

Network Working Group  
Internet Draft  
Expires: February 2004

Thomas D. Nadeau  
Cisco Systems, Inc.

Cheenu Srinivasan  
Bloomberg L.P.

Adrian Farrel  
Old Dog Consulting

Tim Hall  
Ed Harrison  
Data Connection Ltd.

August 2003

**Generalized Multiprotocol Label Switching (GMPLS)  
Label Switching Router (LSR) Management Information Base**

[draft-ietf-ccamp-gmpls-lsr-mib-01.txt](#)

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of [Section 10 of RFC 2026](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/1id-abstracts.txt>.

Abstract

This memo defines a portion of the Management Information Base (MIB) for use with network management protocols in the Internet community. In particular, it describes managed objects to configure and/or monitor a Generalized Multiprotocol Label Switching (GMPLS) Label Switching Router (LSRs).



## Table of Contents

1. Introduction	2
1.1. Migration Strategy	2
2. Terminology	3
3. The SNMP Management Framework	3
4. Outline	3
4.1. Summary of the GMPLS LSR MIB Module	4
5. Bidirectional LSPs	4
6. Example of LSP Setup	5
7. GMPLS Label Switching Router MIB Definitions	6
8. GMPLS Label MIB Definitions	14
9. Security Considerations	25
10. Acknowledgments	26
11. IANA Considerations	27
11.1. IANA Considerations for GMPLS-LSR-STD-MIB	27
11.2. IANA Considerations FOR GMPLS-LABEL-STD-MIB	27
12. References	27
12.1. Normative References	27
12.2. Informational References	28
13. Authors' Addresses	29
14. Full Copyright Statement	30
15. Intellectual Property Notice	30
16. Changes and Pending Work	31
16.1. Pending Work	31

## **1. Introduction**

This memo defines a portion of the Management Information Base (MIB) for use with network management protocols in the Internet community. In particular, it describes managed objects for modeling a Generalized Multiprotocol Label Switching (GMPLS) [[GMPLSArch](#)] Label Switching Router (LSR).

Comments should be made directly to the CCAMP mailing list at [ccamp@ops.ietf.org](mailto:ccamp@ops.ietf.org).

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](#)].

### **1.1. Migration Strategy**

This MIB module extends the Label Switching Router MIB module defined for use with MPLS [[LSRMIB](#)]. The only changes made are additions for support of GMPLS or changes that are necessary to support the increased complexity of a GMPLS system.

The companion document modeling and managing GMPLS based traffic engineering [[GMPLSTEMIB](#)] extends the MPLS TE MIB module [[TEMIB](#)] with the same intentions.

Textual conventions and OBJECT-IDENTIFIERS are defined in [[GMPLSTCMIB](#)] which extends the set of textual conventions originally defined in [[TCMIB](#)].

## 2. Terminology

This document uses terminology from the document describing the MPLS architecture [[RFC3031](#)] and the GMPLS architecture [[GMPLSArch](#)].

A label switched path (LSP) is modeled as a connection consisting of one or more incoming segments (in-segments) and/or one or more outgoing segments (out-segments) at an LSR. The association or interconnection of the in-segments and out-segments is accomplished by using a cross-connect. We use the terminology "connection" and "LSP" interchangeably where the meaning is clear from the context.

in-segment	This is analogous to a GMPLS label on an interface.
out-segment	This is analogous to a GMPLS label on an interface.
cross-connect	This describes the conceptual connection between a set of in-segments and out-segments. Note that either set may be 0; that is, a cross-connect may connect only out-segments together with no in-segments in the case where an LSP is originating on an LSR.

## 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. Outline

Configuring statically provisioned GMPLS LSPs through an LSR involves the following steps:

- Configuring an interface using the MPLS LSR MIB module.
- Enabling GMPLS on GMPLS capable interfaces using this MIB module.
- Configuring in-segments and out-segments using the MPLS LSR MIB module.

- Configuring GMPLS extensions to the in-segments and out-segments using this MIB module.
- Setting up the cross-connect table in the MPLS LSR MIB module to associate segments and/or to indicate connection origination and termination.

- Optionally setting up labels in the label table in this MIB module if the textual convention `MplsLabel` is not capable of holding the required label (for example, if the label requires more than 32 bits to encode it), or if the operator wishes to disambiguate GMPLS label types.
- Optionally specifying label stack actions in the MPLS LSR MIB module.
- Optionally specifying segment traffic parameters in the MPLS LSR MIB module.

#### **4.1. Summary of the GMPLS LSR MIB Module**

The MIB tables in this MIB module are as follows.

- The interface configuration table (`gmplsInterfaceTable`), which extends `mplsInterfaceTable` to enable the GMPLS protocol on MPLS-capable interfaces.
- The in-segment (`gmplsInSegmentTable`) and out-segment (`gmplsOutSegmentTable`) tables extend `mplsInSegmentTable` and `mplsOutSegmentTable` to configuring GMPLS-specific parameters for LSP segments at an LSR.
- The `gmplsLabelTable` extends `mplsLabelTable` and allows Generalized Labels to be defined and managed in a central location. Generalized Labels can be of variable length and have distinct bit-by-bit interpretations according to the use that is made of them.

These tables are described in the subsequent sections.

### **5. Bidirectional LSPs**

This MIB supports bidirectional LSPs as required for GMPLS. A single value of `mplsXCIndex` is shared by all of the segments for the entire bidirectional LSP. This facilitates a simple reference from [\[TEMIB\]](#) and [\[GMPLSTEMIB\]](#), and makes fate-sharing more obvious.

It is, however, important that the direction of segments is understood to avoid connecting all in-segments to all out-segments. This is achieved by an object in each segment that indicates the direction of the segment with respect to data flow.

A segment that is marked as 'forward' carries data from the 'head' of the LSP to the 'tail'. A segment marked as 'reverse' carries data in the reverse direction.

Where an LSP is signaled using a conventional signaling protocol, the 'head' of the LSP is the source of the signaling (also known as the ingress) and the 'tail' is the destination (also known as the egress). For manually configured LSPs an arbitrary decision must be made about which segments are 'forward' and which 'reverse'. For



consistency this decision should be made across all LSRs that participate in the LSP by assigning 'head' and 'tail' ends to the LSP.

## 6. Example of LSP Setup

In this section we provide a brief example of using the MIB objects described in [section 9](#) to set up an LSP. While this example is not meant to illustrate every nuance of the MIB, it is intended as an aid to understanding some of the key concepts. It is meant to be read after going through the MIB itself. A prerequisite is an understanding of [[LSRMIB](#)].

Suppose that one would like to manually create a best-effort, bi-directional LSP. Assume that, in the forward direction, the LSP enters the LSR via MPLS interface A with ifIndex 12 and exits the LSR via MPLS interface B with ifIndex 13. For the reverse direction, we assume the LSP enters via interface B and leaves via interface A (i.e. the forward and reverse directions use the same bi-directional interfaces). Let us also assume that we do not wish to have a label stack beneath the top label on the outgoing labeled packets. The following example illustrates which rows and corresponding objects might be created to accomplish this.

First, the traffic parameter entries must be set-up for both segments. These are configured entirely in the MPLS LSR MIB through the `mplsTrafficParamTable`.

Note that for a bi-directional LSP, the segments in the reverse direction MAY share the traffic parameter entries (and hence resources) with the segments in the forward direction.

We must next create the appropriate in-segment and out-segment entries with suitable traffic parameters by pointing to the appropriate traffic parameter entries that we have just created. This is also done in the MPLS LSR MIB using `mplsInSegmentTable` and `mplsOutSegmentTable`.

These table entries are extended by entries in `gmplsInSegmentTable` and `gmplsOutSegmentTable`. Note that the nature of the 'extends' relationship is that the entry in `gmplsInSegmentTable` has the same index values as the entry in `mplsInSegmentTable`. Similarly, the entry in `gmplsOutSegmentTable` has the same index values as the entry in `mplsOutSegmentTable`.

First for the forward direction:

```
In gmplsInSegmentTable
{
```

```
    gmplsInSegmentDirection      = forward (1),  
}  
  
In gmplsOutSegmentTable  
{  
    gmplsOutSegmentDirection     = forward (1)  
}
```

Next for the reverse direction:

```
In gmplsInSegmentTable
{
    gmplsInSegmentDirection      = reverse (2)
}

In gmplsOutSegmentTable
{
    gmplsOutSegmentDirection     = reverse (2)
}
```

Next, two cross-connect entries are created in the mplsXCTable of the MPLS LSR MIB, thereby associating the newly created segments together.

## 7. GMPLS Label Switching Router MIB Definitions

GMPLS-LSR-STD-MIB DEFINITIONS ::= BEGIN

IMPORTS

```
    MODULE-IDENTITY, OBJECT-TYPE, Unsigned32
        FROM SNMPv2-SMI
    MODULE-COMPLIANCE, OBJECT-GROUP
        FROM SNMPv2-CONF
    GmplsSegmentDirection
        FROM GMPLS-TC-STD-MIB
;
```

gmplsLsrStdMIB MODULE-IDENTITY

LAST-UPDATED

"200308190900Z " -- 19 August 2003 9:00:00 GMT"

ORGANIZATION

"Common Control And Management Protocols (CCAMP)  
Working Group"

CONTACT-INFO

" Thomas D. Nadeau  
Cisco Systems, Inc.  
Email: tnadeau@cisco.com

Cheenu Srinivasan  
Bloomberg L.P.  
Email: cheenu@bloomberg.net

Adrian Farrel  
Old Dog Consulting  
Email: adrian@olddog.co.uk

Ed Harrison

Data Connection Ltd.  
Email: ed.harrison@dataconnection.com

Tim Hall  
Data Connection Ltd.  
Email: timhall@dataconnection.com

Comments about this document should be emailed direct to the  
CCAMP working group mailing list at [ccamp@ops.ietf.org](mailto:ccamp@ops.ietf.org)"

## DESCRIPTION

"This MIB module contains managed object definitions  
for the Generalized Multiprotocol Label Switching  
(GMPLS) Router as defined in: Ashwood-Smith et al.,  
Generalized Multiprotocol Label Switching (GMPLS)  
Architecture, Internet Draft <[draft-many-gmpls-  
architecture-01.txt](#)>, March 2001, work in progress.

Copyright (C) The Internet Society (2003). This  
version of this MIB module is part of RFCXXX; see  
the RFC itself for full legal notices."

-- Revision history.

## REVISION

"200308190900Z" -- 19 August 2003 09:00:00 GMT

## DESCRIPTION

"Initial revision, published as part of RFC XXXX."

::= { gmplsStdMIB xx }

-- Top level components of this MIB module.

-- Notifications

-- no notifications are currently defined.

gmplsLsrNotifications OBJECT IDENTIFIER ::= { gmplsLsrStdMIB 0 }

-- Tables, Scalars

gmplsLsrObjects OBJECT IDENTIFIER ::= { gmplsLsrStdMIB 1 }

-- Conformance

gmplsLsrConformance OBJECT IDENTIFIER ::= { gmplsLsrStdMIB 2 }

-- GMPLS Interface Table.

gmplsInterfaceTable OBJECT-TYPE

SYNTAX SEQUENCE OF GmplsInterfaceEntry

MAX-ACCESS not-accessible

STATUS current

## DESCRIPTION

"This table specifies per-interface GMPLS capability  
and associated information. It extends the  
information in mplsInterfaceTable."

::= { gmplsLsrObjects 1 }

gmplsInterfaceEntry OBJECT-TYPE

SYNTAX GmplsInterfaceEntry

MAX-ACCESS not-accessible

STATUS current

#### DESCRIPTION

"A conceptual row in this table is created automatically by an LSR for every interface capable of supporting GMPLS and which is configured to do so. A conceptual row in this table will exist if and only if a corresponding entry in mplsInterfaceTable exists, and a corresponding entry in ifTable exists

with ifType = mpls(166). If the associated entry in ifTable is operationally disabled (thus removing the GMPLS capabilities on the interface) or the entry in mplsInterfaceTable is deleted, the corresponding entry in this table MUST be deleted shortly thereafter.

The indexing is the same as that for mplsInterfaceTable. Thus, the entry with index 0 represents the per-platform label space and contains parameters that apply to all interfaces that participate in the per-platform label space."

```
INDEX { mplsInterfaceIndex }  
 ::= { gmplsInterfaceTable 1 }
```

```
GmplsInterfaceEntry ::= SEQUENCE {  
    gmplsInterfaceSignalingCaps    BITS  
}
```

gmplsInterfaceSignalingCaps OBJECT-TYPE

```
SYNTAX BITS {  
    rsvp-gmpls (1),  
    crldp-gmpls (2), -- note the use of CR-LDP is deprecated  
    other-gmpls (3)  
}
```

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"Defines the signaling capabilities on this interface. Multiple bits may legitimately be set at once. Setting no bits implies that GMPLS signaling cannot be performed on this interface and all LSPs must be manually provisioned."

```
::= { gmplsInterfaceEntry 1 }
```

-- End of gmplsInterfaceTable

-- In-segment table.

gmplsInSegmentTable OBJECT-TYPE

```
SYNTAX SEQUENCE OF GmplsInSegmentEntry
```

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table extends the mplsInSegmentTable to provide GMPLS-specific information about incoming segments to an LSR."

```
::= { gmplsLsrObjects 2 }
```

gmplsInSegmentEntry OBJECT-TYPE  
SYNTAX GmplsInSegmentEntry  
MAX-ACCESS not-accessible  
STATUS current  
DESCRIPTION

"An entry in this table extends the representation of  
an incoming segment represented by an entry in  
mplsInSegmentTable. An entry can be created by a



network administrator or an SNMP agent, or a GMPLS signaling protocol.

Note that the storage type for this entry SHOULD be inherited from the corresponding entry in the `mplsInSegmentTable` given by the value of the `mplsInSegmentStorageType` object."

```
INDEX { mplsInSegmentIndex }  
::= { gmplsInSegmentTable 1 }
```

```
GmplsInSegmentEntry ::= SEQUENCE {  
    gmplsInSegmentDirection    GmplsSegmentDirection  
}
```

`gmplsInSegmentDirection` OBJECT-TYPE

SYNTAX GmplsSegmentDirection

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"This object indicates the direction of data flow on this segment. This object cannot be modified if `mplsInSegmentRowStatus` for the associated entry in the `mplsInSegmentTable` is `active(1)`."

DEFVAL { forward }

```
::= { gmplsInSegmentEntry 1 }
```

-- End of `gmplsInSegmentTable`

-- Out-segment table.

`gmplsOutSegmentTable` OBJECT-TYPE

SYNTAX SEQUENCE OF GmplsOutSegmentEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table extends the `mplsOutSegmentTable` to provide GMPLS-specific information about outgoing segments from an LSR."

```
::= { gmplsLsrObjects 3 }
```

`gmplsOutSegmentEntry` OBJECT-TYPE

SYNTAX GmplsOutSegmentEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"An entry in this table extends the representation of an outgoing segment represented by an entry in `mplsOutSegmentTable`. An entry can be created by a

network administrator or an SNMP agent, or a GMPLS signaling protocol.

Note that the storage type for this entry SHOULD be inherited from the corresponding entry in the mplsOutSegmentTable given by the value of the mplsOutSegmentStorageType object."

```
INDEX { mplsOutSegmentIndex }  
::= { gmplsOutSegmentTable 1 }
```

```
GmplsOutSegmentEntry ::= SEQUENCE {
    gmplsOutSegmentDirection      GmplsSegmentDirection,
    gmplsOutSegmentTTLDecrement   Unsigned32
}

gmplsOutSegmentDirection OBJECT-TYPE
    SYNTAX      GmplsSegmentDirection
    MAX-ACCESS   read-create
    STATUS      current
    DESCRIPTION
        "This object indicates the direction of data flow on
        this segment. This object cannot be modified if
        mplsOutSegmentRowStatus for the associated entry in
        the mplsOutSegmentTable is active(1)."
```

DEFVAL { forward }

```
::= { gmplsOutSegmentEntry 1 }
```

gmplsOutSegmentTTLDecrement OBJECT-TYPE

```
    SYNTAX      Unsigned32 (0..255)
    MAX-ACCESS   read-create
    STATUS      current
    DESCRIPTION
        "This object indicates the amount by which to
        decrement the TTL of any payload packets forwarded
        on this segment if per-hop decrementing is being
        done.
```

A value of zero indicates that no decrement should be made or that per-hop decrementing is not in force.

See the gmplsTunnelTTLDecrement object in the gmplsTunnelTable of [\[GMPLSTEMIB\]](#) for a value by which to decrement the TTL for the whole of a tunnel.

This object cannot be modified if mplsOutSegmentRowStatus for the associated entry in the mplsOutSegmentTable is active(1)."

```
    DEFVAL { 0 }
    ::= { gmplsOutSegmentEntry 2 }
```

-- End of gmplsOutSegmentTable

-- Module compliance.

gmplsLsrGroups

```
    OBJECT IDENTIFIER ::= { gmplsLsrConformance 1 }
```

gmplsLsrCompliances

OBJECT IDENTIFIER ::= { gmplsLsrConformance 2 }

-- Compliance requirement for fully compliant implementations.

gmplsLsrModuleFullCompliance MODULE-COMPLIANCE

STATUS current

DESCRIPTION

"Compliance statement for agents that provide full support for GMPLS-LSR-STD-MIB."

MODULE IF-MIB -- The Interfaces Group MIB, [RFC 2863](#).

MANDATORY-GROUPS {

ifGeneralInformationGroup,

ifCounterDiscontinuityGroup

}

MODULE MPLS-LSR-STD-MIB ù The MPLS LSR MIB

MANDATORY-GROUPS {

mplsInterfaceGroup,

mplsInSegmentGroup,

mplsOutSegmentGroup,

mplsXCGroup,

mplsPerfGroup,

mplsLsrNotificationGroup

}

MODULE -- this module

MANDATORY-GROUPS {

gmplsInterfaceGroup,

gmplsInSegmentGroup,

gmplsOutSegmentGroup

}

-- gmplsInSegmentTable

OBJECT gmplsInSegmentDirection

SYNTAX GmplsSegmentDirection

MIN-ACCESS read-write

DESCRIPTION

"Only forward(1) needs to be supported by implementations that only support unidirectional LSPs."

-- gmplsOutSegmentTable

OBJECT gmplsOutSegmentDirection

SYNTAX GmplsSegmentDirection

MIN-ACCESS read-write

DESCRIPTION

"Only forward(1) needs to be supported by  
implementations that only support unidirectional  
LSPs."

OBJECT        gmplsOutSegmentTTLDcrement  
SYNTAX        Unsigned32 (0..255)  
MIN-ACCESS    read-only  
DESCRIPTION

    "Write access is not required."

::= { gmplsLsrCompliances 1 }

-- Compliance requirement for implementations that provide read-only  
-- access.

gmplsLsrModuleReadOnlyCompliance MODULE-COMPLIANCE

    STATUS current

    DESCRIPTION

        "Compliance requirement for implementations that only  
        provide read-only support for GMPLS-LSR-STD-MIB. Such  
        devices can then be monitored but cannot be configured  
        using this MIB modules."

MODULE IF-MIB -- The interfaces Group MIB, [RFC 2863](#)

MANDATORY-GROUPS {  
    ifGeneralInformationGroup,  
    ifCounterDiscontinuityGroup  
}

MODULE MPLS-LSR-STD-MIB

MANDATORY-GROUPS {  
    mplsInterfaceGroup,  
    mplsInSegmentGroup,  
    mplsOutSegmentGroup,  
    mplsXCGroup,  
    mplsPerfGroup  
}

MODULE -- this module

MANDATORY-GROUPS {  
    gmplsInterfaceGroup,  
    gmplsInSegmentGroup,  
    gmplsOutSegmentGroup  
}

-- gmplsInterfaceGroup

OBJECT        gmplsInterfaceSignalingCaps  
SYNTAX        BITS  
MIN-ACCESS    read-only  
DESCRIPTION

"Write access is not required."



```
-- gmplsInSegmentTable

OBJECT      gmplsInSegmentDirection
SYNTAX      GmplsSegmentDirection
MIN-ACCESS  read-only
DESCRIPTION
    "Write access is not required. Only forward(1) needs
     to be supported by implementations that only support
     unidirectional LSPs."

-- gmplsOutSegmentTable

OBJECT      gmplsOutSegmentDirection
SYNTAX      GmplsSegmentDirection
MIN-ACCESS  read-only
DESCRIPTION
    "Write access is not required. Only forward(1) needs
     to be supported by implementations that only support
     unidirectional LSPs."

OBJECT      gmplsOutSegmentTTLDcrement
SYNTAX      Unsigned32 (0..255)
MIN-ACCESS  read-only
DESCRIPTION
    "Write access is not required."

::= { gmplsLsrCompliances 2 }

-- Units of conformance.

gmplsInterfaceGroup OBJECT-GROUP
  OBJECTS {
    gmplsInterfaceSignalingCaps
  }
  STATUS current
  DESCRIPTION
    "Collection of objects needed for GMPLS interface
     configuration and performance information."
::= { gmplsLsrGroups 1 }

gmplsInSegmentGroup OBJECT-GROUP
  OBJECTS {
    gmplsInSegmentDirection
  }
  STATUS current
  DESCRIPTION
    "Collection of objects needed to implement a GMPLS
     in-segment."
::= { gmplsLsrGroups 2 }
```



```
gmplsOutSegmentGroup  OBJECT-GROUP
  OBJECTS {
    gmplsOutSegmentDirection,
    gmplsOutSegmentTTLDcrement
  }
  STATUS  current
  DESCRIPTION
    "Collection of objects needed to implement a GMPLS
    out-segment."
 ::= { gmplsLsrGroups 3 }

END
```

## 8. GMPLS Label MIB Definitions

```
GMPLS-LABEL-STD-MIB DEFINITIONS ::= BEGIN

IMPORTS
  MODULE-IDENTITY, OBJECT-TYPE, Unsigned32
    FROM SNMPv2-SMI
  MODULE-COMPLIANCE, OBJECT-GROUP
    FROM SNMPv2-CONF
  RowStatus, StorageType
    FROM SNMPv2-TC
  InterfaceIndexOrZero
    FROM IF-MIB
  MplsLabel
    FROM MPLS-TC-STD-MIB
  GmplsGeneralizedLabelTypes, GmplsFreeformLabel
    FROM GMPLS-TC-STD-MIB
;

gmplsLabelStdMIB MODULE-IDENTITY
  LAST-UPDATED
    "200308190900Z " -- 19 August 2003 9:00:00 GMT"
  ORGANIZATION
    "Common Control And Management Protocols (CCAMP)
    Working Group"
  CONTACT-INFO
    "      Thomas D. Nadeau
      Cisco Systems, Inc.
      Email: tnadeau@cisco.com

      Cheenu Srinivasan
      Bloomberg L.P.
      Email: cheenu@bloomberg.net

      Adrian Farrel
```

Old Dog Consulting  
Email: [adrian@olddog.co.uk](mailto:adrian@olddog.co.uk)

Ed Harrison  
Data Connection Ltd.  
Email: [ed.harrison@dataconnection.com](mailto:ed.harrison@dataconnection.com)

Tim Hall  
Data Connection Ltd.  
Email: [timhall@dataconnection.com](mailto:timhall@dataconnection.com)

Nadeau, Srinivasan, Farrel, Hall and Harrison

[Page 14]

Comments about this document should be emailed direct to the  
CCAMP working group mailing list at [ccamp@ops.ietf.org](mailto:ccamp@ops.ietf.org)"

## DESCRIPTION

"This MIB module contains managed object definitions  
for labels within GMPLS systems."

-- Revision history.

## REVISION

"200308190900Z" -- 19 August 2003 09:00:00 GMT

## DESCRIPTION

"Initial revision, published as part of RFC XXXX."

::= { gmplsStdMIB 2 }

-- Top level components of this MIB module.

-- Notifications

-- no notifications are currently defined.

gmplsLsrNotifications OBJECT IDENTIFIER ::= { gmplsLabelStdMIB 0 }

-- Tables, Scalars

gmplsLabelObjects OBJECT IDENTIFIER ::= { gmplsLabelStdMIB 1 }

-- Conformance

gmplsLabelConformance OBJECT IDENTIFIER ::= { gmplsLabelStdMIB 2 }

-- GMPLS Label Table.

gmplsLabelTable OBJECT-TYPE

SYNTAX SEQUENCE OF GmplsLabelEntry

MAX-ACCESS not-accessible

STATUS current

## DESCRIPTION

"Table of GMPLS Labels. This table allows the  
representation of the more complex label forms  
required for GMPLS which cannot be held within the  
textual convention MplsLabel. That is labels that  
cannot be encoded within 32 bits. It is,  
nevertheless also capable of holding 32 bit labels  
or regular MPLS labels if desired.

Each entry in this table represents an individual  
GMPLS label value. Labels in the tables in other  
MIBs are referred to using row pointer into this  
table. The indexing of this table provides for  
arbitrary indexing and also for concatenation of  
labels. "

::= { gmplsLabelObjects 1 }

gmplsLabelEntry OBJECT-TYPE

SYNTAX GmplsLabelEntry

MAX-ACCESS      not-accessible

STATUS           current

DESCRIPTION

"An entry in this table represents a single label value. There are three indexes into the table.

- The interface index may be helpful to distinguish which labels are in use on which interfaces or to

handle cases where there are a very large number of labels in use in the system. When label representation is desired to apply to the whole system or when it is not important to distinguish labels by their interfaces, this index MAY be set to zero.

- The label index provides a way of identifying the label.
- The label sub-index is only used for concatenated labels. It identifies each component label. When non-concatenated labels are used, this index SHOULD be set to zero.

A storage type object is supplied to control the storage type for each entry, but implementations should note that the storage type of conceptual rows in other tables that include row pointers to an entry in this table SHOULD dictate the storage type of the rows in this table where the row in the other table is more persistent."

```
INDEX {  
    gmplsLabelInterface,  
    gmplsLabelIndex,  
    gmplsLabelSubindex }  
::= { gmplsLabelTable 1 }
```

```
GmplsLabelEntry ::= SEQUENCE {  
    gmplsLabelInterface      InterfaceIndexOrZero,  
    gmplsLabelIndex          Unsigned32,  
    gmplsLabelSubindex       Unsigned32,  
    gmplsLabelType           GmplsGeneralizedLabelTypes,  
    gmplsLabelMplsLabel      MplsLabel,  
    gmplsLabelPortWavelength Unsigned32,  
    gmplsLabelFreeformLength INTEGER,  
    gmplsLabelFreeform       GmplsFreeformLabel,  
    gmplsLabelSonetSdhSignalIndex INTEGER,  
    gmplsLabelSdhVc          INTEGER,  
    gmplsLabelSdhVcBranch    INTEGER,  
    gmplsLabelSonetSdhBranch  INTEGER,  
    gmplsLabelSonetSdhGroupBranch INTEGER,  
    gmplsLabelWavebandId     Unsigned32,  
    gmplsLabelWavebandStartLabel Unsigned32,  
    gmplsLabelWavebandEndLabel Unsigned32,  
    gmplsLabelRowStatus       RowStatus,  
    gmplsLabelStorageType     StorageType  
}
```

```
gmplsLabelInterface OBJECT-TYPE  
    SYNTAX      InterfaceIndexOrZero
```

```
MAX-ACCESS    not-accessible
STATUS        current
DESCRIPTION
    "The interface on which this label is used. If the
      label has or could have applicability across the
      whole system, this object SHOULD be set to zero."
 ::= { gmplsLabelEntry 1 }
```



**gmplsLabelIndex OBJECT-TYPE**

SYNTAX Unsigned32 (0..4294967295)

MAX-ACCESS not-accessible

STATUS current

**DESCRIPTION**

"An arbitrary index into the table to identify a label.

Note that implementations that are representing 32 bit labels within this table MAY choose to align this index with the value of the label, but should be aware of the implications of sparsely populated tables."

::= { gmplsLabelEntry 2 }

**gmplsLabelSubindex OBJECT-TYPE**

SYNTAX Unsigned32 (0..4294967295)

MAX-ACCESS not-accessible

STATUS current

**DESCRIPTION**

"In conjunction with gmplsLabelInterface and gmplsLabelIndex, this object uniquely identifies this row. This sub-index allows a single GMPLS label to be defined as a concatenation of labels. This is particularly useful in TDM.

The ordering of sub-labels is strict with the sub-label with lowest gmplsLabelSubindex appearing first. Note that all sub-labels of a single GMPLS label must share the same gmplsLabelInterface and gmplsLabelIndex values. For labels that are not composed of concatenated sub-labels, this value SHOULD be set to zero."

::= { gmplsLabelEntry 3 }

**gmplsLabelType OBJECT-TYPE**

SYNTAX GmplsGeneralizedLabelTypes

MAX-ACCESS read-create

STATUS current

**DESCRIPTION**

"Identifies the type of this label. Note that this object does not determine whether MPLS or GMPLS signaling is in use: a value of gmplsMplsLabel (1) denotes that a 23 bit MPLS packet label is present, but does not describe whether this is signaled using MPLS or GMPLS.

The value of this object helps determine which of the following objects are valid.

This object cannot be modified if gmplsLabelRowStatus is active(1)."

::= { gmplsLabelEntry 4 }

gmplsLabelMplsLabel OBJECT-TYPE

SYNTAX MplsLabel

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The value of an MPLS label (that is a packet label)  
if this table is used to store it. This may be used

in MPLS systems even though the label values can be adequately stored in the MPLS MIB modules. Further, in mixed MPLS and GMPLS systems it may be advantageous to store all labels in a single label table. Lastly, in GMPLS systems where packet labels are used (that is in systems that use GMPLS signaling and GMPLS labels for packet switching) it may be desirable to use this table.

This object is only valid if gmplsLabelType is set to gmplsMplsLabel (1).

This object cannot be modified if gmplsLabelRowStatus is active(1)."

::= { gmplsLabelEntry 6 }

gmplsLabelPortWavelength OBJECT-TYPE

SYNTAX Unsigned32

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The value of a Port or Wavelength Label when carried as a Generalized Label. Only valid if gmplsLabelType is set to gmplsPortWavelengthLabel(2).

This object cannot be modified if gmplsLabelRowStatus is active(1)."

::= { gmplsLabelEntry 7 }

gmplsLabelFreeformLength OBJECT-TYPE

SYNTAX INTEGER (1..64)

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The length of a freeform Generalized Label indicated in the gmplsFreeformLabel object. Only valid if gmplsLabelType is set to gmplsFreeformGeneralizedLabel(3).

This object cannot be modified if gmplsLabelRowStatus is active(1)."

::= { gmplsLabelEntry 8 }

gmplsLabelFreeform OBJECT-TYPE

SYNTAX GmplsFreeformLabel

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The value of a freeform Generalized Label that does not conform to one of the standardized label encoding or that an implementation chooses to

represent as an octet string without further

decoding. The length of this object is given by the value of `gmplsFreeformLength`. Only valid if `gmplsLabelType` is set to `gmplsFreeformGeneralizedLabel(3)`. This object cannot be modified if `gmplsLabelRowStatus` is `active(1)`."

```
::= { gmplsLabelEntry 9 }
```

**gmplsLabelSonetSdhSignalIndex OBJECT-TYPE**

SYNTAX INTEGER (0..4095)

MAX-ACCESS read-create

STATUS current

**DESCRIPTION**

"The Signal Index value (S) of a SONET or SDH Generalized Label. Zero indicates that this field is not significant. Only valid if gmplsLabelType is set to gmplsSonetLabel(4) or gmplsSdhLabel(5).

This object cannot be modified if gmplsLabelRowStatus is active(1)."

::= { gmplsLabelEntry 10 }

**gmplsLabelSdhVc OBJECT-TYPE**

SYNTAX INTEGER (0..15)

MAX-ACCESS read-create

STATUS current

**DESCRIPTION**

"The VC Indicator (U) of an SDH Generalized Label. Zero indicates that this field is non-significant. Only valid if gmplsLabelType is set to gmplsSdhLabel(5).

This object cannot be modified if gmplsLabelRowStatus is active(1)."

::= { gmplsLabelEntry 11 }

**gmplsLabelSdhVcBranch OBJECT-TYPE**

SYNTAX INTEGER (0..15)

MAX-ACCESS read-create

STATUS current

**DESCRIPTION**

"The VC Branch Indicator (K) of an SDH Generalized Label. Zero indicates that this field is non-significant. Only valid if gmplsLabelType is set to gmplsSdhLabel(5).

This object cannot be modified if gmplsLabelRowStatus is active(1)."

::= { gmplsLabelEntry 12 }

**gmplsLabelSonetSdhBranch OBJECT-TYPE**

SYNTAX INTEGER (0..15)

MAX-ACCESS read-create

STATUS current

**DESCRIPTION**

"The Branch Indicator (L) of a SONET or SDH Generalized Label. Zero indicates that this field is non-significant. Only valid gmplsLabelType is set to gmplsSonetLabel(4) or gmplsSdhLabel(5).

This object cannot be modified if

```
    gmplsLabelRowStatus is active(1)."  
 ::= { gmplsLabelEntry 13 }
```

**gmplsLabelSonetSdhGroupBranch OBJECT-TYPE**

SYNTAX INTEGER (0..15)

MAX-ACCESS read-create

STATUS current

## DESCRIPTION

"The Group Branch Indicator (M) of a SONET or SDH Generalized Label. Zero indicates that this field is non-significant. Only valid if gmplsLabelType is set to gmplsSonetLabel(4) or gmplsSdhLabel(5).

This object cannot be modified if gmplsLabelRowStatus is active(1)."

::= { gmplsLabelEntry 14 }

**gmplsLabelWavebandId OBJECT-TYPE**

SYNTAX Unsigned32

MAX-ACCESS read-create

STATUS current

## DESCRIPTION

"The waveband identifier component of a waveband label. Only valid if gmplsLabelType is set to gmplsWavebandLabel(6).

This object cannot be modified if gmplsLabelRowStatus is active(1)."

::= { gmplsLabelEntry 15 }

**gmplsLabelWavebandStart OBJECT-TYPE**

SYNTAX Unsigned32

MAX-ACCESS read-create

STATUS current

## DESCRIPTION

"The starting label component of a waveband label. Only valid if gmplsLabelType is set to gmplsWavebandLabel(6).

This object cannot be modified if gmplsLabelRowStatus is active(1)."

::= { gmplsLabelEntry 16 }

**gmplsLabelWavebandEnd OBJECT-TYPE**

SYNTAX Unsigned32

MAX-ACCESS read-create

STATUS current

## DESCRIPTION

"The end label component of a waveband label. Only valid if gmplsLabelType is set to gmplsWavebandLabel(6).

This object cannot be modified if gmplsLabelRowStatus is active(1)."

::= { gmplsLabelEntry 17 }





**gmplsLabelRowStatus OBJECT-TYPE**

SYNTAX RowStatus  
MAX-ACCESS read-create  
STATUS current

**DESCRIPTION**

"This variable is used to create, modify, and/or delete a row in this table. When a row in this table has a row in the active(1) state, no objects in this row can be modified except the gmplsLabelRowStatus and gmplsLabelStorageType."

::= { gmplsLabelEntry 18 }

**gmplsLabelStorageType OBJECT-TYPE**

SYNTAX StorageType  
MAX-ACCESS read-create  
STATUS current

**DESCRIPTION**

"This variable indicates the storage type for this object.

The agent MUST ensure that this object's value remains consistent with the storage type of any rows in other tables that contain pointers to this row. In particular, the storage type of this row must be at least as permanent as that of any row that point to it.

Conceptual rows having the value 'permanent' need not allow write-access to any columnar objects in the row."

**REFERENCE**

"See [RFC2579](#)."

DEFVAL { volatile }

::= { gmplsLabelEntry 19 }

-- End of GMPLS Label Table

-- Module compliance.

**gmplsLabelGroups**

OBJECT IDENTIFIER ::= { gmplsLabelConformance 1 }

**gmplsLabelCompliances**

OBJECT IDENTIFIER ::= { gmplsLabelConformance 2 }

**gmplsLabelModuleFullCompliance MODULE-COMPLIANCE**

STATUS current

**DESCRIPTION**

"Compliance statement for agents that support

the GMPLS Label MIB module."

MODULE -- this module

-- The mandatory groups have to be implemented by  
-- LSRs claiming support for this MIB module. This MIB module is,  
-- however, not mandatory for a working implementation of a GMPLS  
-- LSR with full MIB support if the GMPLS labels in use can be  
-- represented within a 32 bit quantity.

MANDATORY-GROUPS {  
 gmplsLabelTableGroup  
}

-- Units of conformance.

GROUP gmplsLabelTableGroup

DESCRIPTION

"This group is mandatory for devices which support  
the gmplsLabelTable."

GROUP gmplsLabelPacketGroup

DESCRIPTION

"This group extends gmplsLabelTableGroup for  
implementations that support packet labels."

GROUP gmplsLabelPortWavelengthGroup

DESCRIPTION

"This group extends gmplsLabelTableGroup for  
implementations that support port and wavelength  
labels."

GROUP gmplsLabelFreeformGroup

DESCRIPTION

"This group extends gmplsLabelTableGroup for  
implementations that support freeform labels."

GROUP gmplsLabelSonetSdhGroup

DESCRIPTION

"This group extends gmplsLabelTableGroup for  
implementations that support SONET or SDH labels."

GROUP gmplsLabelWavebandGroup

DESCRIPTION

"This group extends gmplsLabelTableGroup for  
implementations that support Waveband labels."

-- gmplsLabelTable

OBJECT gmplsLabelType

MIN-ACCESS read-only

DESCRIPTION

"Write access is not required."

OBJECT gmpIsLabelMplsLabel

MIN-ACCESS read-only

DESCRIPTION

"Write access is not required."

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

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

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

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

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

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

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

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

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

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

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

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

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

::= { gmplsLabelCompliances 1 }

-- Units of conformance.

gmplsLabelTableGroup OBJECT-GROUP
OBJECTS {
    gmplsLabelType,
    gmplsLabelSubtype,
    gmplsLabelRowStatus,
    gmplsLabelStorageType
}
STATUS     current
DESCRIPTION
    "Necessary, but not sufficient, set of objects to
    implement label table support.  In addition,
    depending on the type of labels supported (for
    example, wavelength labels), the following other
    groups defined below are mandatory:
    gmplsLabelPacketGroup and/or
    gmplsLabelPortWavelengthGroup and/or
    gmplsLabelFreeformGroup and/or
    gmplsLabelSonetSdhGroup."
::= { gmplsLabelGroups 1 }

gmplsLabelPacketGroup OBJECT-GROUP
OBJECTS {
    gmplsLabelMplsLabel
}
STATUS     current
```

DESCRIPTION

"Object needed to implement Packet (MPLS) labels."  
::= { gmplsLabelGroups 2 }



```
gmplsLabelPortWavelengthGroup OBJECT-GROUP
  OBJECTS {
    gmplsLabelPortWavelength
  }
  STATUS current
  DESCRIPTION
    "Object needed to implement Port and Wavelength
    labels."
 ::= { gmplsLabelGroups 3 }

gmplsLabelFreeformGroup OBJECT-GROUP
  OBJECTS {
    gmplsLabelFreeformLength,
    gmplsLabelFreeform
  }
  STATUS current
  DESCRIPTION
    "Object needed to implement Freeform labels."
 ::= { gmplsLabelGroups 4 }

gmplsLabelSonetSdhGroup OBJECT-GROUP
  OBJECTS {
    gmplsLabelSonetSdhSignalIndex,
    gmplsLabelSdhVc,
    gmplsLabelSdhVcBranch,
    gmplsLabelSonetSdhBranch,
    gmplsLabelSonetSdhGroupBranch
  }
  STATUS current
  DESCRIPTION
    "Object needed to implement SONET and SDH labels."
 ::= { gmplsLabelGroups 5 }

gmplsLabelWavebandGroup OBJECT-GROUP
  OBJECTS {
    gmplsLabelWavebandId,
    gmplsLabelWavebandStart,
    gmplsLabelWavebandEnd
  }
  STATUS current
  DESCRIPTION
    "Object needed to implement Waveband labels."
 ::= { gmplsLabelGroups 6 }

END
```

## **9. Security Considerations**

It is clear that the MIB modules described in this document in association with the MPLS-LSR-STD-MIB are potentially useful for monitoring of GMPLS LSRs. These MIB modules 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 these MIB modules 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 gmplsInterfaceTable, gmplsInSegmentTable, gmplsOutSegmentTable and gmplsLabelTable collectively contain objects to provision GMPLS interfaces, LSPs and their associated parameters on a Label Switching Router (LSR). Unauthorized write access to objects in these tables, could result in disruption of traffic on the network. This is especially true if an LSP has already been established. The use of stronger mechanisms such as SNMPv3 security should be considered where possible. Specifically, SNMPv3 VACM and USM MUST be used with any SNMPv3 agent which implements these MIB modules.

Some of the readable objects in these MIB modules "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 gmplsInterfaceTable, gmplsInSegmentTable, gmplsOutSegmentTable and gmplsLabelTable collectively show the LSP network topology and its capabilities. If an Administrator does not want to reveal this information, then these tables should be considered sensitive/vulnerable.

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 these MIB modules. 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.

## **10. Acknowledgments**

This draft extends [[LSRMIB](#)]. The authors would like to express their gratitude to all those who worked on that earlier MIB document.

The authors would like to express their thanks to Dan Joyle for his careful review and comments on early versions of the Label Table.

## **11. IANA Considerations**

MPLS related standards track MIB modules are rooted under the mplsStdMIB subtree.

One of the MIB modules contained in this document extends tables contained in MPLS MIB modules.

As requested in requested in the GMPLS-TC-STD-MIB [[GMPLSTCMIB](#)] the two MIB modules contained in this document should be placed in the mplsStdMIB subtree as well.

New assignments can only be made via a Standards Action as specified in [[RFC2434](#)].

### **11.1. IANA Considerations for GMPLS-LSR-STD-MIB**

The IANA is requested to assign { mplsStdMIB xx } to the GMPLS-LSR-STD-MIB module specified in this document.

### **11.2. IANA Considerations for GMPLS-LABEL-STD-MIB**

The IANA is requested to assign { mplsStdMIB xx } to the GMPLS-LABEL-STD-MIB module specified in this document.

## **12. References**

### **12.1. Normative References**

- [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.
- [RFC2863] McCloghrie, K. and F. Kastenholz, "The Interfaces Group MIB", [RFC 2863](#), June 2000.

[RFC3471]

Berger, L. (Editor), "Generalized Multi-Protocol  
Label Switching (GMPLS) Signaling Functional  
Description", [RFC 3471](#), January 2003.

- [RFC3472] Ashwood-Smith, P., Berger, L. (Editors), "Generalized MPLS Signaling - CR-LDP Extensions", [RFC 3472](#), January 2003.
- [RFC3473] Berger, L. (Editor), "Generalized MPLS Signaling - RSVP-TE Extensions", [RFC 3473](#) January 2003.
- [GMPLSArch] Mannie, E. (Editor), "Generalized Multiprotocol Label Switching (GMPLS) Architecture", Internet Draft <[draft-many-gmpls-architecture-07.txt](#)>, May 2003, work in progress.
- [GMPLSSonetSDH] Mannie, E., Papadimitriou, D. (Editors), "Generalized Multi-Protocol Label Switching Extensions for SONET and SDH Control", Internet Draft <[draft-ietf-ccamp-gmpls-sonet-sdh-08.txt](#)>, February 2003, work in progress.
- [TCMIB] Nadeau, T., Cucchiara, J. (Editors) "Definitions of Textual Conventions for Multiprotocol Label Switching (MPLS) Management", Internet Draft <[draft-ietf-mpls-tc-mib-09.txt](#)>, August 2003, work in progress.
- [LSRMIB] Srinivasan, C., Viswanathan, A. and T. Nadeau, "Multiprotocol Label Switching (MPLS) Label Switching Router (LSR) Management Information Base", Internet Draft <[draft-ietf-mpls-lsr-mib-12.txt](#)>, August 2003, work in progress.
- [GMPLSTEMIB] Nadeau, T., Srinivasan, C., Farrel, A., Hall, T., and Harrison, E., "Generalized Multiprotocol Label Switching (GMPLS) Traffic Engineering Management Information Base", [draft-ietf-ccamp-gmpls-te-mib-01.txt](#), August 2003, work in progress.
- [GMPLSTCMIB] Nadeau, T., Srinivasan, C., Farrel, A., Hall, T., and Harrison, E., "Definitions of Textual Conventions for Multiprotocol Label Switching (MPLS) Management", [draft-ietf-ccamp-gmpls-te-mib-01.txt](#), August 2003, work in progress.

## **12.2. Informational References**

- [RFC2026] S. Bradner, "The Internet Standards Process -- Revision 3", [RFC 2026](#), October 1996.
- [RFC3413] Levi, D., Meyer, P., Stewart, B., "SNMP Applications", [RFC 3413](#), December 2002.

[RFC3410]

Case, J., Mundy, R., Partain, D. and B. Stewart,  
"Introduction and Applicability Statements for  
Internet-Standard Management Framework", [RFC 3410](#),  
December 2002.



- [RFC3411] Harrington, D., Presuhn, R., and B. Wijnen, "An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks", [RFC 3411](#), December 2002.
- [RFC3031] Rosen, E., Viswanathan, A., and R. Callon, "Multiprotocol Label Switching Architecture", [RFC 3031](#), January 2001.
- [RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", [RFC 3209](#), December 2001.
- [RFC3212] Jamoussi, B., Aboul-Magd, O., Andersson, L., Ashwood-Smith, P., Hellstrand, F., Sundell, K., Callon, R., Dantu, R., Wu, L., Doolan, P., Worster, T., Feldman, N., Fredette, A., Girish, M., Gray, E., Halpern, J., Heinanen, J., Kilty, T., Malis, A., and P. Vaananen, "Constraint-Based LSP Setup using LDP", [RFC 3212](#), December 2001."
- [TEMIB] Nadeau, T., Srinivasan, C, Viswanathan, A., "Multiprotocol Label Switching (MPLS) Traffic Engineering Management Information Base", Internet Draft <[draft-ietf-mpls-te-mib-12.txt](#)>, August 2003, work in progress.

### **[13. Authors' Addresses](#)**

Thomas D. Nadeau  
Cisco Systems, Inc.  
300 Apollo Drive  
Chelmsford, MA 01824  
Phone: +1-978-244-3051  
Email: [tnadeau@cisco.com](mailto:tnadeau@cisco.com)

Cheenu Srinivasan  
Bloomberg L.P.  
499 Park Ave.,  
New York, NY 10022  
Phone: +1-212-893-3682  
Email: [cheenu@bloomberg.net](mailto:cheenu@bloomberg.net)

Adrian Farrel  
Old Dog Consulting  
Phone: +44-(0)-1978-860944  
Email: [adrian@olddog.co.uk](mailto:adrian@olddog.co.uk)

Tim Hall  
Data Connection Ltd.  
100 Church Street  
Enfield, Middlesex  
EN2 6BQ, UK  
Phone: +44 20 8366 1177  
Email: [timhall@dataconnection.com](mailto:timhall@dataconnection.com)

Nadeau, Srinivasan, Farrel, Hall and Harrison

[Page 29]

Ed Harrison  
Data Connection Ltd.  
100 Church Street  
Enfield, Middlesex  
EN2 6BQ, UK  
Phone: +44 20 8366 1177  
Email: ed.harrison@dataconnection.com

#### **14. Full Copyright Statement**

Copyright (C) The Internet Society (2003). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns. This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS 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.

#### **15. Intellectual Property Notice**

The IETF takes no position regarding the validity or scope of any intellectual property 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; neither does it represent that it has made any effort to identify any such rights. Information on the IETF's procedures with respect to rights in standards-track and standards-related documentation can be found in [BCP-11](#) [[RFC2028](#)].

Copies of claims of rights made available for publication 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 Secretariat.

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 practice this standard. Please address the information to the IETF Executive Director.

## **16. Changes and Pending Work**

This section must be removed before the draft progresses to RFC.

### **16.1. Pending Work**

The following work items have been identified for this draft. They will be addressed in a future version.

- Expand conformance statements to give one for monitoring only, and one for monitoring and control.
- Extend mplsTrafficParamsTable from the MPLS-LSR-MIB to support GMPLS traffic parameters for signaled and static LSPs.
- Provide support for monitoring tunnel resources in GMPLS systems. For example, SONET/SDH or G.709. This might be done through an arbitrary RowPointer to an external MIB.
- Extend the performance tables from the MPLS-LSR-MIB for technology-specific GMPLS LSPs.
- Supply more and better examples, especially ones using the Label Table.
- Provide a next index object to supply the next available arbitrary index into the Label Table.

