Network Working Group   Peter Ashwood-Smith (Nortel Networks Corp.)
Internet Draft   Ayan Banerjee (Calient Networks)
Expiration Date: May 2002   Lou Berger (Movaz Networks)
                                     Greg Bernstein (Ciena Corporation)
                                          John Drake (Calient Networks)
                                           Yanhe Fan (Axiowave Networks)
                             Kireeti Kompella (Juniper Networks, Inc.)
                                                   Eric Mannie (EBONE)
                                Jonathan P. Lang (Calient Networks)
                                     Bala Rajagopalan (Tellium, Inc.)
                              Yakov Rekhter (Juniper Networks, Inc.)
                                       Debanjan Saha (Tellium, Inc.)
                                     Vishal Sharma (Metanoia, Inc.)
                                     George Swallow (Cisco Systems)
                                          Z. Bo Tang (Tellium, Inc.)

                                                     November 2001

Generalized MPLS - Signaling Functional Description

draft-ietf-mpls-generalized-signaling-07.txt

Status of this Memo

Abstract

   This document describes extensions to MPLS signaling required to
   support Generalized MPLS. Generalized MPLS extends the MPLS control
   plane to encompass time-division (e.g. SONET ADMs), wavelength
   (optical lambdas) and spatial switching (e.g. incoming port or fiber
   to outgoing port or fiber). This document presents a functional
   description of the extensions. Protocol specific formats and
   mechanisms are specified in [GMPLS-RSVP] and [GMPLS-LDP]. Technology
   specific details are expected to be specified in independent
   technology specific documents, e.g., [GMPLS-SONET].

Contents

[Editor's note: changes to be removed prior to publication as an RFC.]
Changes from previous version:

o Removed various parameter values based on comments

o Minor editorial changes and clarifications

[1]. Introduction

   The Multiprotocol Label Switching (MPLS) architecture [MPLS-ARCH] has
   been defined to support the forwarding of data based on a label. In
   this architecture, Label Switching Routers (LSRs) were assumed to
   have a forwarding plane that is capable of (a) recognizing either
   packet or cell boundaries, and (b) being able to process either
   packet headers (for LSRs capable of recognizing packet boundaries) or
   cell headers (for LSRs capable of recognizing cell boundaries).

   The original architecture has recently been extended to include LSRs
   whose forwarding plane recognizes neither packet, nor cell
   boundaries, and therefore, can't forward data based on the
   information carried in either packet or cell headers. Specifically,
   such LSRs include devices where the forwarding decision is based on
   time slots, wavelengths, or physical ports.

   Given the above, LSRs, or more precisely interfaces on LSRs, can be
   subdivided into the following classes:

   1. Interfaces that recognize packet/cell boundaries and can forward
      data based on the content of the packet/cell header. Examples
      include interfaces on routers that forward data based on the
      content of the "shim" header, interfaces on ATM-LSRs that forward
      data based on the ATM VPI/VCI. Such interfaces are referred to as
      Packet-Switch Capable (PSC).

   2. Interfaces that forward data based on the data's time slot in a
      repeating cycle. An example of such an interface is an interface
      on a SONET Cross-Connect. Such interfaces are referred to as
      Time-Division Multiplex Capable (TDM).

   3. Interfaces that forward data based on the wavelength on which the
      data is received. An example of such an interface is an interface
      on an Optical Cross-Connect that can operate at the level of an
      individual wavelength. Such interfaces are referred to as Lambda
      Switch Capable (LSC).

   4. Interfaces that forward data based on a position of the data in
      the real world physical spaces. An example of such an interface
      is an interface on an Optical Cross-Connect that can operate at

Berger, et. al.   [Page 3]

the level of a single (or multiple) fibers. Such interfaces are
referred to as Fiber-Switch Capable (FSC).

Using the concept of nested LSPs allows the system to scale by
building a forwarding hierarchy. At the top of this hierarchy are
FSC interfaces, followed by LSC interfaces, followed by TDM
interfaces, followed by PSC interfaces. This way, an LSP that starts
and ends on a PSC interface can be nested (together with other LSPs)
into an LSP that starts and ends on a TDM interface. This LSP, in
turn, can be nested (together with other LSPs) into an LSP that
starts and ends on an LSC interface, which in turn can be nested
(together with other LSPs) into an LSP that starts and ends on a FSC
interface. See [MPLS-HIERARCHY] for more information on LSP
hierarchies.

The establishment of LSPs that span only the first class of
interfaces is defined in [LDP, CR-LDP, RSVP-TE]. This document
presents a functional description of the extensions needed to
generalize the MPLS control plane to support each of the four classes
of interfaces. Only signaling protocol independent formats and
definitions are provided in this document. Protocol specific formats
are defined in [GMPLS-RSVP] and [GMPLS-LDP]. Technology specific
details are outside the scope of this document and will be specified
in technology specific documents, such as [GMPLS-SONET].

2. Overview

Generalized MPLS differs from traditional MPLS in that it supports
multiple types of switching, i.e., the addition of support for TDM,
lambda, and fiber (port) switching. The support for the additional
types of switching has driven generalized MPLS to extend certain base
functions of traditional MPLS and, in some cases, to add
functionality. These changes and additions impact basic LSP
properties, how labels are requested and communicated, the
unidirectional nature of LSPs, how errors are propagated, and
information provided for synchronizing the ingress and egress.

In traditional MPLS Traffic Engineering, links traversed by an LSP
can include an intermix of links with heterogeneous label encodings.
For example, an LSP may span links between routers, links between
routers and ATM-LSRs, and links between ATM-LSRs. Generalized MPLS
extends this by including links where the label is encoded as a time
slot, or a wavelength, or a position in the real world physical
space. Just like with traditional MPLS TE, where not all LSRs are
capable of recognizing (IP) packet boundaries (e.g., an ATM-LSR) in
their forwarding plane, generalized MPLS includes support for LSRs
that can't recognize (IP) packet boundaries in their forwarding

plane. In traditional MPLS TE an LSP that carries IP has to start
and end on a router. Generalized MPLS extends this by requiring an
LSP to start and end on similar type of LSRs. Also, in generalized
MPLS the type of a payload that can be carried by an LSP is extended
to allow such payloads as SONET/SDH, or 1 or 10Gb Ethernet. These
changes from traditional MPLS are reflected in how labels are
requested and communicated in generalized MPLS, see Sections [3.1](3.1) and
3.2. A special case of Lambda switching, called Waveband switching
is also described in Section 3.3.

Another basic difference between traditional and non-PSC types of
generalized MPLS LSPs, is that bandwidth allocation for an LSP can be
performed only in discrete units, see Section 3.1.3. There are also
likely to be (much) fewer labels on non-PSC links than on PSC links.
Note that the use of Forwarding Adjacencies (FA), see [MPLS-
HIERARCHY], provides a mechanism that may improve bandwidth
utilization, when bandwidth allocation can be performed only in
discrete units, as well as a mechanism to aggregate forwarding state,
thus allowing the number of required labels to be reduced.

Generalized MPLS allows for a label to be suggested by an upstream
node, see Section 3.4. This suggestion may be overridden by a
downstream node but, in some cases, at the cost of higher LSP setup
time. The suggested label is valuable when establishing LSPs through
certain kinds of optical equipment where there may be a lengthy (in
electrical terms) delay in configuring the switching fabric. For
example micro mirrors may have to be elevated or moved, and this
physical motion and subsequent damping takes time. If the labels and
hence switching fabric are configured in the reverse direction (the
norm) the MAPPING/Resv message may need to be delayed by 10's of
milliseconds per hop in order to establish a usable forwarding path.
The suggested label is also valuable when recovering from nodal
faults.

Generalized MPLS extends on the notion of restricting the range of
labels that may be selected by a downstream node, see Section 3.5.
In generalized MPLS, an ingress or other upstream node may restrict
the labels that may be used by an LSP along either a single hop or
along the whole LSP path. This feature is driven from the optical
domain where there are cases where wavelengths used by the path must
be restricted either to a small subset of possible wavelengths, or to
one specific wavelength. This requirement occurs because some
equipment may only be able to generate a small set of the wavelengths
that intermediate equipment may be able to switch, or because
intermediate equipment may not be able to switch a wavelength at all,
being only able to redirect it to a different fiber.

While traditional traffic engineered MPLS (and even LDP) are

unidirectional, generalized MPLS supports the establishment of
bidirectional LSPs, see Section 4. The need for bidirectional LSPs
comes from non-PSC applications. There are multiple reasons why such
LSPs are needed, particularly possible resource contention when
allocating reciprocal LSPs via separate signaling sessions, and
simplifying failure restoration procedures in the non-PSC case.
Bidirectional LSPs also have the benefit of lower setup latency and
lower number of messages required during setup.

Generalized MPLS supports the communication of a specific label to
use on a specific interface, see Section 6. [GMPLS-RSVP] also
supports an RSVP specific mechanism for rapid failure notification.

Generalized MPLS formalizes possible separation of control and data
channels, see Section 9. Such support is particularly important to
support technologies where control traffic cannot be sent in-band
with the data traffic.

Generalized MPLS also allows for the inclusion of technology specific
parameters in signaling. The intent is for all technology specific
parameters to be carried, when using RSVP, in the SENDER_TSPEC and
other related objects, and when using CR-LDP, in the Traffic
Parameters TLV. Technology specific formats will be defined on an as
needed basis. For an example definition, see [GMPLS-SONET].

3. Label Related Formats

To deal with the widening scope of MPLS into the optical and time
domain, several new forms of "label" are required. These new forms
of label are collectively referred to as a "generalized label". A
generalized label contains enough information to allow the receiving
node to program its cross connect, regardless of the type of this
cross connect, such that the ingress segments of the path are
properly joined. This section defines a generalized label request, a
generalized label, support for waveband switching, suggested label
and label sets.

Note that since the nodes sending and receiving the new form of label
know what kinds of link they are using, the generalized label does
not contain a type field, instead the nodes are expected to know from
context what type of label to expect.

3.1. Generalized Label Request

   The Generalized Label Request supports communication of
   characteristics required to support the LSP being requested. These
   characteristics include LSP encoding and LSP payload. Note that
   these characteristics may be used by transit nodes, e.g., to support
   penultimate hop popping.

   The Generalized Label Request carries an LSP encoding parameter,
   called LSP Encoding Type. This parameter indicates the encoding
   type, e.g., SONET/SDH/GigE etc., that will be used with the data
   associated with the LSP. The LSP Encoding Type represents the nature
   of the LSP, and not the nature of the links that the LSP traverses.
   A link may support a set of encoding formats, where support means
   that a link is able to carry and switch a signal of one or more of
   these encoding formats depending on the resource availability and
   capacity of the link. For example, consider an LSP signaled with
   "lambda" encoding.   It is expected that such an LSP would be
   supported with no electrical conversion and no knowledge of the
   modulation and speed by the transit nodes. Other formats normally
   require framing knowledge, and field parameters are broken into the
   framing type and speed as shown below.

   The Generalized Label Request also indicates the type of switching
   that is being requested on a link. This field normally is consistent
   across all links of an LSP.

3.1.1. Required Information

   The information carried in a Generalized Label Request is:

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    | LSP Enc. Type |Switching Type |              G-PID            |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

   LSP Encoding Type: 8 bits

      Indicates the encoding of the LSP being requested. The
      following shows permitted values and their meaning:

```
Value       Type
-----       ----
  1         Packet
  2         Ethernet V2/DIX
  3         ANSI PDH
  4         ETSI PDH
  5         SDH ITU-T G.707
  6         SONET ANSI T1.105
  7         Digital Wrapper
  8         Lambda (photonic)
  9         Fiber
 10         Ethernet 802.3
 11         FiberChannel
```

The ANSI PDH and ETSI PDH types designate these respective
networking technologies. DS1 and DS3 are examples of ANSI PDH
LSPs. An E1 LSP would be ETSI PDH. The Lambda encoding type
refers to an LSP that encompasses a whole wavelengths. The
Fiber encoding type refers to an LSP that encompasses a whole
fiber port.

Switching Type: 8 bits

Indicates the type of switching that should be performed on a
particular link. This field is needed for links that advertise
more than one type of switching capability. Values of this
field are as the Switching Capability field defined in [GMPLS-
RTG]

Generalized PID (G-PID): 16 bits

An identifier of the payload carried by an LSP, i.e. an
identifier of the client layer of that LSP. This is used by
the nodes at the endpoints of the LSP, and in some cases by the
penultimate hop. Standard Ethertype values are used for packet
and Ethernet LSPs; other values are:

```
Value   Type                                Technology
-----   ----                                ----------
  0     Unknown                             All
  1     Reserved
  2     Reserved
  3     Reserved
  4     Reserved
  5     Asynchronous mapping of E4          SONET, SDH
  6     Asynchronous mapping of DS3/T3      SONET, SDH
  7     Asynchronous mapping of E3          SONET, SDH
```

```
        8      Bit synchronous mapping of E3          SDH
        9      Byte synchronous mapping of E3         SDH
       10      Asynchronous mapping of DS2/T2         SONET, SDH
       11      Bit synchronous mapping of DS2/T2      SONET, SDH
       12      Reserved
       13      Asynchronous mapping of E1             SONET, SDH
       14      Byte synchronous mapping of E1         SONET, SDH
       15      Byte synchronous mapping of 31 * DS0   SONET, SDH
       16      Asynchronous mapping of DS1/T1         SONET, SDH
       17      Bit synchronous mapping of DS1/T1      SONET, SDH
       18      Byte synchronous mapping of DS1/T1     SONET, SDH
       19      VC-11 in VC-12                         SDH
       20      Reserved
       21      Reserved
       22      DS1 SF Asynchronous                    SONET
       23      DS1 ESF Asynchronous                   SONET
       24      DS3 M23 Asynchronous                   SONET
       25      DS3 C-Bit Parity Asynchronous          SONET
       26      VT/LOVC                                SONET, SDH
       27      STS SPE/HOVC                           SONET, SDH
       28      POS - No Scrambling, 16 bit CRC        SONET, SDH
       29      POS - No Scrambling, 32 bit CRC        SONET, SDH
       30      POS - Scrambling, 16 bit CRC           SONET, SDH
       31      POS - Scrambling, 32 bit CRC           SONET, SDH
       32      ATM mapping                            SONET, SDH
       33      Ethernet                               SDH, Lambda, Fiber
       34      SDH                                    Lambda, Fiber
       35      SONET                                  Lambda, Fiber
       36      Digital Wrapper                        Lambda, Fiber
       37      Lambda                                 Fiber
       38      ETSI PDH                               SDH
       39      ANSI PDH                               SONET, SDH
       40      Link Access Protocol SDH               SONET, SDH
               (LAPS - X.85 and X.86)
       41      FDDI                                   SONET, SDH, Lambda,
Fiber
       42      DQDB (ETSI ETS 300 216)                SONET, SDH
       43      FiberChannel-3 (Services)              FiberChannel
```

3.1.2. Bandwidth Encoding

   Bandwidth encodings are carried in in 32 bit number in IEEE floating
   point format (the unit is bytes per second). For non-packet LSPs, it
   is useful to define discrete values to identify the bandwidth of the
   LSP. Some typical values for the requested bandwidth are enumerated
   below. (These values are guidelines.) Additional values will be
   defined as needed. Bandwidth encoding values are carried in a per

protocol specific manner, see [GMPLS-RSVP] and [GMPLS-LDP].

Berger, et. al.   [Page 9]

Signal Type	(Bit-rate)	Value (Bytes/Sec) (IEEE Floating point)
DS0	(0.064 Mbps)	0x45FA0000
DS1	(1.544 Mbps)	0x483C7A00
E1	(2.048 Mbps)	0x487A0000
DS2	(6.312 Mbps)	0x4940A080
E2	(8.448 Mbps)	0x4980E800
Ethernet	(10.00 Mbps)	0x49989680
E3	(34.368 Mbps)	0x4A831A80
DS3	(44.736 Mbps)	0x4AAAA780
STS-1	(51.84 Mbps)	0x4AC5C100
Fast Ethernet	(100.00 Mbps)	0x4B3EBC20
E4	(139.264 Mbps)	0x4B84D000
FC-0 133M		0x4B7DAD68
OC-3/STM-1	(155.52 Mbps)	0x4B9450C0
FC-0 266M		0x4BFDAD68
FC-0 531M		0x4C7D3356
OC-12/STM-4	(622.08 Mbps)	0x4C9450C0
GigE	(1000.00 Mbps)	0x4CEE6B28
FC-0 1062M		0x4CFD3356
OC-48/STM-16	(2488.32 Mbps)	0x4D9450C0
OC-192/STM-64	(9953.28 Mbps)	0x4E9450C0
10GigE-LAN	(10000.00 Mbps)	0x4E9502F9
OC-768/STM-256	(39813.12 Mbps)	0x4F9450C0

3.2. Generalized Label

The Generalized Label extends the traditional label by allowing the
representation of not only labels which travel in-band with
associated data packets, but also labels which identify time-slots,
wavelengths, or space division multiplexed positions. For example,
the Generalized Label may carry a label that represents (a) a single
fiber in a bundle, (b) a single waveband within fiber, (c) a single
wavelength within a waveband (or fiber), or (d) a set of time-slots
within a wavelength (or fiber). It may also carry a label that
represents a generic MPLS label, a Frame Relay label, or an ATM label
(VCI/VPI).

A Generalized Label does not identify the "class" to which the label
belongs. This is implicit in the multiplexing capabilities of the
link on which the label is used.

A Generalized Label only carries a single level of label, i.e., it is
non-hierarchical. When multiple levels of label (LSPs within LSPs)
are required, each LSP must be established separately, see [MPLS-

HIERARCHY].

Each Generalized Label object carries a variable length label
parameter.

3.2.1. Required Information

The information carried in a Generalized Label is:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Label                             |
|                              ...                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Label: Variable

Carries label information. The interpretation of this field
depends on the type of the link over which the label is used.

3.2.1.1. Port and Wavelength Labels

Some configurations of fiber switching (FSC) and lambda switching
(LSC) use multiple data channels/links controlled by a single control
channel. In such cases the label indicates the data channel/link to
be used for the LSP. Note that this case is not the same as when
[MPLS-BUNDLING] is being used.

The information carried in a Port and Wavelength label is:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Label                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

     Label: 32 bits

          Indicates port/fiber or lambda to be used, from the sender's
          perspective. Values used in this field only have significance
          between two neighbors, and the receiver may need to convert the
          received value into a value that has local significance.
          Values may be configured or dynamically determined using a
          protocol such as [[LMP](LMP)].

[3.2.1.2](3.2.1.2). Other Labels

   Generic MPLS labels and Frame Relay labels are encoded right
   justified aligned in 32 bits (4 octets). ATM labels are encoded with
   the VPI right justified in bits 0-15 and the VCI right justified in
   bits 16-31.

[3.3](3.3). Waveband Switching

   A special case of lambda switching is waveband switching. A waveband
   represents a set of contiguous wavelengths which can be switched
   together to a new waveband. For optimization reasons it may be
   desirable for an optical cross connect to optically switch multiple
   wavelengths as a unit. This may reduce the distortion on the
   individual wavelengths and may allow tighter separation of the
   individual wavelengths. The Waveband Label is defined to support
   this special case.

   Waveband switching naturally introduces another level of label
   hierarchy and as such the waveband is treated the same way all other
   upper layer labels are treated.

   As far as the MPLS protocols are concerned there is little difference
   between a waveband label and a wavelength label except that
   semantically the waveband can be subdivided into wavelengths whereas
   the wavelength can only be subdivided into time or statistically
   multiplexed labels.

[3.3.1](3.3.1). Required information

   Waveband switching uses the same format as the generalized label, see
   section 3.2.1.

   In the context of waveband switching, the generalized label has the
   following format:
```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                         Waveband Id                           |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                         Start Label                           |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                          End Label                            |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

    Waveband Id: 32 bits

       A waveband identifier. The value is selected by the sender and
       reused in all subsequent related messages.

    Start Label: 32 bits

       Indicates the channel identifier, from the sender's
       perspective, of the lowest value wavelength making up the
       waveband.

    End Label: 32 bits

       Indicates the channel identifier, from the sender's
       perspective, of the highest value wavelength making up the
       waveband.

   Channel identifiers are established either by configuration or by
   means of a protocol such as LMP [LMP]. They are normally used in the
   label parameter of the Generalized Label one PSC and LSC.

3.4. Suggested Label

   The Suggested Label is used to provide a downstream node with the
   upstream node's label preference. This permits the upstream node to
   start configuring it's hardware with the proposed label before the
   label is communicated by the downstream node. Such early
   configuration is valuable to systems that take non-trivial time to
   establish a label in hardware. Such early configuration can reduce
   setup latency, and may be important for restoration purposes where
   alternate LSPs may need to be rapidly established as a result of
   network failures.

   The use of Suggested Label is only an optimization. If a downstream
   node passes a different label upstream, an upstream LSR reconfigures

itself so that it uses the label specified by the downstream node,
thereby maintaining the downstream control of a label. Note, the
transmission of a suggested label does not imply that the suggested
label is available for use. In particular, an ingress node should
not transmit data traffic on a suggested label until the downstream
node passes a label upstream.

The information carried in a suggested label is identical to a
generalized label.

3.5. Label Set

The Label Set is used to limit the label choices of a downstream node
to a set of acceptable labels. This limitation applies on a per hop
basis.

We describe four cases where a Label Set is useful in the optical
domain. The first case is where the end equipment is only capable of
transmitting on a small specific set of wavelengths/bands. The
second case is where there is a sequence of interfaces which cannot
support wavelength conversion (CI-incapable) and require the same
wavelength be used end-to-end over a sequence of hops, or even an
entire path. The third case is where it is desirable to limit the
amount of wavelength conversion being performed to reduce the
distortion on the optical signals. The last case is where two ends
of a link support different sets of wavelengths.

Label Set is used to restrict label ranges that may be used for a
particular LSP between two peers. The receiver of a Label Set must
restrict its choice of labels to one which is in the Label Set. Much
like a label, a Label Set may be present across multiple hops. In
this case each node generates it's own outgoing Label Set, possibly
based on the incoming Label Set and the node's hardware capabilities.
This case is expected to be the norm for nodes with conversion
incapable (CI-incapable) interfaces.

The use of Label Set is optional, if not present, all labels from the
valid label range may be used. Conceptually the absence of a Label
Set implies a Label Set whose value is {U}, the set of all valid
labels.

[3.5.1](3.5.1). Required Information

   A label set is composed of one or more Label_Set objects/TLVs. Each
   object/TLV contains one or more elements of the Label Set. Each
   element is referred to as a subchannel identifier and has the same
   format as a generalized label.

   The information carried in a Label_Set is:

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |    Action     |     Reserved      |        Label Type         |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                        Subchannel 1                           |
   |                            ...                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   :                            :                                 :
   :                            :                                 :
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                        Subchannel N                           |
   |                            ...                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
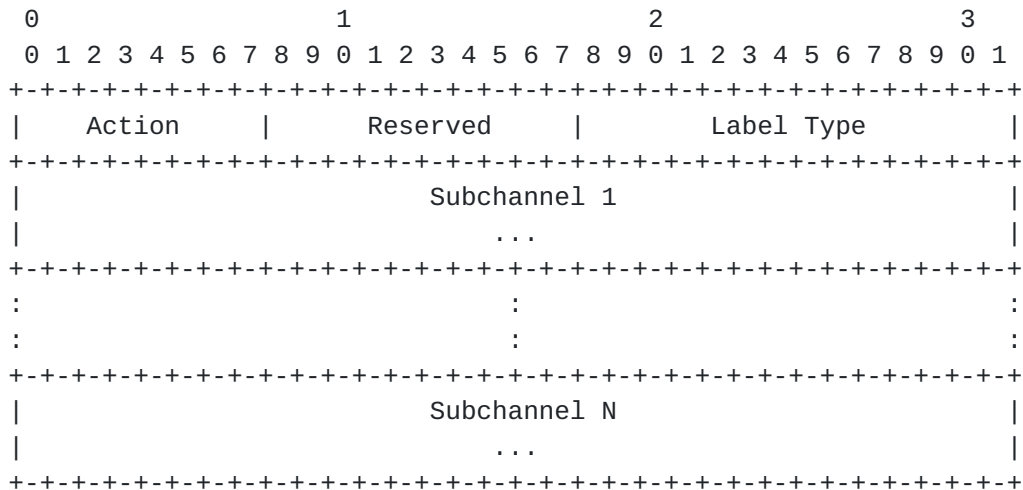
   Action: 8 bits

      0 - Inclusive List

         Indicates that the object/TLV contains one or more
         subchannel elements that are included in the Label Set.

      1 - Exclusive List

         Indicates that the object/TLV contains one or more
         subchannel elements that are excluded from the Label Set.

      2 - Inclusive Range

         Indicates that the object/TLV contains a range of labels.
         The object/TLV contains two subchannel elements. The first
         element indicates the start of the range. The second
         element indicates the end of the range. A value of zero
         indicates that there is no bound on the corresponding
         portion of the range.

3 - Exclusive Range

Indicates that the object/TLV contains a range of labels
that are excluded from the Label Set. The object/TLV
contains two subchannel elements. The first element
indicates the start of the range. The second element
indicates the end of the range. A value of zero indicates
that there is no bound on the corresponding portion of the
range.

Reserved: 10 bits

This field is reserved. It MUST be set to zero on transmission
and MUST be ignored on receipt.

Label Type: 14 bits

Indicates the type and format of the labels carried in the
object/TLV. Values are signaling protocol specific.

Subchannel:

The subchannel represents the label (wavelength, fiber ...)
which is eligible for allocation. This field has the same
format as described for labels under section 3.2.

Note that subchannel to local channel identifiers (e.g.,
wavelength) mappings are a local matter.

4. Bidirectional LSPs

This section defines direct support of bidirectional LSPs. Support
is defined for LSPs that have the same traffic engineering
requirements including fate sharing, protection and restoration,
LSRs, and resource requirements (e.g., latency and jitter) in each
direction. In the remainder of this section, the term "initiator" is
used to refer to a node that starts the establishment of an LSP and
the term "terminator" is used to refer to the node that is the target
of the LSP. Note that for bidirectional LSPs, there is only one
"initiator" and one "terminator".

Normally to establish a bidirectional LSP when using [RSVP-TE] or
[CR-LDP] two unidirectional paths must be independently established.
This approach has the following disadvantages:

* The latency to establish the bidirectional LSP is equal to one
  round trip signaling time plus one initiator-terminator signaling

   transit delay. This not only extends the setup latency for
   successful LSP establishment, but it extends the worst-case
   latency for discovering an unsuccessful LSP to as much as two
   times the initiator-terminator transit delay. These delays are
   particularly significant for LSPs that are established for
   restoration purposes.

 * The control overhead is twice that of a unidirectional LSP.
   This is because separate control messages (e.g. Path and Resv)
   must be generated for both segments of the bidirectional LSP.

 * Because the resources are established in separate segments,
   route selection is complicated. There is also additional
   potential race for conditions in assignment of resources, which
   decreases the overall probability of successfully establishing
   the bidirectional connection.

 * It is more difficult to provide a clean interface for SONET
   equipment that may rely on bidirectional hop-by-hop paths for
   protection switching.

 * Bidirectional optical LSPs (or lightpaths) are seen as a
   requirement for many optical networking service providers.

   With bidirectional LSPs both the downstream and upstream data paths,
   i.e., from initiator to terminator and terminator to initiator, are
   established using a single set of signaling messages. This reduces
   the setup latency to essentially one initiator-terminator round trip
   time plus processing time, and limits the control overhead to the
   same number of messages as a unidirectional LSP.

[4.1](4.1). Required Information

   For bidirectional LSPs, two labels must be allocated. Bidirectional
   LSP setup is indicated by the presence of an Upstream Label
   object/TLV in the appropriate signaling message. An Upstream Label
   has the same format as the generalized label, see Section 3.2.

[4.2](4.2). Contention Resolution

   Contention for labels may occur between two bidirectional LSP setup
   requests traveling in opposite directions. This contention occurs
   when both sides allocate the same resources (labels) at effectively
   the same time. If there is no restriction on the labels that can be
   used for bidirectional LSPs and if there are alternate resources,
   then both nodes will pass different labels upstream and there is no

contention. However, if there is a restriction on the labels that
can be used for the bidirectional LSPs (for example, if they must be
physically coupled on a single I/O card), or if there are no more
resources available, then the contention must be resolved by other
means. To resolve contention, the node with the higher node ID will
win the contention and it MUST issue a PathErr/NOTIFICATION message
with a "Routing problem/Label allocation failure" indication. Upon
receipt of such an error, the node SHOULD try to allocate a different
Upstream label (and a different Suggested Label if used) to the
bidirectional path. However, if no other resources are available,
the node must proceed with standard error handling.

To reduce the probability of contention, one may impose a policy that
the node with the lower ID never suggests a label in the downstream
direction and always accepts a Suggested Label from an upstream node
with a higher ID. Furthermore, since the labels may be exchanged
using LMP, an alternative local policy could further be imposed such
that (with respect to the higher numbered node's label set) the
higher numbered node could allocate labels from the high end of the
label range while the lower numbered node allocates labels from the
low end of the label range. This mechanism would augment any close
packing algorithms that may be used for bandwidth (or wavelength)
optimization. One special case that should be noted when using RSVP
and supporting this approach is that the neighbor's node ID might not
be known when sending an initial Path message. When this case
occurs, a node should suggest a label chosen at random from the
available label space.

An example of contention between two nodes (PXC 1 and PXC 2) is shown
in Figure 1. In this example PXC 1 assigns an Upstream Label for the
channel corresponding to local BCId=2 (local BCId=7 on PXC 2) and
sends a Suggested Label for the channel corresponding to local BCId=1
(local BCId=6 on PXC 2). Simultaneously, PXC 2 assigns an Upstream
Label for the channel corresponding to its local BCId=6 (local BCId=1
on PXC 1) and sends a Suggested Label for the channel corresponding
to its local BCId=7 (local BCId=2 on PXC 1). If there is no
restriction on the labels that can be used for bidirectional LSPs and
if there are alternate resources available, then both PXC 1 and PXC 2
will pass different labels upstream and the contention is resolved
naturally (see Fig. 2). However, if there is a restriction on the
labels that can be used for bidirectional LSPs (for example, if they
must be physically coupled on a single I/O card), then the contention
must be resolved using the node ID (see Fig. 3).
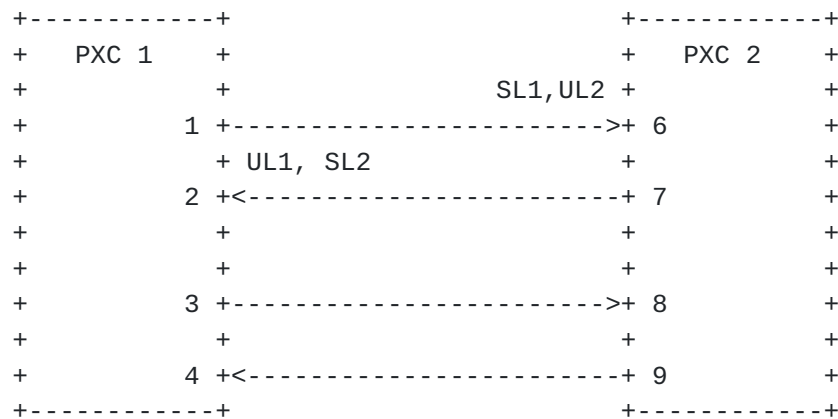
```
        +------------+                     +------------+
        +   PXC 1    +                     +   PXC 2    +
        +           +              SL1,UL2 +           +
        +         1 +------------------------>+ 6        +
        +           + UL1, SL2               +           +
        +         2 +<------------------------+ 7        +
        +           +                         +           +
        +           +                         +           +
        +         3 +------------------------>+ 8        +
        +           +                         +           +
        +         4 +<------------------------+ 9        +
        +------------+                     +------------+
                   Figure 1.  Label Contention
```

In this example, PXC 1 assigns an Upstream Label using BCId=2 (BCId=7
on PXC 2) and a Suggested Label using BCId=1 (BCId=6 on PXC 2).
Simultaneously, PXC 2 assigns an Upstream Label using BCId=6 (BCId=1
on PXC 1) and a Suggested Label using BCId=7 (BCId=2 on PXC 1).

```
        +------------+                     +------------+
        +   PXC 1    +                     +   PXC 2    +
        +           +                  UL2 +           +
        +         1 +------------------------>+ 6        +
        +           + UL1                     +           +
        +         2 +<------------------------+ 7        +
        +           +                         +           +
        +           +                      L1 +           +
        +         3 +------------------------>+ 8        +
        +           + L2                      +           +
        +         4 +<------------------------+ 9        +
        +------------+                     +------------+
  Figure 2. Label Contention Resolution without resource restrictions
```

In this example, there is no restriction on the labels that can be
used by the bidirectional connection and there is no contention.

```
          +------------+                           +------------+
          +   PXC 1    +                           +   PXC 2    +
          +            +                     UL2 +              +
          +          1 +------------------------>+ 6            +
          +            + L2                        +            +
          +          2 +<------------------------+ 7            +
          +            +                          +             +
          +            +                     L1 +              +
          +          3 +------------------------>+ 8            +
          +            + UL1                       +            +
          +          4 +<------------------------+ 9            +
          +------------+                           +------------+
```
     Figure 3. Label Contention Resolution with resource restrictions

   In this example, labels 1,2 and 3,4 on PXC 1 (labels 6,7 and 8,9 on
   PXC 2, respectively) must be used by the same bidirectional
   connection. Since PXC 2 has a higher node ID, it wins the contention
   and PXC 1 must use a different set of labels.

[5](5). **Notification on Label Error**

   There are cases in traditional MPLS and in GMPLS that result in an
   error message containing an "Unacceptable label value" indication,
   see [RSVP-TE], [GMPLS-LDP] and [GMPLS-RSVP]. When these cases occur,
   it can useful for the node generating the error message to indicate
   which labels would be acceptable. To cover this case, GMPLS
   introduces the ability to convey such information via the "Acceptable
   Label Set". An Acceptable Label Set is carried in appropriate
   protocol specific error messages, see [GMPLS-LDP] and [GMPLS-RSVP].

   The format of an Acceptable Label Set is identical to a Label Set,
   see section 3.5.1.

[6](6). **Explicit Label Control**

   In traditional MPLS, the interfaces used by an LSP may be controlled
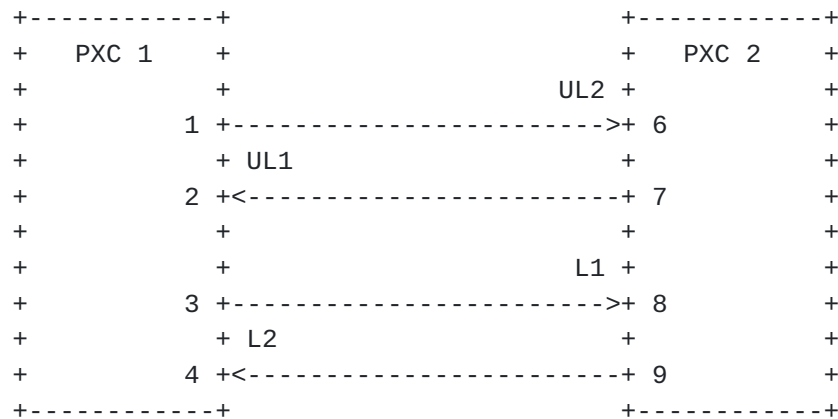   via an explicit route, i.e., ERO or ER-Hop. This enables the
   inclusion of a particular node/interface, and the termination of an
   LSP on a particular outgoing interface of the egress LSR. Where the
   interface may be numbered or unnumbered, see [MPLS-UNNUM].

   There are cases where the existing explicit route semantics do not
   provide enough information to control the LSP to the degree desired.
   This occurs in the case when the LSP initiator wishes to select a
   label used on a link. Specifically, the problem is that ERO and ER-
   Hop do not support explicit label sub-objects. An example case where

   such a mechanism is desirable is where there are two LSPs to be
   "spliced" together, i.e., where the tail of the first LSP would be
   "spliced" into the head of the second LSP. This last case is more
   likely to be used in the non-PSC classes of links.

   To cover this case, the Label ERO subobject / ER Hop is introduced.

6.1. Required Information

   The Label Explicit and Record Routes contains:

      L: 1 bit

         This bit must be set to 0.

      U: 1 bit

         This bit indicates the direction of the label. It is 0 for the
         downstream label. It is set to 1 for the upstream label and is
         only used on bidirectional LSPs.

      Label: Variable

         This field identifies the label to be used. The format of this
         field is identical to the one used by the Label field in
         Generalized Label, see Section 3.2.1.

   Placement and ordering of these parameters are signaling protocol
   specific.

7. Protection Information

   Protection Information is carried in a new object/TLV. It is used to
   indicate link related protection attributes of a requested LSP. The
   use of Protection Information for a particular LSP is optional.
   Protection Information currently indicates the link protection type
   desired for the LSP. If a particular protection type, i.e., 1+1, or
   1:N, is requested, then a connection request is processed only if the
   desired protection type can be honored. Note that the protection
   capabilities of a link may be advertised in routing, see [GMPLS-RTG].
   Path computation algorithms may take this information into account
   when computing paths for setting up LSPs.

   Protection Information also indicates if the LSP is a primary or
   secondary LSP. A secondary LSP is a backup to a primary LSP. The

resources of a secondary LSP are not used until the primary LSP
fails. The resources allocated for a secondary LSP MAY be used by
other LSPs until the primary LSP fails over to the secondary LSP. At
that point, any LSP that is using the resources for the secondary LSP
MUST be preempted.

7.1. Required Information

The following information is carried in Protection Information:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|S|                  Reserved                    | Link Flags|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Secondary (S): 1 bit

   When set, indicates that the requested LSP is a secondary LSP.

Reserved: 25 bits

   This field is reserved. It MUST be set to zero on transmission
   and MUST be ignored on receipt.

Link Flags: 6 bits

   Indicates desired link protection type. As previously
   mentioned, protection capabilities of a link may be advertised
   in routing. A value of 0 implies that any, including no, link
   protection may be used. More than one bit may be set to
   indicate when multiple protection types are acceptable. When
   multiple bits are set and multiple protection types are
   available, the choice of protection type is a local (policy)
   decision.

   The following flags are defined:

      0x20 Enhanced

          Indicates that a protection scheme that is more reliable
          than Dedicated 1+1 should be used, e.g., 4 fiber BLSR/MS-
          SPRING.

        0x10 Dedicated 1+1

           Indicates that a dedicated link layer protection scheme,
           i.e., 1+1 protection, should be used to support the LSP.

        0x08 Dedicated 1:1

           Indicates that a dedicated link layer protection scheme,
           i.e., 1:1 protection, should be used to support the LSP.

        0x04 Shared

           Indicates that a shared link layer protection scheme,
           such as 1:N protection, should be used to support the
           LSP.

        0x02 Unprotected

           Indicates that the LSP should not use any link layer
           protection.

        0x01 Extra Traffic

           Indicates that the LSP should use links that are
           protecting other (primary) traffic. Such LSPs may be
           preempted when the links carrying the (primary) traffic
           being protected fail.

8. Administrative Status Information

   Administrative Status Information is carried in a new object/TLV.
   Administrative Status Information is currently used in two ways. In
   the first, the information indicates administrative state with
   respect to a particular LSP. In this usage, Administrative Status
   Information indicates the state of the LSP. State indications
   include "up" or "down", if it in a "testing" mode, and if deletion is
   in progress. The actions taken by a node based on a state local
   decision. An example action that may be taken is to inhibit alarm
   reporting when an LSP is in "down" or "testing" states, or to report
   alarms associated with the connection at a priority equal to or less
   than "Non service affecting".

   In the second usage of Administrative Status Information, the
   information indicates a request to set an LSP's administrative state.
   This information is always relayed to the ingress node which acts on
   the request.

   The different usages are distinguished in a protocol specific
   fashion. See [[GMPLS-RSVP](#)] and [[GMPLS-LDP](#)] for details. The use of
   Administrative Status Information for a particular LSP is optional.

[8.1](#). Required Information

   The following information is carried in Administrative Status
   Information:

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |R|                     Reserved                         |T|A|D|
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

   Reflect (R): 1 bit

      When set, indicates that the edge node SHOULD reflect the
      object back in the appropriate message. This bit MUST NOT be
      set in state change request, i.e. Notify, messages.

   Reserved: 28 bits

      This field is reserved. It MUST be set to zero on transmission
      and MUST be ignored on receipt.

   Testing (T): 1 bit

      When set, indicates that the local actions related to the
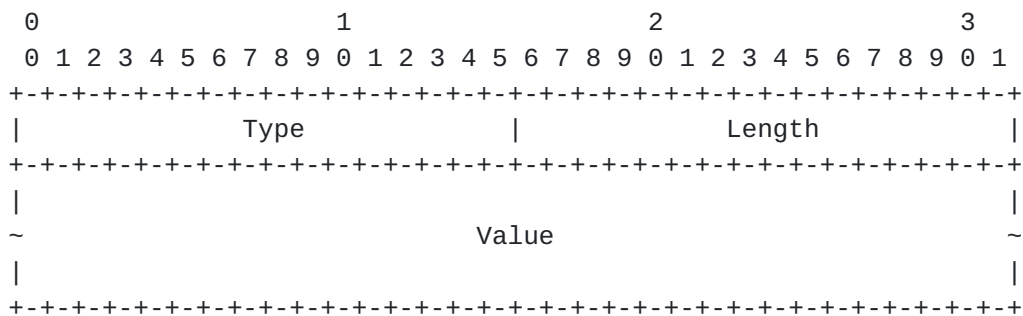   "testing" mode should be taken.

   Administratively down (A): 1 bit

      When set, indicates that the local actions related to the
      "administratively down" state should be taken.

   Deletion in progress (D): 1 bit

      When set, indicates that that the local actions related to LSP
      teardown should be taken. Edge nodes may use this flag to
      control connection teardown.

9. Control Channel Separation

   The concept of a control channel being different than a data channel
   being signaled was introduced to MPLS in connection with link
   bundling, see [MPLS-BUNDLING]. In GMPLS, the separation of control
   and data channel may be due to any number of factors. (Including
   bundling and other cases such as data channels that cannot carry in-
   band control information.) This section will cover the two critical
   related issues: the identification of data channels in signaling and
   handling of control channel failures that don't impact data channels.

9.1. Interface Identification

   In traditional MPLS there as an implicit one-to-one association of a
   control channel to a data channel. When such an association is
   present, no additional or special information is required to
   associate a particular LSP setup transaction with a particular data
   channel. (It's implicit in the control channel over which the
   signaling messages are sent.)

   In cases where there is not an explicit one-to-one association of
   control channels to data channels it is necessary to convey
   additional information in signaling to identify the particular data
   channel being controlled. GMPLS supports explicit data channel
   identification by providing interface identification information.
   GMPLS allows the use of a number of interface identification schemes
   including IPv4 or IPv6 addresses, interface indexes (see [MPLS-
   UNNUM]) and component interfaces (established via configuration or a
   protocol such as [LMP]). In all cases the choice of the data
   interface is indicated by the upstream node using addresses and
   identifiers used by the upstream node.

9.1.1. Required Information

   The following information is carried in Interface_ID:

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   ~                            TLVs                               ~
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Where each TLV has the following format:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|              Type             |             Length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
~                             Value                             ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Length: 16 bits

   Indicates the total length of the TLV, i.e., 4 + the length of
   the value field in octets. A value field whose length is not a
   multiple of four MUST be zero-padded so that the TLV is four-
   octet aligned.

Type: 16 bits

   Indicates type of interface being identified. Defined values
   are:

```
Type  Length  Format       Description
--------------------------------------------------------------------
 1       8    IPv4 Addr.    IPv4
 2      20    IPv6 Addr.    IPv6
 3      12    See below     IF_INDEX                (Interface Index)
 4      12    See below     COMPONENT_IF_DOWNSTREAM (Component interface)
 5      12    See below     COMPONENT_IF_UPSTREAM   (Component interface)
```

   For types 3, 4 and 5 the Value field has the format:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           IP Address                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Interface ID                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
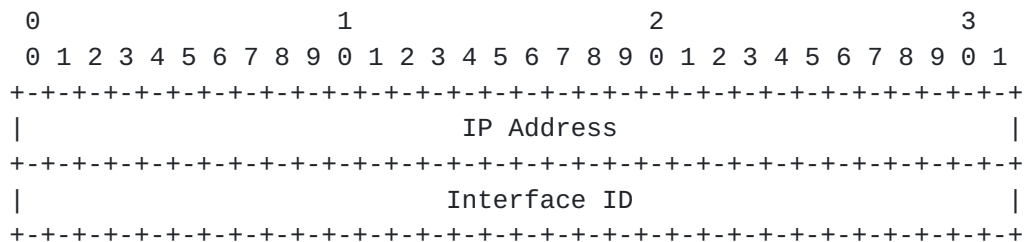
IP Address: 32 bits

The IP address field may carry either an IP address of a
link, or an IP address associated with the router ID,
where router ID may be the value carried in the router ID
TLV of routing. See [MPLS-UNNUM] for details related to
type 3 usage.

Interface ID: 32 bits

For type 3 usage, the Interface ID carries an interface
identifier as defined in [MPLS-UNNUM].

For types 4 and 5, the Interface ID indicates a bundled
component link, see [MPLS-BUNDLE]. The special value
0xFFFFFFFF can be used to indicate the same label is to
be valid across all component links.

9.2. Fault Handling

There are two new faults that must be handled when the control
channel is independent of the data channel. In the first, there is a
link or other type of failure the limits the ability of neighboring
nodes to pass control messages. In this situation, neighboring nodes
are unable to exchange control messages for a period of time. Once
communication is restored the underlying signaling protocol must
indicate that the nodes have maintained their state through the
failure. The signaling protocol must also ensure that any state
changes that were instantiated during the failure are synchronized
between the nodes.

In the the second, a node's control plane fails and then restarts and
losses most of it's state information. In this case, both upstream
and downstream nodes must synchronize their state information with
the restarted node. In order for any resynchronization to occur the
node undergoing the restart will need to preserve some information,
such as it's mappings of incoming to outgoing labels.

Both cases are addressed in protocol specific fashions, see [GMPLS-
RSVP] and [GMPLS-LDP].

Note that these cases only apply when there are mechanisms to detect
data channel failures independent of control channel failures.

10. Acknowledgments

   This draft is the work of numerous authors and consists of a
   composition of a number of previous drafts in this area. A list of
   the drafts from which material and ideas were incorporated follows:

   draft-saha-rsvp-optical-signaling-00.txt
   draft-lang-mpls-rsvp-oxc-00.txt
   draft-kompella-mpls-optical-00.txt
   draft-fan-mpls-lambda-signaling-00.txt

   Valuable comments and input were received from a number of people,
   including Igor Bryskin, Adrian Farrel, Ben Mack-Crane, Dimitri
   Papadimitriou, Fong Liaw and Juergen Heiles. Some sections of this
   document are based on text proposed by Fong Liaw.

11. Security Considerations

   This draft introduce no new security considerations to either [CR-
   LDP] or [RSVP-TE].

12. IANA Considerations

   There are multiple fields and values defined within this document.
   IANA is requested to administer assignment of new values. All values
   should be allocated through an IETF Consensus action.

13. References

[CR-LDP] Jamoussi et al., "Constraint-Based LSP Setup using LDP",
        draft-ietf-mpls-cr-ldp-05.txt, February, 2001.

[GMPLS-LDP] Ashwood-Smith, P. et al, "Generalized MPLS Signaling -
           CR-LDP Extensions", Internet Draft,
           draft-ietf-mpls-generalized-cr-ldp-05.txt,
           November, 2001.

[GMPLS-RSVP] Ashwood-Smith, P. et al, "Generalized MPLS Signaling -
            RSVP-TE Extensions", Internet Draft,
            draft-ietf-mpls-generalized-rsvp-te-06.txt,
            November, 2001.

[GMPLS-RTG] Kompella, K., et al, "Routing Extensions in Support
            of Generalized MPLS", Internet Draft,
            draft-ietf-ccamp-gmpls-routing-00.txt, September, 2001.

[GMPLS-SONET] Ashwood-Smith, P. et al, "GMPLS - SONET / SDH Specifics",
              Internet Draft, draft-ietf-ccamp-gmpls-sonet-sdh-00.txt,
              April, 2001.

[LDP] Andersson et al., "LDP Specification", RFC 3036.

[LMP] Lang, et al. "Link Management Protocol",
      Internet Draft, draft-ietf-mpls-lmp-02.txt, March, 2001.

[MPLS-ARCH] Rosen et al., "Multiprotocol label switching
            Architecture", RFC 3031.

[MPLS-BUNDLE] Kompella, K., Rekhter, Y., and Berger, L., "Link Bundling
              in MPLS Traffic Engineering", Internet Draft,
              draft-kompella-mpls-bundle-05.txt, Feb., 2001.

[MPLS-HIERARCHY] Kompella, K., and Rekhter, Y., "LSP Hierarchy with
                 MPLS TE", Internet Draft,
                 draft-ietf-mpls-lsp-hierarchy-02.txt, Feb., 2001.

[MPLS-UNNUM] Kompella, K., Rekhter, Y., "Signalling Unnumbered Links
              in RSVP-TE", Internet Draft,
              draft-ietf-mpls-rsvp-unnum-01.txt, February 2001

[RSVP-TE] Awduche, et. al., "RSVP-TE: Extensions to RSVP for LSP
          Tunnels," Internet Draft,
          draft-ietf-mpls-rsvp-lsp-tunnel-08.txt, Feb., 2001.

[RECOVERY] Sharma, et al "A Framework for MPLS-based Recovery,"
            draft-ieft-mpls-recovery-frmwrk-02.txt, March, 2001

14. Authors' Addresses

   Peter Ashwood-Smith
   Nortel Networks Corp.
   P.O. Box 3511 Station C,
   Ottawa, ON K1Y 4H7
   Canada
   Phone: +1 613 763 4534
   Email: petera@nortelnetworks.com

Ayan Banerjee
Calient Networks
5853 Rue Ferrari
San Jose, CA 95138
Phone: +1 408 972-3645
Email: abanerjee@calient.net

Lou Berger
Movaz Networks, Inc.
7926 Jones Branch Drive
Suite 615
McLean VA, 22102
Phone: +1 703 847-1801
Email: lberger@movaz.com

Greg Bernstein
Ciena Corporation
10480 Ridgeview Court
Cupertino, CA 94014
Phone: +1 408 366 4713
Email: greg@ciena.com

John Drake
Calient Networks
5853 Rue Ferrari
San Jose, CA 95138
Phone: +1 408 972 3720
Email: jdrake@calient.net

Yanhe Fan
Axiowave Networks, Inc.
200 Nickerson Road
Marlborough, MA 01752
Phone: + 1 774 348 4627
Email: yfan@axiowave.com

Kireeti Kompella
Juniper Networks, Inc.
1194 N. Mathilda Ave.
Sunnyvale, CA 94089
Email: kireeti@juniper.net

Jonathan P. Lang
Calient Networks
25 Castilian
Goleta, CA 93117
Email: jplang@calient.net

Eric Mannie
EBONE
Terhulpsesteenweg 6A
1560 Hoeilaart - Belgium
Phone: +32 2 658 56 52
Mobile: +32 496 58 56 52
Fax: +32 2 658 51 18
Email: eric.mannie@ebone.com

Bala Rajagopalan
Tellium, Inc.
2 Crescent Place
P.O. Box 901
Oceanport, NJ 07757-0901
Phone: +1 732 923 4237
Fax: +1 732 923 9804
Email: braja@tellium.com

Yakov Rekhter
Juniper Networks, Inc.
Email: yakov@juniper.net

Debanjan Saha
Tellium Optical Systems
2 Crescent Place
Oceanport, NJ 07757-0901
Phone: +1 732 923 4264
Fax: +1 732 923 9804
Email: dsaha@tellium.com

Vishal Sharma
Metanoia, Inc.
335 Elan Village Lane, Unit 203
San Jose, CA 95134-2539
Phone: +1 408-943-1794
Email: v.sharma@ieee.org

George Swallow
Cisco Systems, Inc.
250 Apollo Drive
Chelmsford, MA 01824
Voice: +1 978 244 8143
Email: swallow@cisco.com

   Z. Bo Tang
   Tellium, Inc.
   2 Crescent Place
   P.O. Box 901
   Oceanport, NJ 07757-0901
   Phone: +1 732 923 4231
   Fax: +1 732 923 9804
   Email: btang@tellium.com

Generated on: Wed Nov 21 15:29:32 EST 2001