

Multiprotocol Label Switching (MPLS) Label-Controlled  
Asynchronous Transfer Mode (ATM) and Frame-Relay  
Management Interface Definition

Status of This Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

Copyright Notice

Copyright (C) The Internet Society (2006).

Abstract

This memo defines two MIB modules and corresponding MIB Object Definitions that describe how label-switching-controlled Frame-Relay and Asynchronous Transfer Mode (ATM) interfaces can be managed given the interface stacking as defined in the MPLS-LSR-STD-MIB and MPLS-TE-STD-MIB.

Table of Contents

1. Introduction .....	2
2. Terminology .....	2
3. The SNMP Management Framework .....	3
4. Interface Stacking of LC-ATM .....	3
5. Structure of the MPLS-LC-ATM-STD-MIB Module .....	3
6. Structure of the MPLS-LC-FR-STD-MIB Module .....	4
7. MPLS Label-Controlled ATM MIB Definitions .....	5
8. MPLS Label-Controlled Frame Relay MIB Definitions .....	12
9. Acknowledgments .....	18
10. Security Considerations .....	18
11. IANA Considerations .....	19
11.1. IANA Considerations for MPLS-LC-ATM-STD-MIB .....	19
11.2. IANA Considerations for MPLS-LC-FR-STD-MIB .....	19
12. References .....	20
12.1. Normative References .....	20
12.2. Informative References .....	21

## 1. Introduction

This memo defines how label-switching-controlled Frame-Relay [RFC3034] and ATM [RFC3035] interfaces can be realized given the interface stacking as defined in the MPLS-LSR-STD [RFC3813] and MPLS-TE-STD [RFC3812] MIBs. This document also contains a MIB module that sparsely extends the MPLS-LSR-STD MIB's `mplsInterfaceConfTable` in such a way as to identify which MPLS-type interfaces have LC-ATM or LC-FR capabilities. Comments should be made directly to the MPLS mailing list at `mpls@uu.net`.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119, reference [RFC2119].

## 2. Terminology

This document uses terminology from the document describing the MPLS architecture [RFC3031], as well as from RFC 3035 and RFC 3034. Specifically, the following terms will be used in this document.

**C-FR** RFC 3034 defines a label-switching-controlled Frame Relay (LC-FR) interface. Packets traversing such an interface carry labels in the DLCI field

**C-ATM** RFC 3035 defines a label-switching-controlled ATM (LC-ATM) interface as an ATM interface controlled by the label switching control component. When a packet traversing such an interface is received, it is treated as a labeled packet. The packet's top label is inferred from either the contents of the Virtual Channel Identifier (VCI) field or the combined contents of the Virtual Path Identifier (VPI) and VCI fields. Any two LDP peers that are connected via an LC-ATM interface will use LDP negotiations to determine which of these cases is applicable to that interface. Static configuration of labels is also possible.

When LDP is used to distribute labels for use on label-controlled interfaces, label configuration information may be available in the MPLS-LDP-ATM-STD-MIB [RFC3815] when LC-ATM interfaces are used, or the MPLS-LDP-FRAME-RELAY-STD-MIB [RFC3815] when LC-FR interfaces are used.

### 3. The SNMP Management Framework

For a detailed overview of the documents that describe the current Internet-Standard Management Framework, please refer to section 7 of RFC 3410 [RFC3410].

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. MIB objects are generally accessed through the Simple Network Management Protocol (SNMP). Objects in the MIB are defined using the mechanisms defined in the Structure of Management Information (SMI). This memo specifies a MIB module that is compliant to the SMIV2, which is described in STD 58, RFC 2578 [RFC2578], STD 58, RFC 2579 [RFC2579] and STD 58, RFC 2580 [RFC2580].

### 4. Interface Stacking of LC-ATM

Since LC-ATM interfaces [RFC2863] can carry labeled MPLS traffic, they too are considered MPLS subinterfaces with ifType = mpls(166). They differ slightly in their capability from a packet-oriented MPLS interface in that they may carry ATM- or Frame-Relay-encapsulated traffic. It is thus beneficial to identify them as such. To do this, two tables are defined that extend the MPLS-LSR-STD MIB's mplsInterfaceTable (see section 5 for LC-ATM or section 6 for LC-FR).

### 5. Structure of the MPLS-LC-ATM-STD-MIB Module

The MPLS-LC-ATM-STD-MIB module is structured simply as a table of entries that sparsely extend those found in the interfaces table. In particular, the entries in the mplsLcAtmStdInterfaceConfTable extend interfaces capable of supporting MPLS, as is defined in [RFC3813], to include entries that also support LC-ATM (and their unique attributes). Therefore, the module can be visualized as follows. Note that the ifTable comes from [RFC2863], the mplsInterfaceTable from [RFC3813], and the mplsLcAtmStdInterfaceConfTable from the MPLS-LC-ATM-STD-MIB module described below.

```
ifTable mplsInterfaceTable mplsLcAtmStdInterfaceConfTable
.1
.2      .2
.3
.4      .4      .4
.5
```

In the example shown above, five interfaces exist on the device in question. Of those interfaces, those with ifIndex = .2 and .4 are of ifType = mpls(166) indicating that they are capable of MPLS. Of those two, the entry with index .4 is capable of MPLS LC-ATM operations.

Note that the label partition model utilized by the authors of this document reflects widespread implementation and is seen by the MPLS working group as sufficiently flexible to meet the operational needs, even if it is more restrictive than [RFC3035] allows. To this end, we have limited the control and unlabeled VPI and VCI to single values. Note that mplsLcAtmStdUnlabTrafVci and mplsLcAtmStdCtrlVci MUST not be equal; nor should mplsLcAtmStdCtrlVpi or mplsLcAtmStdUnlabTrafVpi be equal.

## 6. Structure of the MPLS-LC-FR-STD-MIB Module

The MPLS-LC-FR-STD-MIB module is structured simply as a table of entries that sparsely extend those found in the interfaces table. In particular, the entries in the mplsLcFrStdInterfaceConfTable extend interfaces capable of supporting MPLS, as is defined in [RFC3813], to include entries that also support LC-Frame Relay (and their unique attributes). Therefore, the module can be visualized as follows. Note that the ifTable comes from [RFC2863], the mplsInterfaceTable from [RFC3813], and the mplsLcAtmStdInterfaceConfTable from the MPLS-LC-FR-STD-MIB module described below.

```
ifTable mplsInterfaceTable mplsLcFrStdInterfaceConfTable
.1
.2      .2
.3
.4      .4      .4
.5
```

In the example shown above, five interfaces exist on the device in question. Of those interfaces, those with ifIndex = .2 and .4 are of ifType = mpls(166) indicating that they are capable of MPLS. Of those two, the entry with index .4 is capable of MPLS LC-Frame Relay operations.

Note that even though the architecture as described in [RFC3034] calls for supporting mixed labeled and unlabeled traffic, this MIB does not support that, as this capability does not seem to be used operationally. Note that the DLCI ranges represented by mplsLcFrStdTrafficMinDlci to mplsLcFrStdTrafficMaxDlci and mplsLcFrStdCtrlMinDlci to mplsLcFrStdCtrlMaxDlci MUST not overlap.

## 7. MPLS Label-Controlled ATM MIB Definitions

The following MIB module imports from [RFC2514], [RFC3811], and [RFC3813].

```

MPLS-LC-ATM-STD-MIB DEFINITIONS ::= BEGIN
IMPORTS
    MODULE-IDENTITY, OBJECT-TYPE
        FROM SNMPv2-SMI
    MODULE-COMPLIANCE, OBJECT-GROUP
        FROM SNMPv2-CONF
    RowStatus, StorageType, TruthValue
        FROM SNMPv2-TC
    AtmVpIdentifier
        FROM ATM-TC-MIB
    mplsStdMIB, MplsAtmVcIdentifier
        FROM MPLS-TC-STD-MIB
    mplsInterfaceIndex
        FROM MPLS-LSR-STD-MIB
    ;

mplsLcAtmStdMIB MODULE-IDENTITY
    LAST-UPDATED "200601120000Z" -- 12 January 2006
    ORGANIZATION "Multiprotocol Label Switching (MPLS) Working Group"
    CONTACT-INFO
        "
            Thomas D. Nadeau
            Postal: Cisco Systems, Inc.
                    250 Apollo Drive
                    Chelmsford, MA 01824
            Tel:      +1-978-244-3051
            Email:    tnadeau@cisco.com

            Subrahmanya Hegde
            Postal: Cisco Systems, Inc.
                    225 East Tazman Drive
            Tel:      +1-408-525-6562
            Email:    subrah@cisco.com
            General comments should be sent to mpls@uu.net
        "
    DESCRIPTION
        "This MIB module contains managed object definitions for
        MPLS Label-Controlled ATM interfaces as defined in
        [RFC3035].

        Copyright (C) The Internet Society (2006).  This
        version of this MIB module is part of RFC 4368; see
        the RFC itself for full legal notices."

```

```

-- Revision history.
REVISION
    "200601120000Z" -- 12 January 2006
DESCRIPTION
    "Initial revision, published as part of RFC 4368."
 ::= { mplsStdMIB 9 }

-- Top level components of this MIB module.

-- Tables, Scalars, Notifications, Conformance

mplsLcAtmStdNotifications OBJECT IDENTIFIER ::= { mplsLcAtmStdMIB 0 }

mplsLcAtmStdObjects          OBJECT IDENTIFIER ::= { mplsLcAtmStdMIB 1 }

mplsLcAtmStdConformance     OBJECT IDENTIFIER ::= { mplsLcAtmStdMIB 2 }

-- MPLS LC-ATM Interface Configuration Table.
mplsLcAtmStdInterfaceConfTable OBJECT-TYPE
    SYNTAX          SEQUENCE OF MplsLcAtmStdInterfaceConfEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "This table specifies per-interface MPLS LC-ATM
        capability and associated information. In particular,
        this table sparsely extends the MPLS-LSR-STD-MIB's
        mplsInterfaceConfTable."
    ::= { mplsLcAtmStdObjects 1 }

mplsLcAtmStdInterfaceConfEntry OBJECT-TYPE
    SYNTAX          MplsLcAtmStdInterfaceConfEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "An entry in this table is created by an LSR for
        every interface capable of supporting MPLS LC-ATM.
        Each entry in this table will exist only if a
        corresponding entry in ifTable and mplsInterfaceConfTable
        exists. If the associated entries in ifTable and
        mplsInterfaceConfTable are deleted, the corresponding
        entry in this table must also be deleted shortly
        thereafter."
    INDEX          { mplsInterfaceIndex }
    ::= { mplsLcAtmStdInterfaceConfTable 1 }

MplsLcAtmStdInterfaceConfEntry ::= SEQUENCE {
    mplsLcAtmStdCtrlVpi          AtmVpIdentifier,
    mplsLcAtmStdCtrlVci          MplsAtmVcIdentifier,

```

mplsLcAtmStdUnlabTrafVpi	AtmVpIdentifier,
mplsLcAtmStdUnlabTrafVci	MplsAtmVcIdentifier,
mplsLcAtmStdVcMerge	TruthValue,
mplsLcAtmVcDirectlyConnected	TruthValue,
mplsLcAtmLcAtmVPI	AtmVpIdentifier,
mplsLcAtmStdIfConfRowStatus	RowStatus,
mplsLcAtmStdIfConfStorageType	StorageType

}

mplsLcAtmStdCtrlVpi OBJECT-TYPE

SYNTAX AtmVpIdentifier

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"This is the VPI value over which this LSR is willing to accept control traffic on this interface."

::= { mplsLcAtmStdInterfaceConfEntry 1 }

mplsLcAtmStdCtrlVci OBJECT-TYPE

SYNTAX MplsAtmVcIdentifier

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"This is the VCI value over which this LSR is willing to accept control traffic on this interface."

::= { mplsLcAtmStdInterfaceConfEntry 2 }

mplsLcAtmStdUnlabTrafVpi OBJECT-TYPE

SYNTAX AtmVpIdentifier

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"This is the VPI value over which this LSR is willing to accept unlabeled traffic on this interface."

::= { mplsLcAtmStdInterfaceConfEntry 3 }

mplsLcAtmStdUnlabTrafVci OBJECT-TYPE

SYNTAX MplsAtmVcIdentifier

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"This is the VCI value over which this LSR is willing to accept unlabeled traffic on this interface."

::= { mplsLcAtmStdInterfaceConfEntry 4 }

**mplsLcAtmStdVcMerge OBJECT-TYPE**

SYNTAX TruthValue

MAX-ACCESS read-create

STATUS current

## DESCRIPTION

"If set to true(1), indicates that this interface is capable of ATM VC merge; otherwise, it MUST be set to false(2)."

DEFVAL { false }

::= { mplsLcAtmStdInterfaceConfEntry 5 }

**mplsLcAtmVcDirectlyConnected OBJECT-TYPE**

SYNTAX TruthValue

MAX-ACCESS read-create

STATUS current

## DESCRIPTION

"This value indicates whether an LC-ATM is directly or indirectly (by means of a VP) connected. If set to true(1), indicates that this interface is directly connected LC-ATM; otherwise, it MUST be set to false(2). Note that although it can be intimated from RFC 3057 that multiple VPs may be used, in practice only a single one is used, and therefore the authors of this MIB module have chosen to model it as such."

DEFVAL { true }

::= { mplsLcAtmStdInterfaceConfEntry 6 }

**mplsLcAtmLcAtmVPI OBJECT-TYPE**

SYNTAX AtmVpIdentifier

MAX-ACCESS read-create

STATUS current

## DESCRIPTION

"This is the VPI value used for indirectly connected LC-ATM interfaces. For these interfaces, the VPI field is not available to MPLS, and the label MUST be encoded entirely within the VCI field (see [RFC3035]). If the interface is directly connected, this value MUST be set to zero."

DEFVAL { 0 }

::= { mplsLcAtmStdInterfaceConfEntry 7 }

**mplsLcAtmStdIfConfRowStatus OBJECT-TYPE**

SYNTAX RowStatus

MAX-ACCESS read-create

STATUS current

## DESCRIPTION



```

    "This object is used to create and
    delete entries in this table.  When configuring
    entries in this table, the corresponding
    ifEntry and mplsInterfaceConfEntry
    MUST exist beforehand.  If a manager attempts to
    create an entry for a corresponding
    mplsInterfaceConfEntry that does not support LC-ATM,
    the agent MUST return an inconsistentValue error.
    If this table is implemented read-only, then the
    agent must set this object to active(1) when this
    row is made active.  If this table is implemented
    writable, then an agent MUST not allow modification
    to its objects once this value is set to active(1),
    except to mplsLcAtmStdIfConfRowStatus and
    mplsLcAtmStdIfConfStorageType."
 ::= { mplsLcAtmStdInterfaceConfEntry 8 }

mplsLcAtmStdIfConfStorageType OBJECT-TYPE
    SYNTAX          StorageType
    MAX-ACCESS      read-create
    STATUS          current
    DESCRIPTION
        "The storage type for this conceptual row.
        Conceptual rows having the value 'permanent(4)'
        need not allow write-access to any columnar
        objects in the row."
    DEFVAL { nonVolatile }
    ::= { mplsLcAtmStdInterfaceConfEntry 9 }

-- End of mplsLcAtmStdInterfaceConfTable

-- Module compliance.

mplsLcAtmStdCompliances
    OBJECT IDENTIFIER ::= { mplsLcAtmStdConformance 1 }

mplsLcAtmStdGroups
    OBJECT IDENTIFIER ::= { mplsLcAtmStdConformance 2 }

-- Compliance requirement for full compliance

mplsLcAtmStdModuleFullCompliance MODULE-COMPLIANCE
    STATUS current
    DESCRIPTION
        "Compliance statement for agents that provide
        full support for MPLS-LC-ATM-STD-MIB.  Such
        devices can be monitored and also be configured
        using this MIB module."

```

```

MODULE -- this module
    MANDATORY-GROUPS {
        mplsLcAtmStdIfGroup
    }

    OBJECT            mplsLcAtmStdIfConfRowStatus
    SYNTAX            RowStatus { active(1), notInService(2) }
    WRITE-SYNTAX RowStatus { active(1), notInService(2),
                            createAndGo(4), destroy(6)
                        }
    DESCRIPTION "Support for createAndWait and notReady is
                not required."

 ::= { mplsLcAtmStdCompliances 1 }

-- Compliance requirement for read-only implementations.

mplsLcAtmStdModuleReadOnlyCompliance MODULE-COMPLIANCE
    STATUS current
    DESCRIPTION
        "Compliance requirement for implementations that only
        provide read-only support for MPLS-LC-ATM-STD-MIB.
        Such devices can be monitored but cannot be configured
        using this MIB module.
        "
MODULE -- this module
    MANDATORY-GROUPS {
        mplsLcAtmStdIfGroup
    }

    -- mplsLcAtmStdInterfaceConfTable

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

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

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

    OBJECT            mplsLcAtmStdUnlabTrafVci

```

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

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

OBJECT mplsLcAtmStdIfConfRowStatus  
 SYNTAX RowStatus { active(1) }  
 MIN-ACCESS read-only  
 DESCRIPTION "Write access is not required."

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

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

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

::= { mplsLcAtmStdCompliances 2 }

-- Units of conformance.

mplsLcAtmStdIfGroup OBJECT-GROUP  
 OBJECTS {  
   mplsLcAtmStdCtrlVpi,  
   mplsLcAtmStdCtrlVci,  
   mplsLcAtmStdUnlabTrafVpi,  
   mplsLcAtmStdUnlabTrafVci,  
   mplsLcAtmStdVcMerge,  
   mplsLcAtmVcDirectlyConnected,  
   mplsLcAtmLcAtmVPI,  
   mplsLcAtmStdIfConfRowStatus,  
   mplsLcAtmStdIfConfStorageType  
 }  
 STATUS current  
 DESCRIPTION  
   "Collection of objects needed for MPLS LC-ATM"

```
        interface configuration."
 ::= { mplsLcAtmStdGroups 1 }
```

END

## 8. MPLS Label-Controlled Frame Relay MIB Definitions

The following MIB module imports from [RFC2115], [RFC3811], and [RFC3813].

```
MPLS-LC-FR-STD-MIB DEFINITIONS ::= BEGIN
```

```
IMPORTS
```

```
    MODULE-IDENTITY, OBJECT-TYPE
        FROM SNMPv2-SMI
    MODULE-COMPLIANCE, OBJECT-GROUP
        FROM SNMPv2-CONF
    RowStatus, StorageType
        FROM SNMPv2-TC
    mplsInterfaceIndex
        FROM MPLS-LSR-STD-MIB
    DLCI
        FROM FRAME-RELAY-DTE-MIB
    mplsStdMIB
        FROM MPLS-TC-STD-MIB
    ;
```

```
mplsLcFrStdMIB MODULE-IDENTITY
```

```
    LAST-UPDATED "200601120000Z" -- 12 January 2006
    ORGANIZATION "Multiprotocol Label Switching (MPLS) Working Group"
    CONTACT-INFO
        "
            Thomas D. Nadeau
            Cisco Systems, Inc.
            Email: tnadeau@cisco.com

            Subrahmanya Hegde
            Email: subrah@cisco.com

            General comments should be sent to mpls@uu.net
        "
```

```
DESCRIPTION
```

```
"This MIB module contains managed object definitions for
MPLS Label-Controlled Frame-Relay interfaces as defined
in (RFC3034).
```

```
Copyright (C) The Internet Society (2006). This
version of this MIB module is part of RFC 4368; see
the RFC itself for full legal notices."
```

```

-- Revision history.
REVISION
    "200601120000Z" -- 12 January 2006
DESCRIPTION
    "Initial revision, published as part of RFC 4368."
 ::= { mplsStdMIB 10 }

-- Top level components of this MIB module.
-- Tables, Scalars, Notifications, Conformance

mplsLcFrStdNotifications OBJECT IDENTIFIER ::= { mplsLcFrStdMIB 0 }
mplsLcFrStdObjects       OBJECT IDENTIFIER ::= { mplsLcFrStdMIB 1 }
mplsLcFrStdConformance   OBJECT IDENTIFIER ::= { mplsLcFrStdMIB 2 }

-- MPLS LC-FR Interface Configuration Table.
mplsLcFrStdInterfaceConfTable OBJECT-TYPE
    SYNTAX          SEQUENCE OF MplsLcFrStdInterfaceConfEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "This table specifies per-interface MPLS LC-FR
        capability and associated information. In particular,
        this table sparsely extends the MPLS-LSR-STD-MIB's
        mplsInterfaceConfTable."
    ::= { mplsLcFrStdObjects 1 }

mplsLcFrStdInterfaceConfEntry OBJECT-TYPE
    SYNTAX          MplsLcFrStdInterfaceConfEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "An entry in this table is created by an LSR for
        every interface capable of supporting MPLS LC-FR.
        Each entry in this table will exist only if a
        corresponding entry in ifTable and mplsInterfaceConfTable
        exists. If the associated entries in ifTable and
        mplsInterfaceConfTable are deleted, the corresponding
        entry in this table must also be deleted shortly
        thereafter."
    INDEX          { mplsInterfaceIndex }
    ::= { mplsLcFrStdInterfaceConfTable 1 }

MplsLcFrStdInterfaceConfEntry ::= SEQUENCE {
    mplsLcFrStdTrafficMinDlci          DLCI,
    mplsLcFrStdTrafficMaxDlci          DLCI,
    mplsLcFrStdCtrlMinDlci             DLCI,
    mplsLcFrStdCtrlMaxDlci             DLCI,
    mplsLcFrStdInterfaceConfRowStatus  RowStatus,

```

```
    mplsLcFrStdInterfaceConfStorageType StorageType
}

mplsLcFrStdTrafficMinDlci OBJECT-TYPE
    SYNTAX          DLCI
    MAX-ACCESS      read-create
    STATUS          current
    DESCRIPTION
        "This is the minimum DLCI value over which this
        LSR is willing to accept traffic on this
        interface."
    ::= { mplsLcFrStdInterfaceConfEntry 1 }

mplsLcFrStdTrafficMaxDlci OBJECT-TYPE
    SYNTAX          DLCI
    MAX-ACCESS      read-create
    STATUS          current
    DESCRIPTION
        "This is the max DLCI value over which this
        LSR is willing to accept traffic on this
        interface."
    ::= { mplsLcFrStdInterfaceConfEntry 2 }

mplsLcFrStdCtrlMinDlci OBJECT-TYPE
    SYNTAX          DLCI
    MAX-ACCESS      read-create
    STATUS          current
    DESCRIPTION
        "This is the min DLCI value over which this
        LSR is willing to accept control traffic
        on this interface."
    ::= { mplsLcFrStdInterfaceConfEntry 3 }

mplsLcFrStdCtrlMaxDlci OBJECT-TYPE
    SYNTAX          DLCI
    MAX-ACCESS      read-create
    STATUS          current
    DESCRIPTION
        "This is the max DLCI value over which this
        LSR is willing to accept control traffic
        on this interface."
    ::= { mplsLcFrStdInterfaceConfEntry 4 }

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

```

    "This object is used to create and
    delete entries in this table.  When configuring
    entries in this table, the corresponding ifEntry and
    mplsInterfaceConfEntry MUST exist beforehand.  If a manager
    attempts to create an entry for a corresponding
    mplsInterfaceConfEntry that does not support LC-FR,
    the agent MUST return an inconsistentValue error.
    If this table is implemented read-only, then the
    agent must set this object to active(1) when this
    row is made active.  If this table is implemented
    writable, then an agent MUST not allow modification
    to its objects once this value is set to active(1),
    except to mplsLcFrStdInterfaceConfRowStatus and
    mplsLcFrStdInterfaceConfStorageType."
 ::= { mplsLcFrStdInterfaceConfEntry 5 }

mplsLcFrStdInterfaceConfStorageType OBJECT-TYPE
    SYNTAX          StorageType
    MAX-ACCESS      read-create
    STATUS          current
    DESCRIPTION
        "The storage type for this conceptual row.
        Conceptual rows having the value 'permanent(4)'
        need not allow write-access to any columnar
        objects in the row."
    DEFVAL { nonVolatile }
    ::= { mplsLcFrStdInterfaceConfEntry 6 }

-- End of mplsLcFrStdInterfaceConfTable

-- Module compliance.

mplsLcFrStdCompliances
    OBJECT IDENTIFIER ::= { mplsLcFrStdConformance 1 }

mplsLcFrStdGroups
    OBJECT IDENTIFIER ::= { mplsLcFrStdConformance 2 }

-- Compliance requirement for full compliance

mplsLcFrStdModuleFullCompliance MODULE-COMPLIANCE
    STATUS current
    DESCRIPTION
        "Compliance statement for agents that provide
        full support for MPLS-LC-FR-STD-MIB.  Such
        devices can be monitored and also be configured
        using this MIB module."

```

```

MODULE -- this module
    MANDATORY-GROUPS {
        mplsLcFrStdIfGroup
    }

    OBJECT      mplsLcFrStdInterfaceConfRowStatus
    SYNTAX      RowStatus { active(1), notInService(2) }
    WRITE-SYNTAX RowStatus { active(1), notInService(2),
                            createAndGo(4), destroy(6)
                            }
    DESCRIPTION "Support for createAndWait and notReady is
                not required."

 ::= { mplsLcFrStdCompliances 1 }

-- Compliance requirement for read-only implementations.

mplsLcFrStdModuleReadOnlyCompliance MODULE-COMPLIANCE
    STATUS current
    DESCRIPTION
        "Compliance requirement for implementations that only
        provide read-only support for MPLS-LC-FR-STD-MIB.
        Such devices can be monitored but cannot be configured
        using this MIB module.
        "

MODULE -- this module
    MANDATORY-GROUPS {
        mplsLcFrStdIfGroup
    }

    -- mplsLcFrStdInterfaceConfTable

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

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

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

```



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

OBJECT      mplsLcFrStdInterfaceConfRowStatus
SYNTAX      RowStatus { active(1) }
MIN-ACCESS  read-only
DESCRIPTION  "Write access is not required."

OBJECT      mplsLcFrStdInterfaceConfStorageType
MIN-ACCESS  read-only
DESCRIPTION
    "Write access is not required."
::= { mplsLcFrStdCompliances 2 }

-- Units of conformance.

mplsLcFrStdIfGroup OBJECT-GROUP
OBJECTS {
    mplsLcFrStdTrafficMinDlci,
    mplsLcFrStdTrafficMaxDlci,
    mplsLcFrStdCtrlMinDlci,
    mplsLcFrStdCtrlMaxDlci,
    mplsLcFrStdInterfaceConfRowStatus,
    mplsLcFrStdInterfaceConfStorageType
}
STATUS      current

DESCRIPTION
    "Collection of objects needed for MPLS LC-FR
    interface configuration."
::= { mplsLcFrStdGroups 1 }

END
```

## 9. Acknowledgments

We wish to thank Joan Cucchiara and Carlos Pignataro for their comments on this document.

## 10. Security Considerations

It is clear that these MIB modules are potentially useful for monitoring MPLS LSRs supporting LC-ATM and/or LC-FR. These MIBs can also be used for configuration of certain objects, and anything that can be configured can be incorrectly configured, with potentially disastrous results.

There are a number of management objects defined in this MIB module with a MAX-ACCESS clause of read-write and/or read-create. Such objects may be considered sensitive or vulnerable in some network environments. The support for SET operations in a non-secure environment without proper protection can have a negative effect on network operations. These are the tables and objects and their sensitivity/vulnerability:

- o the MplsLcAtmStdInterfaceConfTable and mplsLcFrStdInterfaceConfTable collectively contain objects that may be used to provision MPLS LC or FR-enabled interfaces. Unauthorized access to objects in these tables could result in disruption of traffic on the network. This is especially true if traffic has been established over these interfaces. The use of stronger mechanisms such as SNMPv3 security should be considered where possible. Specifically, SNMPv3 VACM and USM MUST be used with any v3 agent that implements this MIB module. Administrators should consider whether read access to these objects should be allowed, since read access may be undesirable under certain circumstances.

Some of the readable objects in this MIB module (i.e., objects with a MAX-ACCESS other than not-accessible) may be considered sensitive or vulnerable in some network environments. It is thus important to control even GET and/or NOTIFY access to these objects and possibly to even encrypt the values of these objects when sending them over the network via SNMP. These are the tables and objects and their sensitivity/vulnerability:

- o the MplsLcAtmStdInterfaceConfTable and mplsLcFrStdInterfaceConfTable collectively show the LC-ATM and/or LC-FR interfaces, their associated configurations, and their linkages to other MPLS-related configuration and/or performance statistics. Administrators not wishing to reveal

this information should consider these objects sensitive/vulnerable and take precautions so they are not revealed.

SNMP versions prior to SNMPv3 did not include adequate security. Even if the network itself is secure (for example by using IPsec), even then, there is no control as to who on the secure network is allowed to access and GET/SET (read/change/create/delete) the objects in this MIB module.

It is RECOMMENDED that implementers consider the security features as provided by the SNMPv3 framework (see [RFC3410], section 8), including full support for the SNMPv3 cryptographic mechanisms (for authentication and privacy).

Further, deployment of SNMP versions prior to SNMPv3 is NOT RECOMMENDED. Instead, it is RECOMMENDED to deploy SNMPv3 and to enable cryptographic security. It is then a customer/operator responsibility to ensure that the SNMP entity giving access to an instance of this MIB module, is properly configured to give access to the objects only to those principals (users) that have legitimate rights to indeed GET or SET (change/create/delete) them.

## 11. IANA Considerations

As described in and as requested in the MPLS-TC-STD-MIB [RFC3811], MPLS-related standards track MIB modules should be rooted under the mplsStdMIB subtree. There are 2 MPLS MIB modules contained in this document; each of the following "IANA Considerations" subsections requested from IANA a new assignment under the mplsStdMIB subtree. New assignments can only be made via a Standards Action as specified in [RFC2434].

### 11.1. IANA Considerations for MPLS-LC-ATM-STD-MIB

The IANA has assigned { mplsStdMIB 9 } to the MPLS-LC-ATM-STD-MIB module specified in this document.

### 11.2. IANA Considerations for MPLS-LC-FR-STD-MIB

The IANA has assigned { mplsStdMIB 10 } to the MPLS-LC-FR-STD-MIB module specified in this document.

## 12. References

### 12.1. Normative References

- [RFC3034] Conta, A., Doolan, P., and A. Malis, "Use of Label Switching on Frame Relay Networks Specification", RFC 3034, January 2001.
- [RFC3035] Davie, B., Lawrence, J., McCloghrie, K., Rosen, E., Swallow, G., Rekhter, Y., and P. Doolan, "MPLS using LDP and ATM VC Switching", RFC 3035, January 2001.
- [RFC2115] Brown, C. and F. Baker, "Management Information Base for Frame Relay DTEs Using SMIV2", RFC 2115, September 1997.
- [RFC2514] Noto, M., Spiegel, E., and K. Tesink, "Definitions of Textual Conventions and OBJECT-IDENTITIES for ATM Management", RFC 2514, February 1999.
- [RFC2863] McCloghrie, K. and F. Kastenholz, "The Interfaces Group MIB", RFC 2863, June 2000.
- [RFC3031] Rosen, E., Viswanathan, A., and R. Callon, "Multiprotocol Label Switching Architecture", RFC 3031, January 2001.
- [RFC3811] Nadeau, T. and J. Cucchiara, "Definitions of Textual Conventions (TCs) for Multiprotocol Label Switching (MPLS) Management", RFC 3811, June 2004.
- [RFC3812] Srinivasan, C., Viswanathan, A., and T. Nadeau, "Multiprotocol Label Switching (MPLS) Traffic Engineering (TE) Management Information Base (MIB)", RFC 3812, June 2004.
- [RFC3813] Srinivasan, C., Viswanathan, A., and T. Nadeau, "Multiprotocol Label Switching (MPLS) Label Switching Router (LSR) Management Information Base (MIB)", RFC 3813, June 2004.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2578] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M., and S. Waldbusser, "Structure of Management Information Version 2 (SMIV2)", STD 58, RFC 2578, April 1999.

- [RFC2579] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M., and S. Waldbusser, "Textual Conventions for SMIV2", STD 58, RFC 2579, April 1999.
- [RFC2580] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M., and S. Waldbusser, "Conformance Statements for SMIV2", STD 58, RFC 2580, April 1999.

## 12.2. Informative References

- [RFC2434] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 2434, October 1998.
- [RFC3410] Case, J., Mundy, R., Partain, D., and B. Stewart, "Introduction and Applicability Statements for Internet-Standard Management Framework", RFC 3410, December 2002.
- [RFC3815] Cucchiara, J., Sjostrand, H., and J. Luciani, "Definitions of Managed Objects for the Multiprotocol Label Switching (MPLS), Label Distribution Protocol (LDP)", RFC 3815, June 2004.

## Authors' Addresses

Thomas D. Nadeau  
Cisco Systems, Inc.  
300 Beaver Brook Road  
Boxboro, MA 01719

Phone: +1-978-936-1470  
EMail: tnadeau@cisco.com

Subrahmanya Hegde  
Cisco Systems, Inc.  
225 East Tazman Drive  
San Jose, CA 95134

Phone: +1-408-525-6562  
EMail: subrah@cisco.com

## Full Copyright Statement

Copyright (C) The Internet Society (2006).

This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

## Intellectual Property

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at <http://www.ietf.org/ipr>.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at [ietf-ipr@ietf.org](mailto:ietf-ipr@ietf.org).

## Acknowledgement

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).

