

Definitions of Textual Conventions (TCs) for
Multiprotocol Label Switching (MPLS) Management

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 (2004).

Abstract

This memo defines a Management Information Base (MIB) module which contains Textual Conventions to represent commonly used Multiprotocol Label Switching (MPLS) management information. The intent is that these TEXTUAL CONVENTIONS (TCs) will be imported and used in MPLS related MIB modules that would otherwise define their own representations.

Table of Contents

1. Introduction.	2
2. The Internet-Standard Management Framework.	2
3. MPLS Textual Conventions MIB Definitions.	2
4. References.	16
4.1. Normative References.	16
4.2. Informative References.	17
5. Security Considerations	17
6. IANA Considerations	17
7. Contributors.	18
8. Acknowledgements.	19
9. Authors' Addresses.	19
10. Full Copyright Statement.	20

1. Introduction

This document defines a MIB module which contains Textual Conventions for Multiprotocol Label Switching (MPLS) networks. These Textual Conventions should be imported by MIB modules which manage MPLS networks.

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 [RFC2119].

For an introduction to the concepts of MPLS, see [RFC3031].

2. The Internet-Standard 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].

3. MPLS Textual Conventions MIB Definitions

```
MPLS-TC-STD-MIB DEFINITIONS ::= BEGIN
```

```
IMPORTS
```

```
    MODULE-IDENTITY,
    Unsigned32, Integer32,
    transmission          FROM SNMPv2-SMI          -- [RFC2578]
```

```
    TEXTUAL-CONVENTION
    FROM SNMPv2-TC;          -- [RFC2579]
```

```
mplsTCStdMIB MODULE-IDENTITY
    LAST-UPDATED "200406030000Z" -- June 3, 2004
    ORGANIZATION
        "IETF Multiprotocol Label Switching (MPLS) Working
        Group."
    CONTACT-INFO
        "          Thomas D. Nadeau
```

Cisco Systems, Inc.
tnadeau@cisco.com

Joan Cucchiara
Marconi Communications, Inc.
jcucchiara@mindspring.com

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

Arun Viswanathan
Forcel0 Networks, Inc.
arunv@forcel0networks.com

Hans Sjostrand
ipUnplugged
hans@ipunplugged.com

Kireeti Kompella
Juniper Networks
kireeti@juniper.net

Email comments to the MPLS WG Mailing List at
mpls@uu.net."

DESCRIPTION

"Copyright (C) The Internet Society (2004). The initial version of this MIB module was published in RFC 3811. For full legal notices see the RFC itself or see:
<http://www.ietf.org/copyrights/ianamib.html>

This MIB module defines TEXTUAL-CONVENTIONS for concepts used in Multiprotocol Label Switching (MPLS) networks."

REVISION "200406030000Z" -- June 3, 2004

DESCRIPTION

"Initial version published as part of RFC 3811."

::= { mplsStdMIB 1 }

mplsStdMIB OBJECT IDENTIFIER

::= { transmission 166 }

MplsAtmVcIdentifier ::= TEXTUAL-CONVENTION
DISPLAY-HINT "d"

STATUS current

DESCRIPTION

"A Label Switching Router (LSR) that creates LDP sessions on ATM interfaces uses the VCI or VPI/VCI field to hold the LDP Label.

VCI values MUST NOT be in the 0-31 range. The values 0 to 31 are reserved for other uses by the ITU and ATM Forum. The value of 32 can only be used for the Control VC, although values greater than 32 could be configured for the Control VC.

If a value from 0 to 31 is used for a VCI the management entity controlling the LDP subsystem should reject this with an inconsistentValue error. Also, if the value of 32 is used for a VC which is NOT the Control VC, this should result in an inconsistentValue error."

REFERENCE

"MPLS using LDP and ATM VC Switching, RFC3035."

SYNTAX Integer32 (32..65535)

MplsBitRate ::= TEXTUAL-CONVENTION

DISPLAY-HINT "d"

STATUS current

DESCRIPTION

"If the value of this object is greater than zero, then this represents the bandwidth of this MPLS interface (or Label Switched Path) in units of '1,000 bits per second'.

The value, when greater than zero, represents the bandwidth of this MPLS interface (rounded to the nearest 1,000) in units of 1,000 bits per second. If the bandwidth of the MPLS interface is between $((n * 1000) - 500)$ and $((n * 1000) + 499)$, the value of this object is n, such that $n > 0$.

If the value of this object is 0 (zero), this means that the traffic over this MPLS interface is considered to be best effort."

SYNTAX Unsigned32 (0|1..4294967295)

MplsBurstSize ::= TEXTUAL-CONVENTION

DISPLAY-HINT "d"

STATUS current
DESCRIPTION

"The number of octets of MPLS data that the stream may send back-to-back without concern for policing. The value of zero indicates that an implementation does not support Burst Size."

SYNTAX Unsigned32 (0..4294967295)

MplsExtendedTunnelId ::= TEXTUAL-CONVENTION

STATUS current
DESCRIPTION

"A unique identifier for an MPLS Tunnel. This may represent an IPv4 address of the ingress or egress LSR for the tunnel. This value is derived from the Extended Tunnel Id in RSVP or the Ingress Router ID for CR-LDP."

REFERENCE

"RSVP-TE: Extensions to RSVP for LSP Tunnels, [RFC3209].

Constraint-Based LSP Setup using LDP, [RFC3212]."
SYNTAX Unsigned32(0..4294967295)

MplsLabel ::= TEXTUAL-CONVENTION

STATUS current
DESCRIPTION

"This value represents an MPLS label as defined in [RFC3031], [RFC3032], [RFC3034], [RFC3035] and [RFC3471].

The label contents are specific to the label being represented, such as:

- * The label carried in an MPLS shim header (for LDP this is the Generic Label) is a 20-bit number represented by 4 octets. Bits 0-19 contain a label or a reserved label value. Bits 20-31 MUST be zero.

The following is quoted directly from [RFC3032]. There are several reserved label values:

- i. A value of 0 represents the 'IPv4 Explicit NULL Label'. This label value is only legal at the bottom of the label stack. It indicates that the label stack must be popped, and the forwarding of the packet must then be based on the

IPv4 header.

- ii. A value of 1 represents the 'Router Alert Label'. This label value is legal anywhere in the label stack except at the bottom. When a received packet contains this label value at the top of the label stack, it is delivered to a local software module for processing. The actual forwarding of the packet is determined by the label beneath it in the stack. However, if the packet is forwarded further, the Router Alert Label should be pushed back onto the label stack before forwarding. The use of this label is analogous to the use of the 'Router Alert Option' in IP packets [RFC2113]. Since this label cannot occur at the bottom of the stack, it is not associated with a particular network layer protocol.
- iii. A value of 2 represents the 'IPv6 Explicit NULL Label'. This label value is only legal at the bottom of the label stack. It indicates that the label stack must be popped, and the forwarding of the packet must then be based on the IPv6 header.
- iv. A value of 3 represents the 'Implicit NULL Label'. This is a label that an LSR may assign and distribute, but which never actually appears in the encapsulation. When an LSR would otherwise replace the label at the top of the stack with a new label, but the new label is 'Implicit NULL', the LSR will pop the stack instead of doing the replacement. Although this value may never appear in the encapsulation, it needs to be specified in the Label Distribution Protocol, so a value is reserved.
- v. Values 4-15 are reserved.

* The frame relay label can be either 10-bits or

23-bits depending on the DLCI field size and the upper 22-bits or upper 9-bits must be zero, respectively.

- * For an ATM label the lower 16-bits represents the VCI, the next 12-bits represents the VPI and the remaining bits MUST be zero.
- * The Generalized-MPLS (GMPLS) label contains a value greater than $2^{24}-1$ and used in GMPLS as defined in [RFC3471]."

REFERENCE

"Multiprotocol Label Switching Architecture, RFC3031.

MPLS Label Stack Encoding, [RFC3032].

Use of Label Switching on Frame Relay Networks, RFC3034.

MPLS using LDP and ATM VC Switching, RFC3035.
Generalized Multiprotocol Label Switching (GMPLS) Architecture, [RFC3471]."

SYNTAX Unsigned32 (0..4294967295)

MplsLabelDistributionMethod ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"The label distribution method which is also called the label advertisement mode [RFC3036]. Each interface on an LSR is configured to operate in either Downstream Unsolicited or Downstream on Demand."

REFERENCE

"Multiprotocol Label Switching Architecture, RFC3031.

LDP Specification, RFC3036, Section 2.6.3."

SYNTAX INTEGER {
 downstreamOnDemand(1),
 downstreamUnsolicited(2)
}

MplsLdpIdentifier ::= TEXTUAL-CONVENTION

DISPLAY-HINT "1d.1d.1d.1d:2d"

STATUS current

DESCRIPTION

"The LDP identifier is a six octet

quantity which is used to identify a Label Switching Router (LSR) label space.

The first four octets identify the LSR and must be a globally unique value, such as a 32-bit router ID assigned to the LSR, and the last two octets identify a specific label space within the LSR."

SYNTAX OCTET STRING (SIZE (6))

MplsLsrIdentifier ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"The Label Switching Router (LSR) identifier is the first 4 bytes of the Label Distribution Protocol (LDP) identifier."

SYNTAX OCTET STRING (SIZE (4))

MplsLdpLabelType ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"The Layer 2 label types which are defined for MPLS LDP and/or CR-LDP are generic(1), atm(2), or frameRelay(3)."

SYNTAX INTEGER {
 generic(1),
 atm(2),
 frameRelay(3)
 }

MplsLSPID ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"A unique identifier within an MPLS network that is assigned to each LSP. This is assigned at the head end of the LSP and can be used by all LSRs to identify this LSP. This value is piggybacked by the signaling protocol when this LSP is signaled within the network. This identifier can then be used at each LSR to identify which labels are being swapped to other labels for this LSP. This object can also be used to disambiguate LSPs that share the same RSVP sessions between the same source and destination.

For LSPs established using CR-LDP, the LSPID is composed of the ingress LSR Router ID (or any of its own IPv4 addresses) and a locally unique CR-LSP ID to that LSR. The first two bytes carry

the CR-LSPID, and the remaining 4 bytes carry the Router ID. The LSPID is useful in network management, in CR-LSP repair, and in using an already established CR-LSP as a hop in an ER-TLV.

For LSPs signaled using RSVP-TE, the LSP ID is defined as a 16-bit (2 byte) identifier used in the SENDER_TEMPLATE and the FILTER_SPEC that can be changed to allow a sender to share resources with itself. The length of this object should only be 2 or 6 bytes. If the length of this octet string is 2 bytes, then it must identify an RSVP-TE LSPID, or it is 6 bytes, it must contain a CR-LDP LSPID."

REFERENCE

"RSVP-TE: Extensions to RSVP for LSP Tunnels, [RFC3209].

Constraint-Based LSP Setup using LDP, [RFC3212]."

SYNTAX OCTET STRING (SIZE (2|6))

MplsLspType ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"Types of Label Switch Paths (LSPs) on a Label Switching Router (LSR) or a Label Edge Router (LER) are:

- unknown(1) -- if the LSP is not known to be one of the following.
- terminatingLsp(2) -- if the LSP terminates on the LSR/LER, then this is an egressing LSP which ends on the LSR/LER,
- originatingLsp(3) -- if the LSP originates from this LSR/LER, then this is an ingressing LSP which is the head-end of the LSP,
- crossConnectingLsp(4) -- if the LSP ingresses and egresses on the LSR, then it is cross-connecting on that

```

                                LSR."
SYNTAX INTEGER {
    unknown(1),
    terminatingLsp(2),
    originatingLsp(3),
    crossConnectingLsp(4)
}

MplsOwner ::= TEXTUAL-CONVENTION
STATUS      current
DESCRIPTION
    "This object indicates the local network
    management subsystem that originally created
    the object(s) in question.  The values of
    this enumeration are defined as follows:

    unknown(1) - the local network management
    subsystem cannot discern which
    component created the object.

    other(2) - the local network management
    subsystem is able to discern which component
    created the object, but the component is not
    listed within the following choices,
    e.g., command line interface (cli).

    snmp(3) - The Simple Network Management Protocol
    was used to configure this object initially.

    ldp(4) - The Label Distribution Protocol was
    used to configure this object initially.

    crldp(5) - The Constraint-Based Label Distribution
    Protocol was used to configure this object
    initially.

    rsvpTe(6) - The Resource Reservation Protocol was
    used to configure this object initially.

    policyAgent(7) - A policy agent (perhaps in
    combination with one of the above protocols) was
    used to configure this object initially.

    An object created by any of the above choices
    MAY be modified or destroyed by the same or a
    different choice."
SYNTAX INTEGER {
    unknown(1),

```

```

        other(2),
        snmp(3),
        ldp(4),
        crldp(5),
        rsvpTe(6),
        policyAgent(7)
    }

```

MplsPathIndexOrZero ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"A unique identifier used to identify a specific path used by a tunnel. A value of 0 (zero) means that no path is in use."

SYNTAX Unsigned32(0..4294967295)

MplsPathIndex ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"A unique value to index (by Path number) an entry in a table."

SYNTAX Unsigned32(1..4294967295)

MplsRetentionMode ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"The label retention mode which specifies whether an LSR maintains a label binding for a FEC learned from a neighbor that is not its next hop for the FEC.

If the value is conservative(1) then advertised label mappings are retained only if they will be used to forward packets, i.e., if label came from a valid next hop.

If the value is liberal(2) then all advertised label mappings are retained whether they are from a valid next hop or not."

REFERENCE

"Multiprotocol Label Switching Architecture, RFC3031.

LDP Specification, RFC3036, Section 2.6.2."

```

SYNTAX INTEGER {
    conservative(1),
    liberal(2)
}

```

```

MplsTunnelAffinity ::= TEXTUAL-CONVENTION
    STATUS          current
    DESCRIPTION
        "Describes the configured 32-bit Include-any,
         include-all, or exclude-all constraint for
         constraint-based link selection."
    REFERENCE
        "RSVP-TE: Extensions to RSVP for LSP Tunnels,
         RFC3209, Section 4.7.4."
    SYNTAX          Unsigned32(0..4294967295)

MplsTunnelIndex ::= TEXTUAL-CONVENTION
    STATUS          current
    DESCRIPTION
        "A unique index into mplsTunnelTable.
         For tunnels signaled using RSVP, this value
         should correspond to the RSVP Tunnel ID
         used for the RSVP-TE session."
    SYNTAX          Unsigned32 (0..65535)

MplsTunnelInstanceIndex ::= TEXTUAL-CONVENTION
    STATUS          current
    DESCRIPTION
        "The tunnel entry with instance index 0
         should refer to the configured tunnel
         interface (if one exists).

         Values greater than 0, but less than or
         equal to 65535, should be used to indicate
         signaled (or backup) tunnel LSP instances.
         For tunnel LSPs signaled using RSVP,
         this value should correspond to the
         RSVP LSP ID used for the RSVP-TE
         LSP.

         Values greater than 65535 apply to FRR
         detour instances."
    SYNTAX          Unsigned32(0|1..65535|65536..4294967295)

TeHopAddressType ::= TEXTUAL-CONVENTION
    STATUS          current
    DESCRIPTION
        "A value that represents a type of address for a
         Traffic Engineered (TE) Tunnel hop.

         unknown(0)    An unknown address type.  This value
                       MUST be used if the value of the
                       corresponding TeHopAddress object is a

```

zero-length string. It may also be used to indicate a TeHopAddress which is not in one of the formats defined below.

- ipv4(1) An IPv4 network address as defined by the InetAddressIPv4 TEXTUAL-CONVENTION [RFC3291].
- ipv6(2) A global IPv6 address as defined by the InetAddressIPv6 TEXTUAL-CONVENTION [RFC3291].
- asnumber(3) An Autonomous System (AS) number as defined by the TeHopAddressAS TEXTUAL-CONVENTION.
- unnum(4) An unnumbered interface index as defined by the TeHopAddressUnnum TEXTUAL-CONVENTION.
- lspid(5) An LSP ID for TE Tunnels (RFC3212) as defined by the MplsLSPID TEXTUAL-CONVENTION.

Each definition of a concrete TeHopAddressType value must be accompanied by a definition of a TEXTUAL-CONVENTION for use with that TeHopAddress.

To support future extensions, the TeHopAddressType TEXTUAL-CONVENTION SHOULD NOT be sub-typed in object type definitions. It MAY be sub-typed in compliance statements in order to require only a subset of these address types for a compliant implementation.

Implementations must ensure that TeHopAddressType objects and any dependent objects (e.g., TeHopAddress objects) are consistent. An inconsistentValue error must be generated if an attempt to change a TeHopAddressType object would, for example, lead to an undefined TeHopAddress value that is not defined herein. In particular, TeHopAddressType/TeHopAddress pairs must be changed together if the address type changes (e.g., from ipv6(2) to ipv4(1))."

REFERENCE

"TEXTUAL-CONVENTIONS for Internet Network Addresses, RFC3291.

Constraint-Based LSP Setup using LDP,
[RFC3212]"

```
SYNTAX      INTEGER {
                unknown(0),
                ipv4(1),
                ipv6(2),
                asnumber(3),
                unnum(4),
                lspid(5)
            }
```

TeHopAddress ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"Denotes a generic Tunnel hop address, that is, the address of a node which an LSP traverses, including the source and destination nodes. An address may be very concrete, for example, an IPv4 host address (i.e., with prefix length 32); if this IPv4 address is an interface address, then that particular interface must be traversed. An address may also specify an 'abstract node', for example, an IPv4 address with prefix length less than 32, in which case, the LSP can traverse any node whose address falls in that range. An address may also specify an Autonomous System (AS), in which case the LSP can traverse any node that falls within that AS.

A TeHopAddress value is always interpreted within the context of an TeHopAddressType value. Every usage of the TeHopAddress TEXTUAL-CONVENTION is required to specify the TeHopAddressType object which provides the context. It is suggested that the TeHopAddressType object is logically registered before the object(s) which use the TeHopAddress TEXTUAL-CONVENTION if they appear in the same logical row.

The value of a TeHopAddress object must always be

consistent with the value of the associated TeHopAddressType object. Attempts to set a TeHopAddress object to a value which is inconsistent with the associated TeHopAddressType must fail with an inconsistentValue error."

SYNTAX OCTET STRING (SIZE (0..32))

TeHopAddressAS ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"Represents a two or four octet AS number. The AS number is represented in network byte order (MSB first). A two-octet AS number has the two MSB octets set to zero."

REFERENCE

"Textual Conventions for Internet Network Addresses, [RFC3291]. The InetAutonomousSystemsNumber TEXTUAL-CONVENTION has a SYNTAX of Unsigned32, whereas this TC has a SYNTAX of OCTET STRING (SIZE (4)). Both TCs represent an autonomous system number but use different syntaxes to do so."

SYNTAX OCTET STRING (SIZE (4))

TeHopAddressUnnum ::= TEXTUAL-CONVENTION

STATUS current

DESCRIPTION

"Represents an unnumbered interface:

octets	contents	encoding
1-4	unnumbered interface	network-byte order

The corresponding TeHopAddressType value is unnum(5)."

SYNTAX OCTET STRING(SIZE(4))

END

4. References

4.1. Normative References

- [RFC2113] Katz, D., "IP Router Alert Option", RFC 2113, February 1997.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2434] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP: 26, RFC 2434, October 1998.
- [RFC2578] McCloghrie, K., Perkins, D., and J. Schoenwaelder, "Structure of Management Information Version 2 (SMIv2)", STD 58, RFC 2578, April 1999.
- [RFC2579] McCloghrie, K., Perkins, D., and J. Schoenwaelder, "Textual Conventions for SMIv2", STD 58, RFC 2579, April 1999.
- [RFC2580] McCloghrie, K., Perkins, D., and J. Schoenwaelder, "Conformance Statements for SMIv2", STD 58, RFC 2580, April 1999.
- [RFC3031] Rosen, E., Viswanathan, A., and R. Callon, "Multiprotocol Label Switching Architecture", RFC 3031, January 2001.
- [RFC3032] Rosen, E., Rekhter, Y., Tappan, D., Farinacci, D., Federokow, G., Li, T., and A. Conta, "MPLS Label Stack Encoding", RFC 3032, January 2001.
- [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.
- [RFC3036] Andersson, L., Doolan, P., Feldman, N., Fredette, A., and B. Thomas, "LDP Specification", RFC 3036, 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., Ed., Andersson, L., Callon, R., Dantu, R., Wu, L., Doolan, P., Worster, T., Feldman, N., Fredette, A., Girish, M., Gray, E., Heinanen, J., Kilty, T., and A. Malis, "Constraint-Based LSP Setup using LDP", RFC 3212, January 2002.
- [RFC3291] Daniele, M., Haberman, B., Routhier, S., and J. Schoenwaelder, "Textual Conventions for Internet Network Addresses", RFC 3291, May 2002.
- [RFC3471] Berger, L., Editor, "Generalized Multi-Protocol Label Switching (GMPLS) Architecture", RFC 3471, January 2003.

4.2. Informative References

- [RFC3410] Case, J., Mundy, R., Partain, D., and B. Stewart, "Introduction and Applicability Statements for Internet-Standard Management Framework", RFC 3410, December 2002.

5. Security Considerations

This module does not define any management objects. Instead, it defines a set of textual conventions which may be used by other MPLS MIB modules to define management objects.

Meaningful security considerations can only be written in the MIB modules that define management objects. Therefore, this document has no impact on the security of the Internet.

6. IANA Considerations

IANA has made a MIB OID assignment under the transmission branch, that is, assigned the `mplsStdMIB` under `{ transmission 166 }`. This sub-id is requested because 166 is the `ifType` for `mpls(166)` and is available under `transmission`.

In the future, MPLS related standards track MIB modules should be rooted under the `mplsStdMIB` subtree. The IANA is requested to manage that namespace. New assignments can only be made via a Standards Action as specified in [RFC2434].

The IANA has also assigned `{ mplsStdMIB 1 }` to the `MPLS-TC-STD-MIB` specified in this document.

7. Contributors

This document was created by combining TEXTUAL-CONVENTIONS from current MPLS MIBs and a TE-WG MIB. Co-authors on each of these MIBs contributed to the TEXTUAL-CONVENTIONS contained in this MIB and also contributed greatly to the revisions of this document. These co-authors addresses are included here because they are useful future contacts for information about this document. These co-authors are:

Cheenu Srinivasan
Bloomberg L.P.
499 Park Ave.
New York, NY 10022

Phone: +1-212-893-3682
EMail: cheenu@bloomberg.net

Arun Viswanathan
Forcel0 Networks, Inc.
1440 McCarthy Blvd
Milpitas, CA 95035

Phone: +1-408-571-3516
EMail: arunv@forcel0networks.com

Hans Sjostrand
ipUnplugged
P.O. Box 101 60
S-121 28 Stockholm, Sweden

Phone: +46-8-725-5900
EMail: hans@ipunplugged.com

Kireeti Kompella
Juniper Networks
1194 Mathilda Ave
Sunnyvale, CA 94089

Phone: +1-408-745-2000
EMail: kireeti@juniper.net

8. Acknowledgements

This document is a product of the MPLS Working Group. The editors and contributors would like to thank Mike MacFadden and Adrian Farrel for their helpful comments on several reviews. Also, the editors and contributors would like to give a special acknowledgement to Bert Wijnen for his many detailed reviews. Bert's assistance and guidance is greatly appreciated.

9. Authors' Addresses

Thomas D. Nadeau
Cisco Systems, Inc.
BXB300/2/
300 Beaver Brook Road
Boxborough, MA 01719

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

Joan E. Cucchiara
Marconi Communications, Inc.
900 Chelmsford Street
Lowell, MA 01851

Phone: +1-978-275-7400
EMail: jcucchiara@mindspring.com

10. Full Copyright Statement

Copyright (C) The Internet Society (2004). 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.

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.

