                              which the service provider is available to the
                              user-side.  The value of 0 has the special meaning
                              of identifying the local UNI."
  ::= { atmfSrvcRegEntry 1 }

atmfSrvcRegServiceID OBJECT-TYPE
        SYNTAX          OBJECT IDENTIFIER
        ACCESS          not-accessible
        STATUS          mandatory
        DESCRIPTION
                "This is the service identifier which uniquely identifies
                the type of service at the address provided in the table."
        ::= { atmfSrvcRegEntry 2 }

atmfSrvcRegATMAddress OBJECT-TYPE
        SYNTAX          AtmAddress
        ACCESS          read-only
        STATUS          mandatory
        DESCRIPTION
                "This is the full address of the service.  The user-side
                ATM UNI port may use this address to establish a connection
                with the service."
        ::= { atmfSrvcRegEntry 3 }

atmfSrvcRegAddressIndex OBJECT-TYPE
        SYNTAX          INTEGER
        ACCESS          not-accessible
        STATUS          mandatory
        DESCRIPTION
                "An arbitrary integer to differentiate multiple rows
       containing different ATM addresses for the same service
       on the same port."
        ::= { atmfSrvcRegEntry 4 }

END
```

## 12.3 State Machine Description of LAN Emulation

This appendix presents a state machine model of the following LAN Emulation elements:

- LAN Emulation Client (LE Client),
- LAN Emulation Configuration Server (LECS),
- LAN Emulation Server (LE Server) and
- Broadcast and Unknown Server (BUS).

The purpose of this appendix is to help ensure the correctness and completeness of the specification and to increase its clarity. This is important to assure interoperability at the LUNI interface between an LE Service from a particular vendor and LE Clients from any other vendor. This model should be viewed as a possible interpretation of the LAN Emulation specification. Note that some of the transitions are implementation dependent, either optional or determined by local policy.

Each state machine is modeled with a set of states and a set of transitions. Each transition interconnects a pair of states. The behavior of a state machines is based on events and actions: if an event happens when the machine is in a particular state, then the action is performed and a state transition takes place (potentially back to the same state). Events are either reception of a frame over a VCC, a command from the higher layer or timer expiration. An event triggers at most one transition. An event that is not specified in the state machine is ignored.

An asterisk in the figures for the State Machine indicates the unlabeled state change is identical to another one marked with the same asterisk.

### 12.3.1 The LAN Emulation Client

Each LE Client is modeled with a main state machine that starts from the LM_LEC_INITIALIZE.request higher layer command and establishes the Operational_LEC state by going through the configuration, joining, and BUS communication phases.

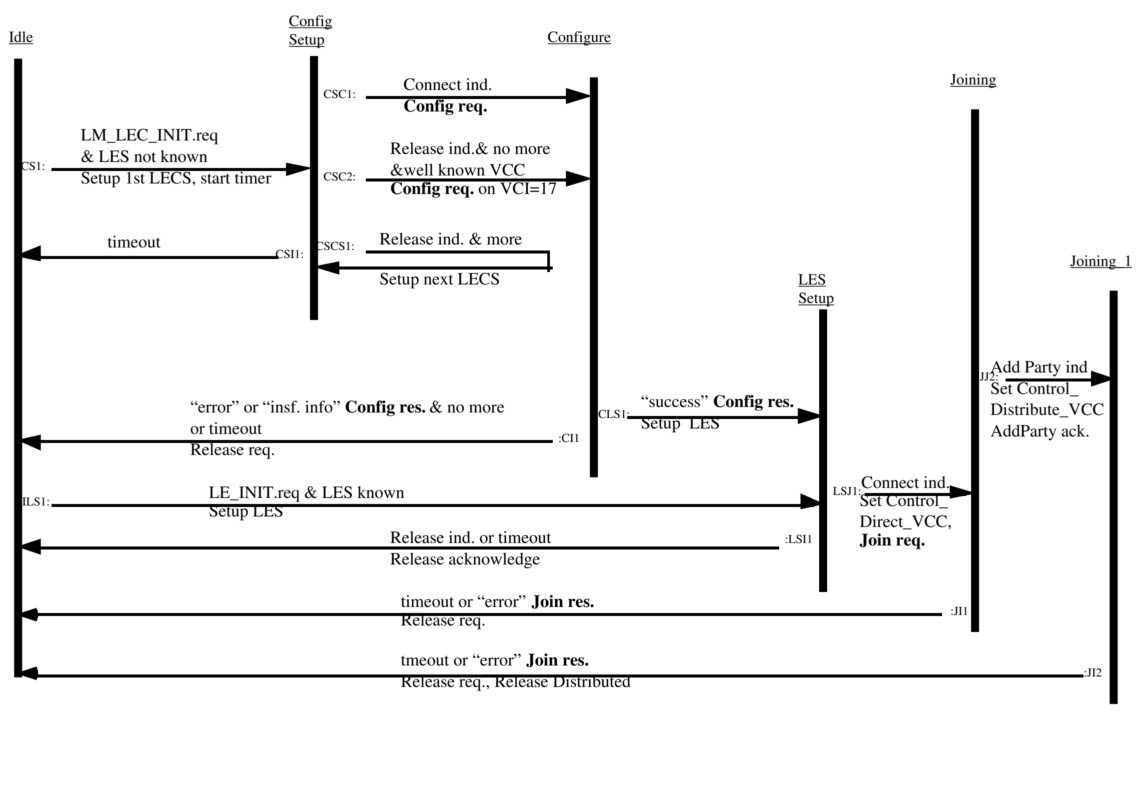
A set of state machines, one per local MAC address, describes the registration process of each address. These machines apply only after the LE Client has completed the joining phase.

An additional set of state machines, one set per LAN Destination (MAC address or route descriptor), describes the establishment of valid LDs while in the Operational_LEC state. Each of these "LD" machines establishes an operational LAN Destination (LD) by performing LE_ARP, VCC setup, etc.

#### 12.3.1.1 The LE Client State Machine

The LE Client state machine is shown in Figure 10 and Figure 11. Each of the transitions is described in more detail below.

**Figure 10. LE Client State Machine (Initialization)**



ICS1: When the higher layer issues an LM_LEC_INITIALIZE.request to the LE Client, the LE Client may or may not know the ATM address of the LE Server. If the LE Client does not know the ATM address of the LE Server, then it tries to setup the Config VCC to an LECS, based on the first ATM address from the list of potential ATM addresses of LECSs. The LE Client obtains this list through interaction with ILMI (the interaction is not specified here). A well known ATM address of an LECS is also part of the list. . It also starts the config/join timer. (see Sec. 3.4.1.1, 5.2.1).

CSCS1: If an attempt to setup the Config VCC to the LECS fails and the LE Client has more ATM addresses of potential LECSs in its list, then the LE Client repeats the attempt to setup the Config VCC, to the next ATM address in the list. (see Sec. 5.2.1.1).

CSC1: If the establishment of the Config VCC is successful, then the LE Client sends a Config request over the newly established Config VCC. This request includes at least the primary ATM address of the LE Client, and optionally more parameters. (see Sec. 5.3.1.1).

CSC2: If the list of the LECSs is exhausted (i.e., the attempt to setup the last connection fails), then the LE Client uses the well known VCC (VCI=17) to send an Config request. (see sec. 5.2.1.3).

CSI1: If the config/join timer expires before establishing a Config VCC, then the state machine returns to the Idle state.

CLS1: Upon receiving a successful Config response, the LE Client tries to set up Control Direct VCC to the LE Server, based on the ATM address from the Config response frame. (see sec 5.3.1.3, 5.3.1.4, 5.4.1.1).

CI1: If the Config response indicates an "error" or "insufficient information", or if the config/join timer expires before receiving a Config response, then the state machine returns to the Idle state and the LE Client releases the Config VCC, . (see sec. 5.3.1.2, 5.3.1.5, 5.3.1.6).

ILS1: If the LE Client knows the ATM address of the LE Server without consulting the LECS, then it tries to setup the Control Direct VCC. It also starts the config/join timer. This causes the state machine to skip to the LES_Setup state.

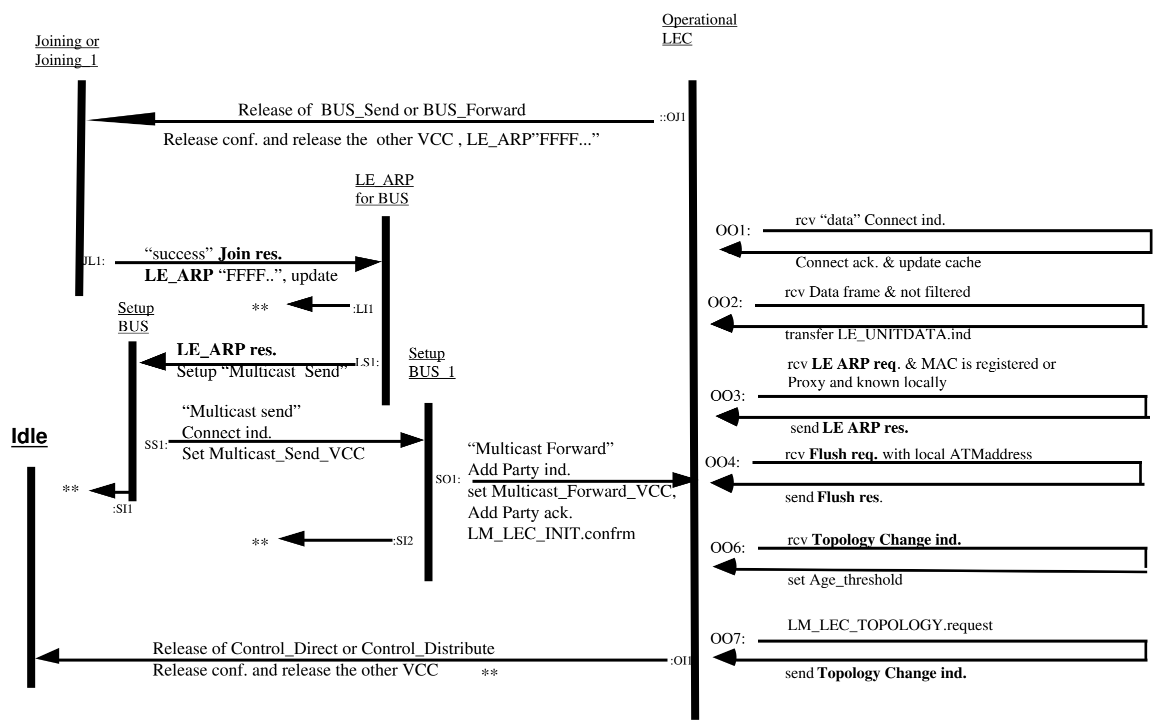
LSJ1: After receiving a Connect indication and completing the establishment of the Control Direct VCC, the LE Client sends a Join request with the proper parameters to the LE Server. (see sec. 5.4.1.2).

LSI1: Back to the Idle state if the attempt to setup the Control Direct VCC fails or the config/join timer expires before the connection is established. (see sec. 5.4.1.9).

JJ2: The LE Client may receive an indication that the LE Server initiated the establishment of a Control Distribute VCC. In that case, the LE Client acknowledges the connection. Note that the first leaf of the point to multipoint Control Distribute VCC is performed with a Setup. Subsequent leafs are established with Add Party. (see sec. 5.4.1.3, 5.4.1.5).

JI1, JI2: Back to the Idle state if the LE Client receives a Join response with "error" or if the config/join timer expires before receiving a successful Join response. (see sec. 5.4.1.8, 5.4.1.10).

**Figure 11. LE Client State Machine (Join, LE_ARP)**




JL1: If the LE Client receives a successful Join response, it issues an LE_ARP for the BUS (an all "1" MAC address) and clears the config/join timer. The LE Client updates it parameters, in particular the LECID, based on the information in the Join response. (see sec 5.4.1.4, 5.4.1.7, 5.6.1.1.1).

LS1: The LE Client receives the LE_ARP response with the ATM address of the BUS. It uses that address to try and set up the Multicast Send VCC. (see sec. 5.6.1.1.2).

SS1: The LE Client receives a Connect indication and completes the establishment of the Multicast Send VCC to the BUS. (see sec. 5.6.1.1.3).

SO1: The LE Client receives an indication from the BUS to establish a Multicast Forward VCC. In case of a point to multipoint VCC, the indication is a Setup for the first leaf and an Add Party for the subsequent ones. The LE Client acknowledges the establishment of the Multicast Forward VCC, enters the Operational_LEC state and issues an LE_LEC_INITIALIZE.indicate to the higher layer. (see sec. 5.6.1.1.3).

LI1, SI1, SI2, OI1: If at any time, the LE Client or LE Server the Control Direct VCC or the Control Distribute VCC, then the LE Client initiates the release of all the rest and goes back to the Idle state. (see sec. 5.6.1.1.4, 8.1.13).

OJ1: If at any time, the LE Client or BUS release the Multicast Send VCC or the Multicast Forward VCC, then the LE Client initiates the release of the other one and goes to the Joining state. (see sec. 5.4.1.11).

The following group of transitions describes the LE Client's behavior while in the Operational_LEC state.

OO1: The LE Client accepts incoming Direct VCCs from other LE Clients if resources are available. The LE Client records the Direct VCCs in its LAN Destination cache. (see sec. 8.1.11).

OO2: The LE Client filters Data frames that arrive on the Direct VCCs and on the Multicast Forward VCC, based on the LECID, Destination MAC address or Source MAC address (subject to local implementation). The LE Client encapsulates the frames that pass filtering in LE_UNITDATA.indications and transfers them to the higher layer. (see sec. 3.1.2, 8.1.4).

OO3: The LE Client responds to each of the LE_ARP requests that include a MAC address that the LE Client had either registered or (if it joined as a Proxy) knows to be reachable. The response includes the ATM address that matches the MAC address. (see sec. 7.1.3, 7.1.4, 7.1.5, 7.1.6).

OO4: The LE Client responds to all the Flush requests that include one of its ATM addresses. The Flush request is received over a data VCC and the Flush response is transmitted over the Control Direct VCC. (see sec. 9.1.1.2, 9.1.1.5, 9.1.1.6).

OO6: The LE Client updates its cache aging threshold based on incoming Topology_Change indications: if the flag is set, then the aging is accelerated, otherwise it is back to its default. (see sec. 7.1.16, 7.1.17, 7.1.18).

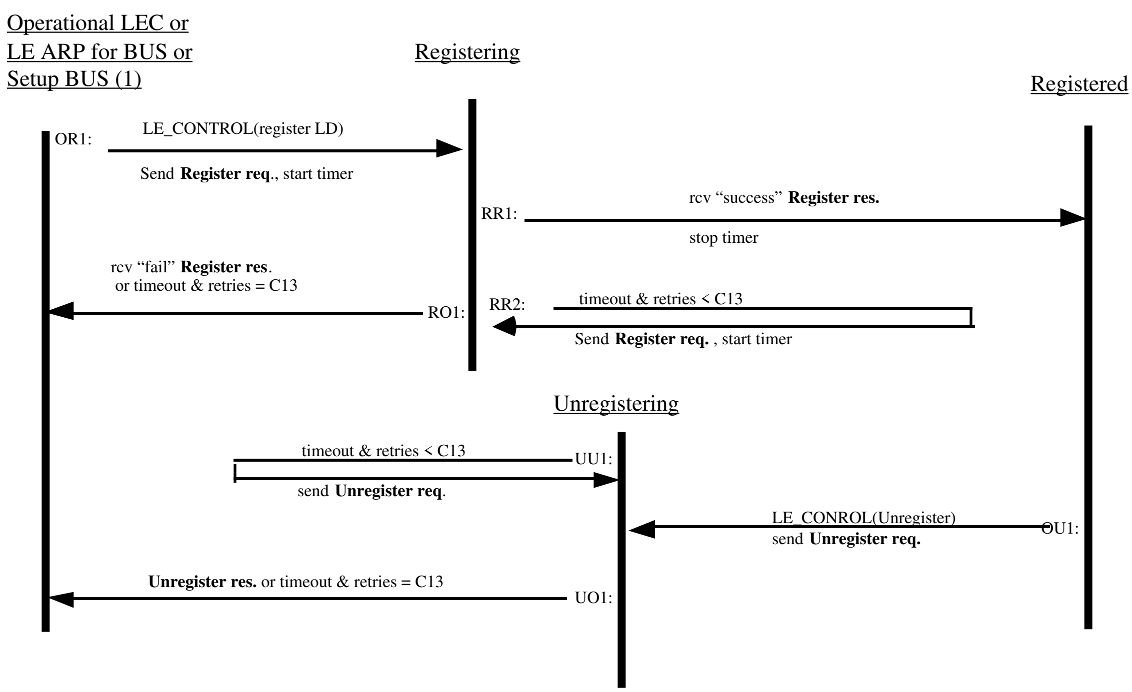
OO7: An LE Client, that is an 802.1-D bridge, issues a Topology Change request following a higher layer LM_LEC_TOPOLOGY.request associated with sending a Topology Change BPDU. (see sec. 3.4.1.7, 7.1.20).

#### 12.3.1.2 Registration State Machines

Registration state machines are invoked by each LM_LEC_REGISTER(LD).request command from the higher layer. This is done only if the LE Client state machine has finished its Joining phase.

The Registration state machine is described in Figure 12. Each of the transitions is described in more details below.

**Figure 12. Registration State Machines**




OR1: The LE Client sends a Register request with the local LD and starts the Register timer upon receiving an LE_CONTROL(register LD).req from the higher layer.

RR1: When the LE Client receives a successful Register response, with the respective LD, it clears the Register timer. At that point, the particular LD is registered.

RR2: If the Register timer expires, and the number of retries is less than the maximum, then the LE Client re-sends the Register request and restarts the Register timer. (see sec. 6.1.1.6).

RO1: If the returning Register response indicates failure or the Register timer expires and the number of retries exceeds the maximum, then the registration of this LD fails. The LE Client notifies the higher layer with an LE_STATUS.indication. (see sec. 6.1.1.8).

OU1: The LE Client sends an Unregister request with the local LD and starts the Register timer upon receiving an LM_LEC_UNREGISTER(LD).request from the higher layer.

UO1: If the returning Unregister response indicates success or the Register timer expires and the number of retries exceeds the maximum, then the unregistration of this LD is complete.

UU1: If the Register timer expires, and the number of retries is less than the maximum, then the LE Client re-sends the Unregister request and restarts the Register timer.

#### 12.3.1.3 The LAN Destination State Machines

A LAN Destination state machine is invoked by each LE_UNITDATA.request from the higher layer. This is done only if the LE Client state machine is in the Operational_LEC state.

The LAN Destination state machine is described in Figure 13 and Figure 14. It includes the following states:

**Figure 13. LAN Destination State Machines (LE_ARP)**

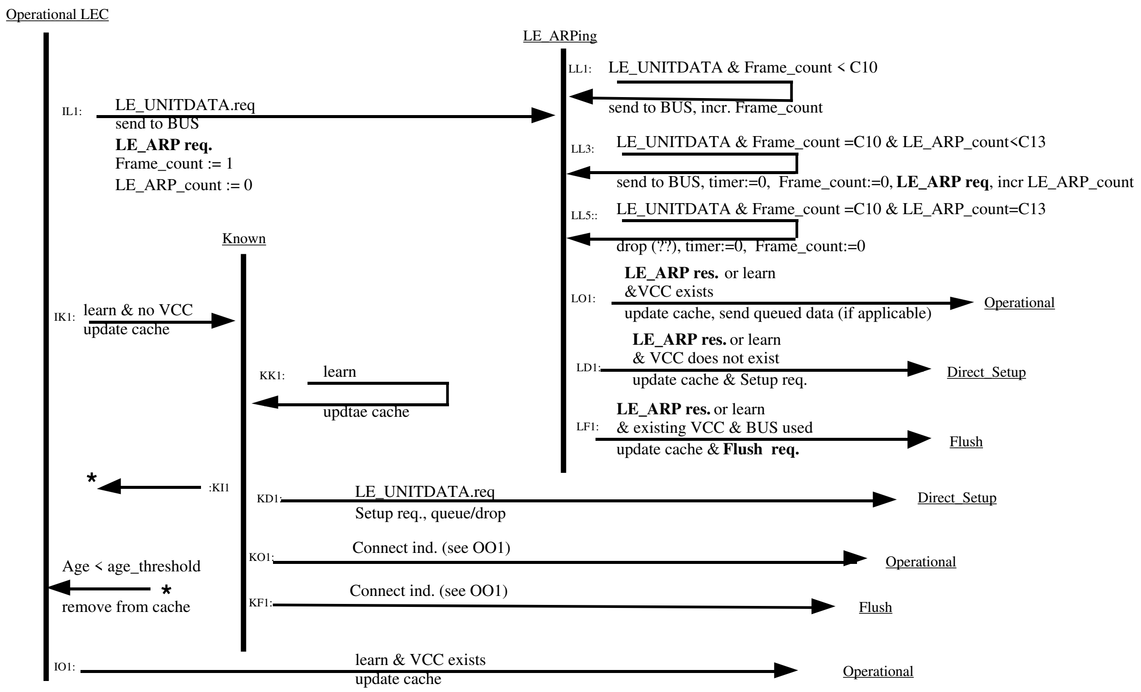
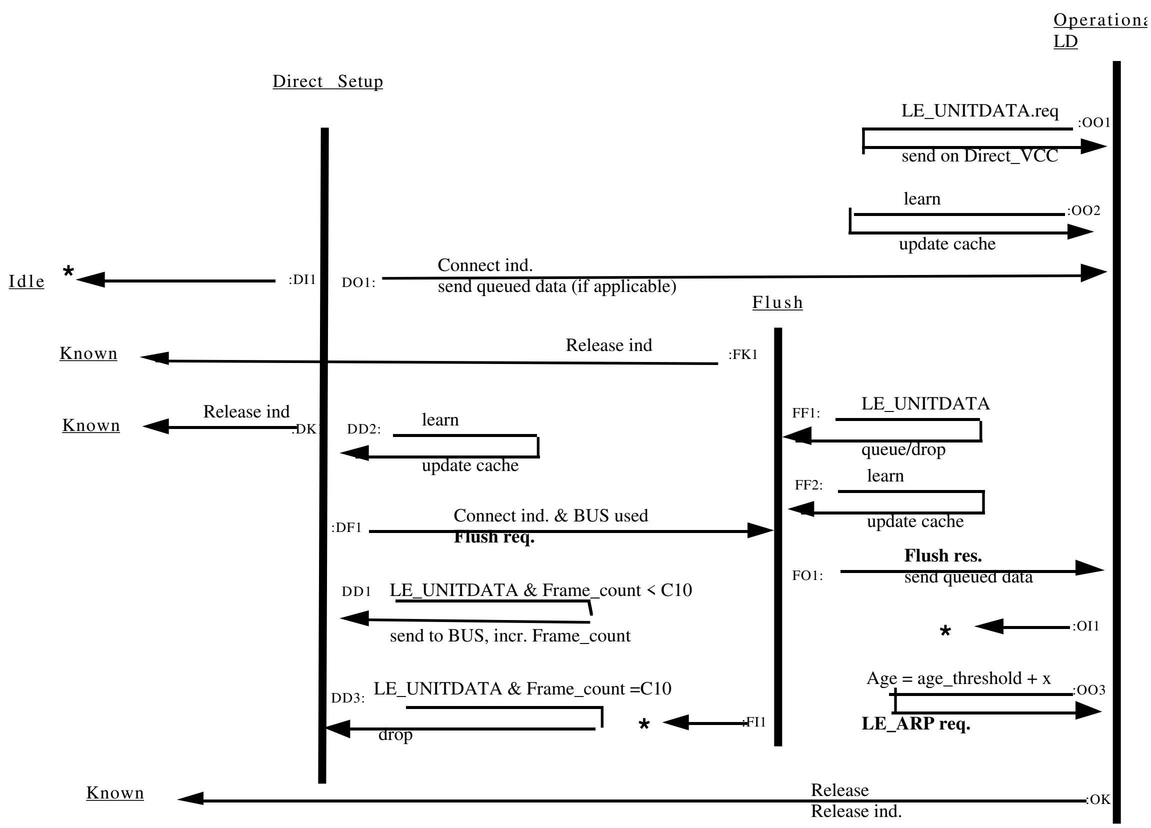



**Figure 14. LAN Destination State Machines (Direct Setup)**



- Operational_LEC: the LE Client does not associate this LD with any ATM address or Direct VCC.
- LE_ARPing: the LE Client has sent an LE_ARP for this LD and is awaiting a response.
- Known: the LE Client associates this LD with an ATM address but not with a Direct VCC.
- Direct_Setup: the LE Client has initiated and is awaiting the completion of the setup of a Direct VCC to the remote LE Client.
- Flush: the LE Client has sent a Flush request and is awaiting a response.
- Operational_LD: the LE Client associates this LD with an ATM address and a Direct VCC to the remote LE Client. In this state it sends LE_UNITDATA.requests from the higher layer over the respective Direct VCC.

The detailed description is presented in three groups: transitions following the reception of LE_UNITDATA.requests, transitions following learning and other transitions.

IL1: When an LE_UNITDATA.request for an LD that is not in the cache arrives from the higher layer, then the LE Client initiates an LE_ARP request and sends the frame to the BUS (or queues/drops it, depending on policy). The decision as to send to the BUS, queue or drop frames whose LD is unknown depends on the local implementation. The LE Client initializes the Frame counter and LE_ARP counter.

LL1, DD1: If the Frame counter is less than the maximum, the LE Client forwards new LE_UNITDATA.requests to the BUS and increments the Frame counter, subject to the restriction of sending no more than C11 frames in a C10 time period to the BUS. The frames may be queued or dropped, based on local policy.

LL3: When the Frame counter reaches its maximum, and the LE_ARP counter does not, then the LE Client forwards the new LE_UNITDATA.requests to the BUS, increments the Frame counter (subject to the restriction of sending no more than C11 frames in a C10 time period), sends an LE_ARP request and increments the LE_ARP counter. The frames may also be queued or dropped, based on local policy.

LL5: When both the Frame counter and the LE_ARP counter reach their maximum, then the LE Client discards the new LE_UNITDATA.request, and initializes the Frame counter and LE_ARP counter.

KD1: When the ATM address is known, the LE Client initiates a connection setup and forwards the new LE_UNITDATA.requests to the BUS. The frames may also be queued or dropped, based on local policy.

DD3: While in the Direct_Setup state, if the Frame counter reaches its maximum, then the LE Client forwards the new LE_UNITDATA.requests to the BUS (subject to the restriction of sending no more than C10 frames in a C11 time period) and increments the Frame counter. The frames may also be queued or dropped, based on local policy.

FF1: While in the Flush state, the LE Client queues or discards all the new LE_UNITDATA.requests from the higher layer.

OO1: While in the Operational_LD state, the LE Client sends all the new LE_UNITDATA.requests over the proper Direct VCC. This transition is the major objective of LAN Emulation!

The following group specifies the behavior during learning:

IK1: When the LE Client learns an ATM address (from an LE_ARP response or from a Data frame) and it does not know the respective Direct VCC, it updates the Cache (initializes the Age timer) and goes to the Known state.

IO1: When the LE Client learns an ATM address (from an LE_ARP response or from a Data frame) and it knows the respective Direct VCC, it updates the Cache (initializes the Age timer) and goes directly to the Operational_LD state.

LD1: While in the LE_ARPing state, if the LE Client receives an LE_ARP response with (or learns in another way) an ATM address LD association and it does not know the Direct VCC, then it initiates the setup of a direct VCC, updates the cache, initializes the Age timer, clears the LE_ARP timer and transitions to the Direct_Setup state.

LF1: While in the LE_ARPing state, if the LE Client receives an LE_ARP response with (or learns in another way) an ATM address LD association and it knows the Direct VCC and (depending on the implementation) it has used the BUS during the last BUS latency time, then it sends a Flush request, updates the cache, initializes the Age timer, clears the LE_ARP timer and transitions to the Flush state.

LO1: While in the LE_ARPing state, if the LE Client receives an LE_ARP response with (or learns in another way) an ATM address LD association and it knows the Direct VCC and it has not used the BUS during the last BUS latency time, then it sends the accumulated Data frames (if applicable), updates the cache, initializes the Age timer, clears the LE_ARP timer and transitions directly to the Operational_LD state.

KK1, DD2, FF2, OO2: When the LE Client (re)learns an ATM address of the LD, it reinitializes the Age timer.

The following group describes other transitions:

KO1: While in the Known state, if the LE Client receives a Setup indication for the respective ATM address (e.g., due to the other LE Client setting up a connection to the first one) and it has not used the BUS during the last BUS latency time window, then it sends the queued Data frames over the Direct VCC and transitions directly to the Operational_LD state.

KF1: While in the Known state, if the LE Client receives a Setup indication for the respective ATM address and it has used the BUS during the last BUS latency time window, then it sends a Flush request for that ATM address over the Multicast Send VCC and transitions to the Flush state.

DO1: While in the Direct_Setup state, if the LE Client receives a Connect indication for the respective ATM address and it has not used the BUS during the last BUS latency time window, then it sends the queued Data frames over the Direct VCC and transitions directly to the Operational_LD state.

DF1: While in the Direct_Setup state, if the LE Client receives a connect indication for the respective ATM address and it has used the BUS during the last BUS latency time window, then it sends a Flush request for that ATM address over the Multicast Send VCC and transitions to the Flush state.

FO1: When the LE Client receives a Flush response that matches its outstanding ATM address, then it send the queued Data frames over the Direct VCC (if applicable) and transitions to the Operational_LD state. (see sec. 9.1.2.2, 9.1.2.5).

OO3: While in the Operational_LD state and the LE Client has sent at least one Data frame over the Direct VCC since the last initialization of the Age timer, the LE Client issues an LE_ARP request a little before the Age timer is about to expire (enough time to allow the LE_ARP response to return). (see sec. 7.1.13, 8.1.12).

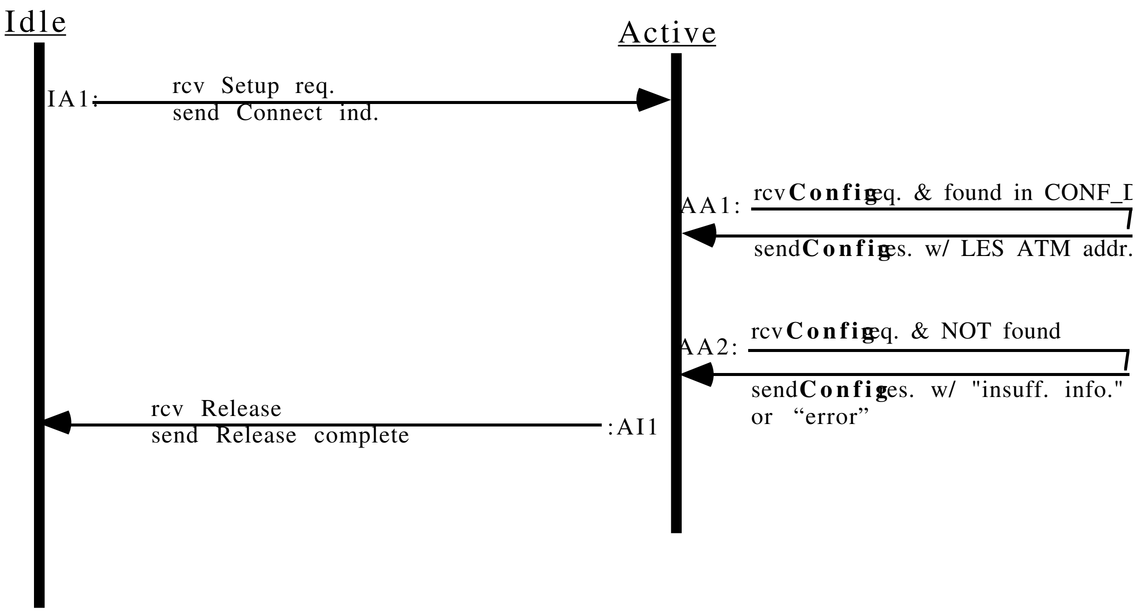
DK1, FK1, OK1: If the LE Client receives a Release indication while in the Direct_Setup, Flush or Operational_LD state, then it returns to the Known state.

DI1, KI1, FI1, OI1: When the Age timer expires, the LE Client removes the LD from the cache and returns to the Idle state. Note that the Age timer may expire due to a change in the Topology Change flag (not just elapsed time).

### 12.3.2 The LE Configuration Server

The LE Configuration Server (LECS) is modeled with a set of state machines, one per Config VCC from a particular LE Client. The state machine is specified in Figure 15. Following is a more detailed description of each transition:

**Figure 15. LE Configuration Server State Machine**




IA1: This is the "wake up" transition for the state machine that configures the new LE client who has initiated a setup request to the LECS. If the LECS exists and decides to accept the call, then it returns a Connect to complete the establishment of the Config VCC. (see sec. 5.2.2)

AA1: The LECS receives a Config request over the Config VCC from an LE client and the Config data base contains an entry indicating the ATM address of an LE Server that will serve this LE client. The LECS returns a Config response with the appropriate configuration information (in particular the ATM address of the LE Server) to the LE client. (see sec. 5.3.2)

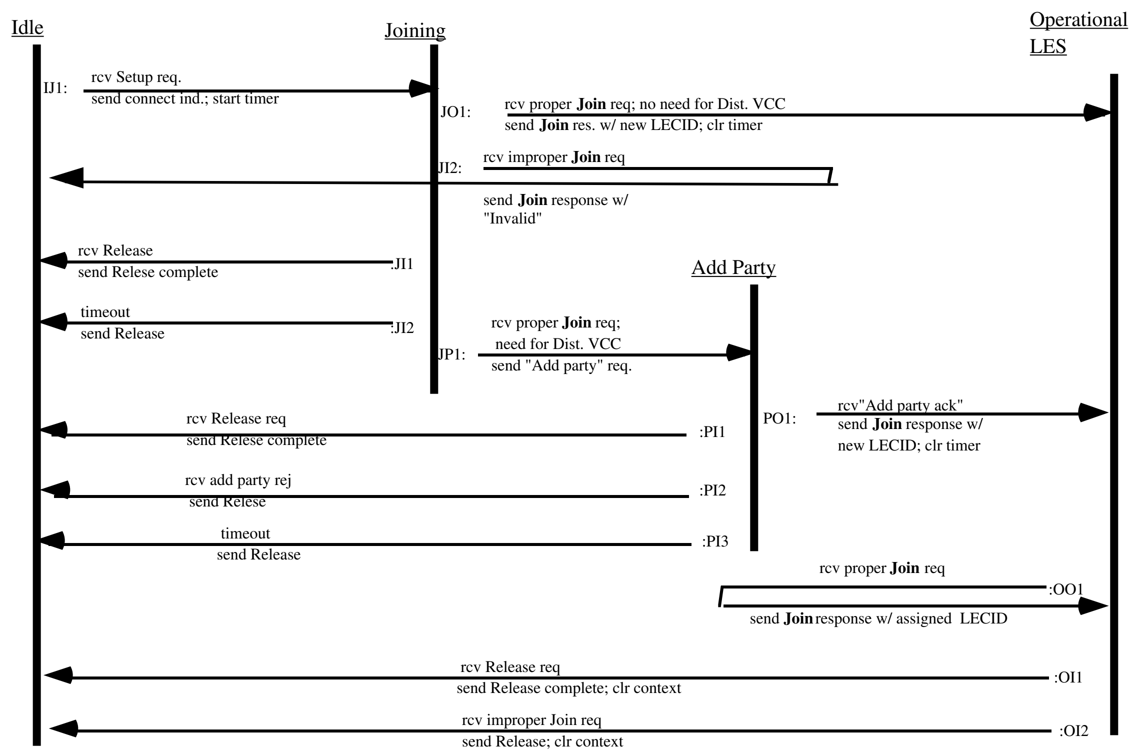
AA2: The LECS receives a Config request from the LE client and the Config data base does not contain an entry indicating the ATM address of an LE Server that will serve this client. The incoming frame may conflict with the Config data base or it may contain insufficient information to derive the ATM address of the LE Server. The LECS returns a Config response indicating "error" or "insufficient information". (see sec. 5.3.2).

AI1: The LECS receives a Release request which is interpreted as indicating that the LE client does not need any more configuration information. The LECS responds with a Release complete to terminate the Config VCC and the state machine returns to the Idle state.

### 12.3.3 The LAN Emulation Server

The LAN Emulation Server (LE Server) is modeled with a set of state machines, one per Control Direct VCC from a particular LE Client. The state machine is specified in Figure 16 and Figure 17. Following is a more detailed description of each transition, first the joining process:

**Figure 16. LAN Emulation Server (Join)**




IJ1: This is the wake up transition for this state machine to provide service to a new LE client. If resources are available, then the LE Server accepts the Setup request by returning a Connect indication to complete the establishment of the Control Direct VCC. The LE Server starts the timer S4, and transitions to the Joining state. (see sec. 5.4.2.1)

JO1: The LE Server receives a Join request with proper values of LAN Type, MAX MTU, and LE client ATM address. If, by local policy, the LE Server decides not to establish a Control Distribute VCC, then it returns a Join response with a unique LECID (and other parameters). The LE Server records LE Client specific information and registers an LD (if one is piggybacked on the Join request). It clears the timer and enters the Operational_LES state. (see sec. 5.4.2.9, 5.4.2.4, 5.4.2.8)

JP1: The LE Server receives a Join request with proper values of LAN Type, MAX MTU, and LE client ATM address. If, by local policy, the LE Server decides to establish a

Control Distribute VCC, then it sends a Setup (for the first leaf) or Add Party request to establish the Control Distribute VCC. The LE Server records LE Client specific information and registers an LD (if one is piggybacked on the Join request).(see sec. 5.4.2.4).

PO1: The LE Server receives a confirmation of the successful addition of the LE client to the Control Distribute VCC. It returns a Join response with a unique LECID (and other parameters). It clears the timer and enters the Operational_LES state. (see sec. 5.4.2.8, 5.4.2.9, 5.4.2.11).

JI2, OI2: While in the Joining or Operational_LES state, if an LE Server receives a Join request with at least one improper value of LAN Type, MAX MTU, or client ATM address, then it returns an "invalid" Join response and returns to the Idle state. (see sec. 5.4.2.4, 5.4.2.5, 5.4.2.6, 5.4.2.8, 5.4.2.14, 5.4.1.15).

JI1, PI1: While in the Joining or Add_Party state, if the LE Server receives a Release request from the LE Client, it responds with a Release complete, it initiates the release of the Control Distribute VCC (if established), clears its data base from the LE Client specific information and from the registered LD (if applicable), and returns to the Idle state. (see sec. 5.4.2.7, 5.4.2.13).

OI1: While in the Operational_LES state, if the LE Server receives a Release request for the Control Direct VCC, the Control Distribute VCC, it responds with a Release complete for that one and initiates the release of the rest. It clears its data base from the LE Client specific information and from the registered LD (if applicable) and returns to the Idle state. (see sec. 5.4.2.13, 8.4.1, 10.2.1).

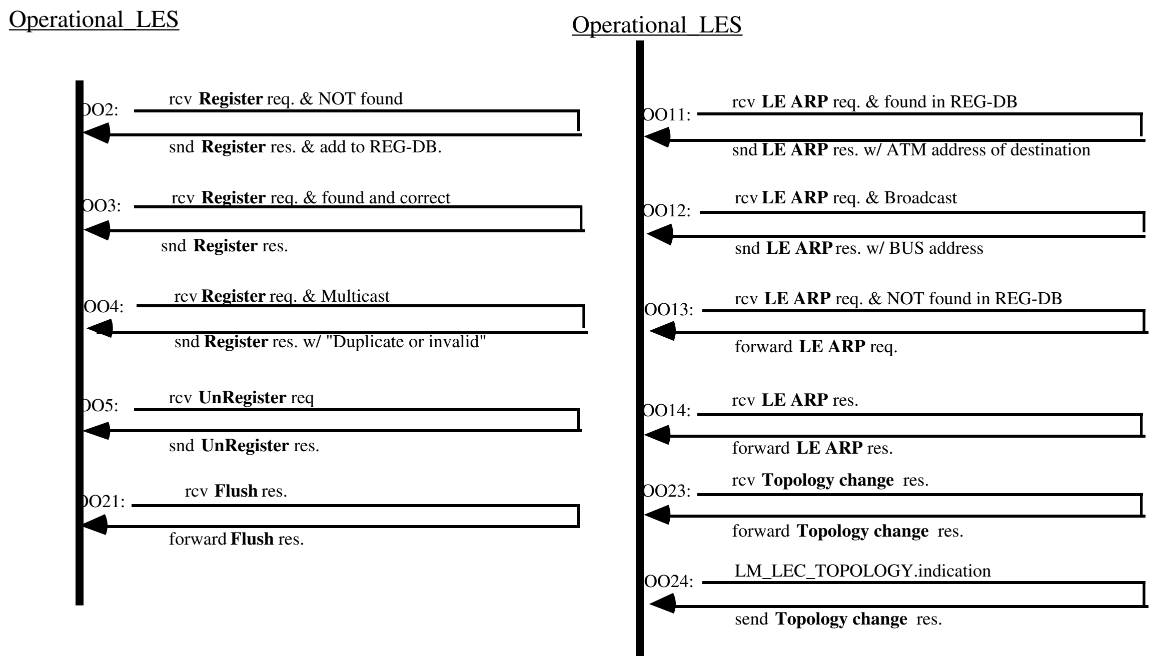
JI3, PI3: If the timer expires then the LE Server interprets that the LE client will not or cannot proceed with the joining process. The LE Server releases the Control Direct VCC and returns to the Idle state. (see sec. 5.4.2.12)

PI2: While in the Add_Party state, if the LE Server receives an Add Party reject or (for the first leaf) a connection release, namely, the establishment of the Control Distribute VCC failed, the LE Server releases the Control Direct VCC and returns to the Idle state. (see sec. 5.4.2.7).

OO1: If the LE Server receives a (duplicate) Join request, probably because the previous responses failed to reach to the client, and the content of the request is identical to the request that caused the state machine to leave the Joining state, then it sends a response on the Control VCC or Control Distribute VCC acknowledging the request. (see sec. 5.4.2.14).

The following group of transitions specifies the transitions while in the Operational_LES state:

**Figure 17. LAN Emulation Server (Registration)**




OO2: When the LE Server receives a Register request for a unicast address that does not exist in the Registration data base, it enters the LD / ATM address association to the data base and returns a "success" Register response to the LE Client.

OO3: When the LE Server receives a Register request for a unicast address for which a correct LD / ATM address exists in the Registration data base, then it returns a "success" Register response to the LE Client.

OO4: When the LE Server receives a Register request for a multicast address or a unicast address that is already registered with a different ATM address, then it returns a "duplicate or invalid" Register response to the LE Client. (see sec. 6.1.2.3, 6.1.2.6).

OO5: When the LE Server receives an Unregister request, then it removes it from the Registration data base (if it exists) and returns a "success" Unregister response to the LE Client. (see sec. 6.1.2.8).

OO11: When the LE Server receives an LE_ARP request and it finds the LD in its registration data base, then it either (based on local policy) returns an LE_ARP response with the associated ATM address or it forwards the LE_ARP request to at least the LE Client that can reach the LD. (see sec. 6.2.7, 7.2.3).

OO12: When the LE Server receives an LE_ARP request with a broadcast address (all "1"s), then it returns an LE_ARP response with the ATM address of the BUS. (see sec. 7.2.9).

OO13: When the LE Server receives an LE_ARP request and it does not find the LD in its registration data base, then it forwards the LE_ARP request to all the LE Clients that have registered as proxies. (see sec. 7.2.4, 7.2.8).

OO14: When the LE Server receives an LE_ARP response, then it forward the frame to the specific LE Client, based e.g., on the Requester-LECID, or (based on local policy) it forwards the frame to all the LE Clients.

OO21: When the LE Server receives a Flush response, then it forward the frame to the specific LE Client, based e.g., on the Requester-LECID, or (based on local policy) it forwards the frame to all the LE Clients.

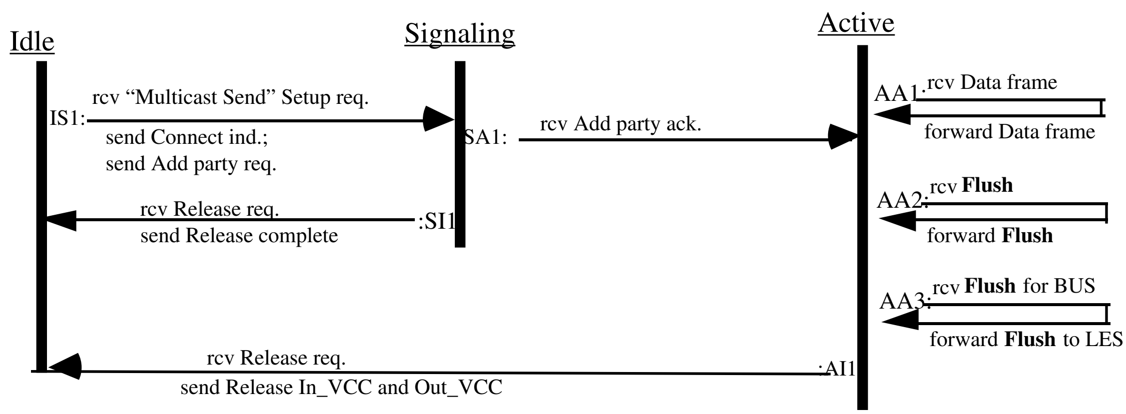
OO23: When the LE Server receives a TOPOLOGY_CHANGE request, then it forward the frame to all the LE Clients.

OO24: When the LE Server receives an LM_LEC_TOPOLOGY.request from the higher, then it send a TOPOLOGY_CHANGE request to all the LE Clients.

### 12.3.4 The Broadcast and Unknown Server

The Broadcast and Unknown Server (BUS) is modeled with a set of state machines, one per potential Multicast Send VCC from a particular LE Client. The state machine is described in Figure 18. Following are more details about each transition:

**Figure 18. Broadcast and Unknown Server**




IS1: This state machine wakes up when a client that is to be served initiates a connection to the BUS. The BUS accepts the Setup request (for the Multicast Send VCC) and initiates the establishment of the Multicast Forward VCC. The first leaf is established with a Setup request and subsequent leafs are setup with Add Party requests, depending on the implementation. (see sec. 5.6.1.2.1).

SA1: When the BUS receives a Connect indication (or Add Party indication) for the Multicast Forward VCC, then it is ready for data forwarding and the state machine enters the Active state. (see sec. 5.6.1.2.2, 5.6.1.2.3)

AA1: While in the Active state, the BUS forwards all the frames that arrive on the Multicast Send VCC to the Multicast Forward VCC, namely, to all the LE Clients. (see sec. 8.3.1, 8.3.2, 8.3.4).

AA2: While in the Active state, the BUS forwards all the Flush requests that arrive on the Multicast Send VCC to the Multicast Forward. (see sec. 8.3.3, 9.1.1.3).

AA3: While in the Active state, the BUS may forward Flush requests with its own ATM address that arrive on the Multicast Send VCC the LE Server.

SI1, AI1: When the BUS receives a Release request for the Multicast Send VCC or the Multicast Forward VCC, it acknowledges the release and initiate the release of the other one. The interpretation is that the client no longer requires service. The state machine returns to the Idle state. (see sec. 5.6.1.2.4, 10.2.1).

## 12.4 Special Topics

### 12.4.1 Maximum Frame Size Calculation

**IEEE 802.3/Ethernet** MAX Frame size is based of the maximum of 1518 octets for the DA (6 octets), SA (6 octets), Type/Length (2 octets), Info (1500 octets), and FCS (4 octets) fields. Since LANE data frames also includes the 2 octet LAN Emulation Header (LEH) but does not include the FCS field, the Emulated Ethernet MAX Frame Size is **1516 octets** (LEH, DA, SA, Type/Length, and Info fields). This results in 32 ATM cells (48 byte payload per cell) or 1536 octets including 12 octets of pad and the 8 octet trailer.

**IEEE 802.5 Token Ring** MAX Frame Size is based on the 9.1 msec token holding timer which allows for 4550 octets at 4 Mbps operation and 18200 octets at 16 Mbps operation. These lengths include the starting delimiter (SD=1 octet), access control (AC=1 octet), frame control (FC=1 octet), DA (6 octets), SA (6 octets), RI (0 to 30 octets), INFO, FCS (4 octets) fields, Ending delimiter (ED=1 octet), frame status (FS=1 octet) field, and the inter-frame gap (IFG). The minimum IFG for 4 Mbps operation is 1 octet and is 5 octets for 16 Mbps operation. Since the LANE data frame includes the 2 octet LEH but does not include the SD, FCS, ED, FS, and IFG, the Emulated Token Ring MAX Frame Size is **4544 octets** for 4 Mbps and **18190 octets** for 16 Mbps. For 4 Mbps operation this results in 95 ATM cells or 4560 octets including 8 octets of pad and the 8 octet trailer. For 16 Mbps operation this results in 380 ATM cells or 18240 octets including 42 octets of pad and the 8 octet trailer.

**RFC 1626 "Default IP MTU for use over AAL5"** MAX Frame Size is based on a maximum MTU of 9180 octets for the user_data plus 8 octets for the LLC/SNAP header. This length does not include the AC (1 octet), FC (1 octet), DA (6 octets), SA (6 octets), and RI (0 to 30 octets) fields. Since the LANE data frame includes the 2 octet LEH the Emulated Token Ring MAX Frame Size is **9234 octets**. This results in 193 ATM cells or 9264 octets including 22 octets of pad and the 8 octet trailer.

The following table shows the basis and how the maximum frame size has been determined.

**Table 29. Maximum Frame Size Calculation**

| AAL-5 SDU Max. octets | AAL-5 PDU Max. octets | Fields: | Basis: | |
|---|---|---|---|---|
| 1516 | 1536 (32 cells) | LEH, DA, SA, Type/Length, INFO | 802.3/Ethernet | 1500 octets Info field |
| 4544 | 4560 (95 cells) | LEH, AC, FC, DA, SA, RI, INFO | 802.5 4Mbps | 9.1 msec THT |
| 9234 | 9264 (193 cells) | LEH, AC, FC, DA, SA, RI, INFO | 802.5 16Mbps | RFC 1626 Default IP MTU for use over ATM AAL5 |
| 18190 | 18240 (380 cells) | LEH, AC, FC, DA, SA, RI, INFO | 802.5 16Mbps | 9.1 msec THT |

Note: IEEE 802.3 and 802.5 INFO field includes the 8 octet LLC/SNAP header

AAL-5 PDU size = SDU size + 8 (PDU trailer) + pad to 48*n octets

### 12.4.2 Token Ring Source Routing Information and Structures

This annex explains how the LE Client decodes the source routing information in a frame.

- How to determine if the RI field is present.

- How to determine the RI_type.
- How to determine if the ELAN is the last hop.
- How to determine the next_RD when the ELAN is not the last hop.

#### 12.4.2.1 Determining if the Routing Information Field is Present.

The frame's source address (SA) field identifies the station that originated the frame. In contrast to the DA field, the Individual/Group (I/G) bit is not encoded in the SA since the SA is constrained to be an individual address and thus the implied value for the I/G is always 0. In its place is the routing information indicator (RII) bit used to indicate the presence or absence of a Routing Information (RI) field in the frame. If the RII bit is 0, then no RI field is present (i.e., an NSR frame). If the RII bit is 1, then the RI field is present. There is no change to the Universal/Local (U/L) bit which still indicates whether the address is universally or locally administered (same as the DA field).

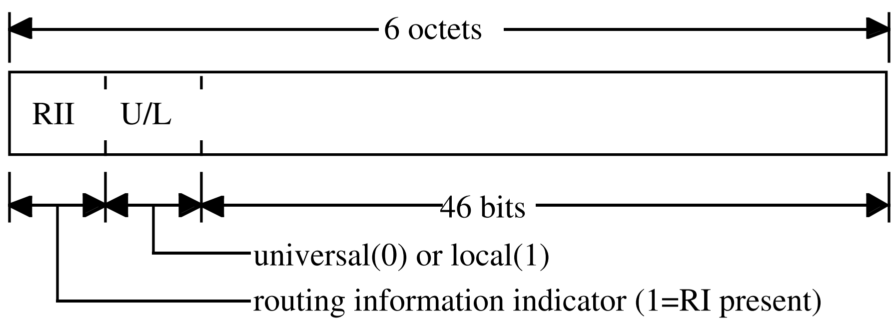
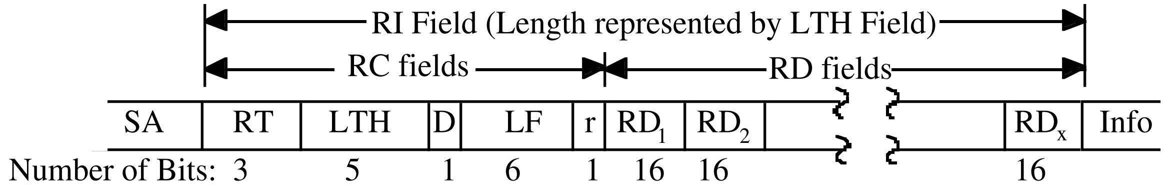



**Figure 19. Source Address Field**

#### 12.4.2.2 ROUTING INFORMATION (RI) FIELD Format.

When a frame's routing information indicator bit in the source address field is equal to 1 (RII=1), the RI field is included in the frame. The RI field immediately follows the SA field as shown in Figure 20. The following provides sufficient information for a LE Client to determine the size of the RI field and parse the frame properly. The detailed structure and contents for the RI field are described in ISO/IEC 10038 (IEEE 802.1D) [4] and ISO/IEC 8802-2 (IEEE 802.2) [9].




**RI** = Routing Information (2 to 30 octets when present)

**RT** = RI_type (3 bits) (SRF, ARE, or STE)

**LTH** = Length of RI field (5 bits) (number of octets)

**D** = Direction (1 bit) indicates ordering of RDs (D=0::$RD_1$ to $RD_X$, D=1::$RD_X$ to $RD_1$)

**LF** = Largest Frame size (6 bits) (encoded)

**r** = reserved (1 bit) (for future use)

**RD** = Route Descriptor (16 bits) (Segment_ID and Bridge)

**Figure 20. Routing Information Field Content**

Figure 21 shows the detailed format for the RI field[24].

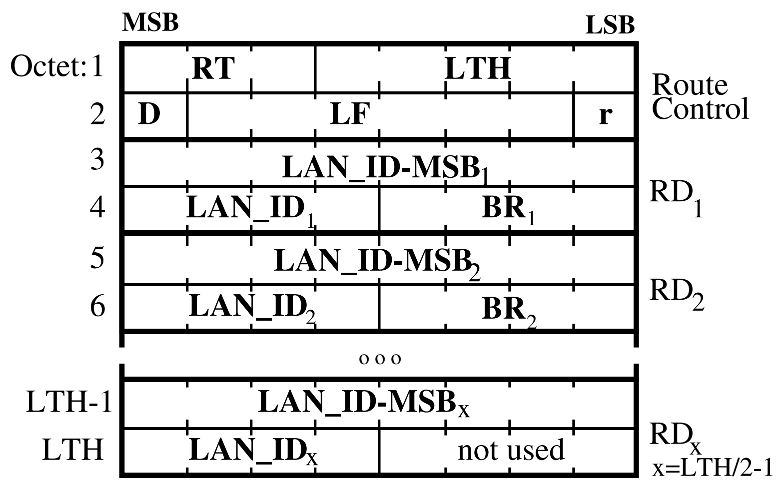



**Figure 21. Routing Information Field Format**

**RI Type (RT).** These three bits indicate the type of the RI field.

RT=000: Specifically Routed Frame (SRF)

RT=10x: All Routes Explorer (ARE)

RT=11x: Spanning Tree Explorer

**LENGTH BITS (LTH).** These five bits indicate the length (in octets) of the RI field. LTH field values are even values between 2 and 30 inclusive.

**Direction (D).** This bit indicates the direction that the frame is traversing the specified LANs. If the D bit is 0, then the frame is traversing the LANs in the order in which the RDs are specified ($RD_1$ to $RD_2$ to ... to $RD_X$). Conversely, if the D bit is set to 1, the

---

[24]See [11].

frame will traverse the LANs in the reverse order ($RD_x$ to $RD_{x-1}$ to ... to $RD_1$). The D bit is only meaningful for SRF frames.

**Largest Frame bits (LF).** These 6 bits are used to indicate the largest frame that may traverse the specified route. These bits are not meaningful to the LE Client entity.

**reserved (r).** This bit is reserved for future use.

**Route Descriptor (RD).** Each route descriptor consists of 16 bits as indicated in Figure 22.

| | MSB | LSB |
|---|---|---|
| | **LAN_ID** | **Bridge_number** |
| Number of Bits: | 12 | 4 |

**Figure 22. Route Descriptor Format**

### 12.4.2.3 Determining the "Next_RD"

The "next_RD" is the term used to describe the Segment_ID and Bridge Number to the next LAN segment. An RI field is built by sending an explorer (ARE or STE) frame through the bridged network. When the RI field is built, each bridge that forwards the frame adds portions of the information until the RI field contains the Segment_ID of the each of the segments and the relative bridge number between any two segments. The information to determine the "next_RD" is relative to the position of the ELAN's Segment_ID and the value of the D bit. The "next_RD" is only applicable for SRF frames.

When the length of a SRF's RI field is less than 6, then there are no hops and the destination is on this ELAN. When the length is 6 or more, the LE Client searches the Segment_IDs in the RI field of the SRF frame until it finds the one that matches the Segment_ID of the emulated LAN. This is $RD_n$ and contains {$Segment\_ID_n$, $BR_n$}. The LEC which is not part of a source routing bridge may assume that $RD_n$ is the first RD in the list ($RD_1$) when the direction bit (D) is 0 or the last RD in the list ($RD_x$) when D=1. If D=0 and $RD_n$ is the last RD in the list, then there is not a "next_RD" and the last hop is this ELAN (i.e., the destination is on the local ELAN segment). Also if D=1 and $RD_n$ is the first RD in the list, then there is not a "next_RD" and the last hop is this ELAN. Otherwise this ELAN is not the last hop and the SRF frame needs to be forwarded to the bridge as specified by the "next_RD".

The RD immediately before $RD_n$ is $RD_{n-1}$ and it contains {$Segment\_ID_{n-1}$, $BR_{n-1}$}. The RD immediately after $RD_n$ is $RD_{n+1}$ and it contains {$Segment\_ID_{n+1}$, $BR_{n+1}$}. When the direction bit (D) is 0 then the "next_RD" is the Segment_ID from the $RD_{n+1}$ and the bridge number from $RD_n$. When the D bit is 1, the "next_RD is composed of the Segment_ID and bridge number from $RD_{n-1}$.

| $RD_{n-1}$ | | $RD_n$ | | $RD_{n+1}$ | |
|---|---|---|---|---|---|
| **SEG_ID** $_{n-1}$ | **BR**$_{n-1}$ | **SEG_ID** $_n$ | **BR**$_n$ | **SEG_ID** $_{n+1}$ | **BR**$_{n+1}$ |

If D=0 then "next_RD" = {$SEG\_ID_{n+1}$, $BR_n$}

If D=1 then "next_RD" = {$SEG\_ID_{n-1}$, $BR_{n-1}$}

**Figure 23. Extracting "next_RD"**

### 12.4.3 PVC Configuration

A goal of this specification is to make the operation of LAN Emulation as independent as possible of the choice of whether to use PVCs, SVCs, or both, to connect the various LAN Emulation components. The specification assumes the use of SVCs, and assumes that the differences between SVCs and PVCs can largely be hidden by means outside the scope of this specification. One means to achieve this independence is by pre-configuring the LAN Emulation components as follows:

- For each PVC an LE Client is configured with:
  - The LE Client's ATM address;
  - The ATM address of the LAN Emulation component at the other end of the PVC;
  - The B-LLI code point associated with the PVC.
- For each Control Direct PVC the LE Server is configured with:
  - The LE Server's ATM address;
  - The LE Client's primary ATM address;
  - The LE Client's LECID;
  - The B-LLI code point associated with the PVC;
  - All {LAN Destination, ATM address} registration pairs for that LE Client.

In both lists, the ATM addresses and B-LLI code points are not exchanged by UNI signaling; they are pre-configured values representing the values that presumably would be exchanged by SVC signaling, if the configured PVC were an SVC. In effect, pre-configuration is being substituted for actual UNI signaling. (See also Section 3.1.1).

If the LE Server is restarted with empty registration tables, it cannot release Control PVCs and cause the LE Clients to re-initialize. An LE Server may be restarted in the interval between two different clients' joins or address registrations. Preconfiguring the LECID and address registration information allows the LE Server to meet its obligations to those two LE Clients to ensure the uniqueness of the LECIDs and address registrations. Other methods are possible, e.g. methods involving non-volatile RAM. Pre-configuration is presented as one of the methods which enable LAN Emulation components to work with PVCs.

The absence of UNI signaling prevents conveying the setup or release of a PVC connection from one LAN Emulation component to another. Except for this absence, configuring PVC information in the LE Client and LE Server does not preclude the normal operation of the LAN Emulation protocols. Operations such as Configuration, join phase, address resolution, and address registration can take place normally and in accordance with the relevant sections of this specification.

Use of non-fully-meshed PVCs in a PVC-only environment limits rate of traffic between two LE Clients for which no Data Direct VCC exists to connect them. (See 7.1.8).

End of Document