          Transmission of IPv6 Packets over Frame Relay Networks
                            Specification

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

Abstract

   This memo describes mechanisms for the transmission of IPv6 packets
   over Frame Relay networks.

Table of Contents

1. Introduction

This document specifies the frame format for transmission of IPv6
packets over Frame Relay networks, the method of forming IPv6 link-
local addresses on Frame Relay links, and the mapping of the IPv6
addresses to Frame Relay addresses. It also specifies the content of
the Source/Target link-layer address option used in Neighbor
Discovery [ND] and Inverse Neighbor Discovery [IND] messages when
those messages are transmitted over a Frame Relay link. It is part
of a set of specifications that define such IPv6 mechanisms for Non
Broadcast Multi Access (NBMA) media [IPv6-NBMA], [IPv6-ATM], and a
larger set that defines such mechanisms for specific link layers
[IPv6-ETH], [IPv6-FDDI], [IPv6-PPP], [IPv6-ATM], etc...

The information in this document applies to Frame Relay devices which
serve as end stations (DTEs) on a public or private Frame Relay
network (for example, provided by a common carrier or PTT.) Frame
Relay end stations can be IPv6 hosts or routers. In this document
they are referred to as nodes.

In a Frame Relay network, a number of virtual circuits form the
connections between the attached stations (nodes). The resulting set
of interconnected devices forms a private Frame Relay group which may
be either fully interconnected with a complete "mesh" of virtual
circuits, or only partially interconnected. In either case, each
virtual circuit is uniquely identified at each Frame Relay interface
(card) by a Data Link Connection Identifier (DLCI). In most
circumstances, DLCIs have strictly local significance at each Frame
Relay interface.

A Frame Relay virtual circuit acts like a virtual-link (also referred
to as logical-link), with its own link parameters, distinct from the
parameters of other virtual circuits established on the same wire or
fiber. Such parameters are the input/output maximum frame size,
incoming/outgoing requested/agreed throughput, incoming/outgoing
acceptable throughput, incoming/outgoing burst size,
incoming/outgoing frame rate.

By default a DLCI is 10 bits in length. Frame Relay specifications
define also 16, 17, or 23 bit DLCIs. The former is not used, while
the latter two are suggested for use with SVCs.

Frame Relay virtual circuits can be created administratively as
Permanent Virtual Circuits -- PVCs -- or dynamically as Switched
Virtual Circuits -- SVCs. The mechanisms defined in this document
are intended to apply to both permanent and switched Frame Relay
virtual circuits, whether they are point to point or point to multi-
point.

The keywords MUST, MUST NOT, MAY, OPTIONAL, REQUIRED, RECOMMENDED,
SHALL, SHALL NOT, SHOULD, SHOULD NOT are to be interpreted as defined
in [RFC 2119].

2. Maximum Transmission Unit

The IPv6 minimum MTU is defined in [IPv6].

In general, Frame Relay devices are configured to have a maximum
frame size of at least 1600 octets. Therefore, the default IPv6 MTU
size for a Frame Relay interface is considered to be 1592.

A smaller than default frame size can be configured but of course not
smaller than the minimum IPv6 MTU.

An adequate larger than default IPv6 MTU and Frame Relay frame size
can be configured to avoid fragmentation. The maximum frame size is
controlled by the CRC generation mechanisms employed at the HDLC
level. CRC16 will protect frames up to 4096 bytes in length, which
reduces the effective maximum frame size to approximately 4088 bytes.
A larger desired frame size (such as that used by FDDI or Token
Ring), would require the CRC32 mechanism, which is not yet widely
used and is not mandatory for frame relay systems conforming to Frame
Relay Forum and ITU-T standards.

In general, if upper layers provide adequate error
protection/detection mechanisms, implementations may allow
configuring a Frame Relay link with a larger than 4080 octets frame
size but with a lesser error protection/detection mechanism at link
layer. However, because IPv6 relies on the upper and lower layer
error detection, configuring the IPv6 MTU to a value larger than 4080
is strongly discouraged.

Although a Frame Relay circuit allows the definition of distinct
maximum frame sizes for input and output, for simplification
purposes, this specification assumes symmetry, i.e. the same MTU for
both input and output.

Furthermore, implementations may limit the setting of the Frame Relay
maximum frame size to the interface (link, or card) level, which then
is enforced on all of the PVCs or SVCs on that interface (on that
link, or card). For an SVC, the maximum frame size parameter
negotiated during circuit setup will not exceed the configured
maximum frame size.

3. IPv6 Frame Format

   The IPv6 frame encapsulation for Frame Relay (for both PVCs and SVCs)
   follows [ENCAPS], which allows a VC to carry IPv6 packets along with
   other protocol packets. The NLPID frame format is used, in which the
   IPv6 NLPID has a value of 0x8E:

```
                      0                    1           (Octets)
                      +--------------------+--------------------+
         (Octets)0    |                    |                    |
                      /             Q.922 Address              /
                      /          (length 'n' equals 2 or 4)    /
                      |                    |                    |
                      +--------------------+--------------------+
              n    | Control (UI)  0x03 |    NLPID   0x8E    | NLPID
                      +--------------------+--------------------+ indicating
            n+2    |                    .                    | IPv6
                      /                    .                    /
                      /             IPv6 packet               /
                      |                    .                    |
                      +--------------------+--------------------+
                      |                    |                    |
                      +                 FCS                +
                      |                    |                    |
                      +--------------------+--------------------+
```
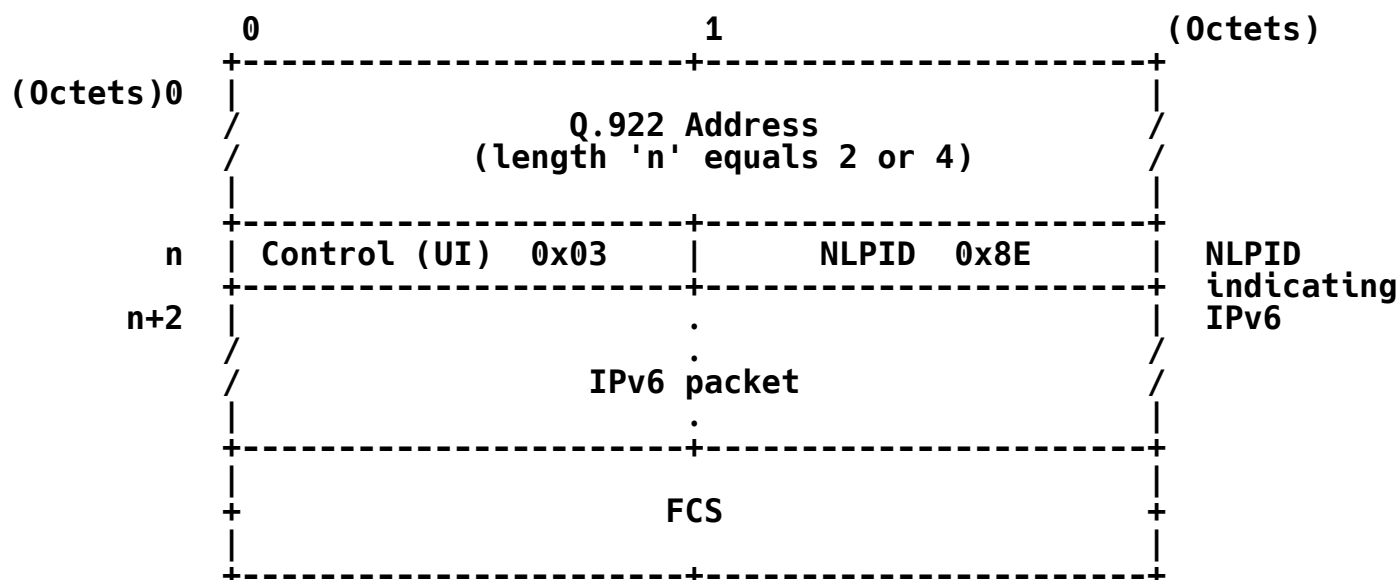
      "n" is the length of the Q.922 address which can be 2 or 4 octets.

      The Q.922 representation of a DLCI (in canonical order - the first
      bit is stored in the least significant, i.e., the right-most bit
      of a byte in memory) [CANON] is the following:

```
                7     6     5     4     3     2     1     0    (bit order)
               +-----+-----+-----+-----+-----+-----+-----+-----+
   (octet) 0   |          DLCI(high order)     |  0  |  0  |
               +-----+-----+-----+-----+-----+-----+-----+-----+
          1   |   DLCI(low order)  |  0  |  0  |  0  |  1  |
               +-----+-----+-----+-----+-----+-----+-----+-----+
```

         10 bits DLCI

```
              7     6     5     4     3     2     1     0   (bit order)
           +-----+-----+-----+-----+-----+-----+-----+-----+
(octet) 0  |            DLCI(high order)          |  0  |  0  |
           +-----+-----+-----+-----+-----+-----+-----+-----+
        1  | DLCI                    |  0  |  0  |  0  |  0  |
           +-----+-----+-----+-----+-----+-----+-----+-----+
        2  |            DLCI(low order)                 |  0  |
           +-----+-----+-----+-----+-----+-----+-----+-----+
        3  |        unused (set to 0)          |  1  |  1  |
           +-----+-----+-----+-----+-----+-----+-----+-----+
```

        17 bits DLCI

```
              7     6     5     4     3     2     1     0   (bit order)
           +-----+-----+-----+-----+-----+-----+-----+-----+
(octet) 0  |            DLCI(high order)          |  0  |  0  |
           +-----+-----+-----+-----+-----+-----+-----+-----+
        1  | DLCI                    |  0  |  0  |  0  |  0  |
           +-----+-----+-----+-----+-----+-----+-----+-----+
        2  |            DLCI                            |  0  |
           +-----+-----+-----+-----+-----+-----+-----+-----+
        3  |        DLCI (low order)          |  0  |  1  |
           +-----+-----+-----+-----+-----+-----+-----+-----+
```
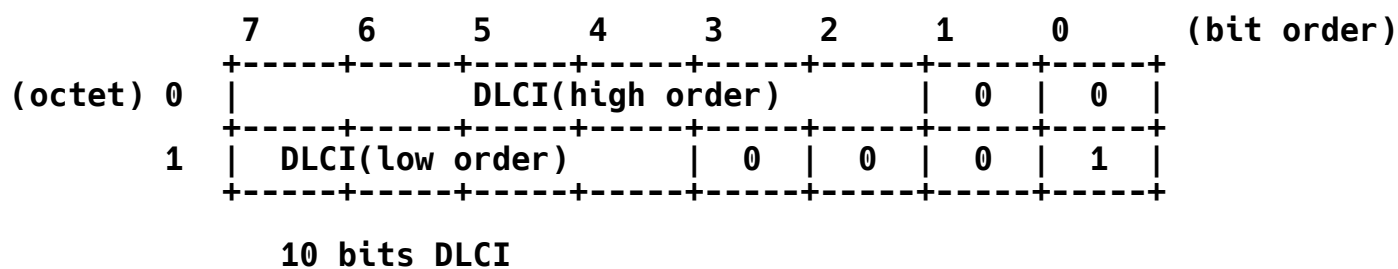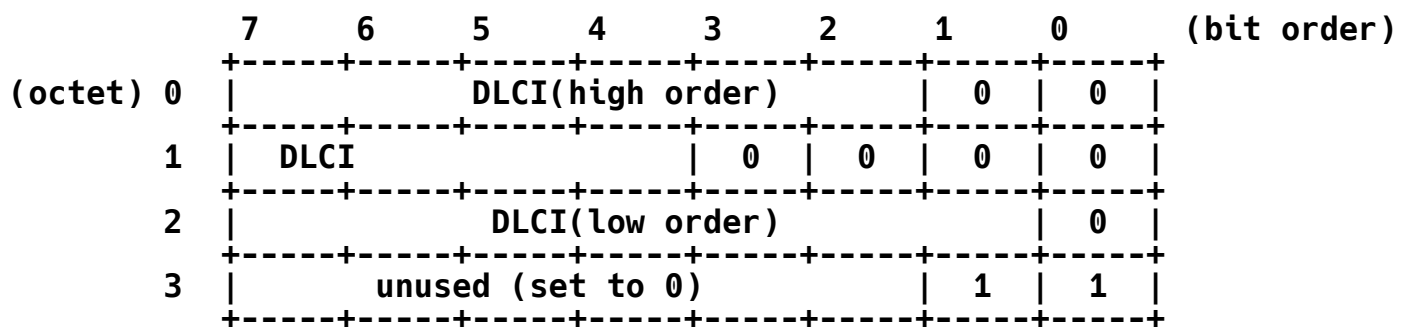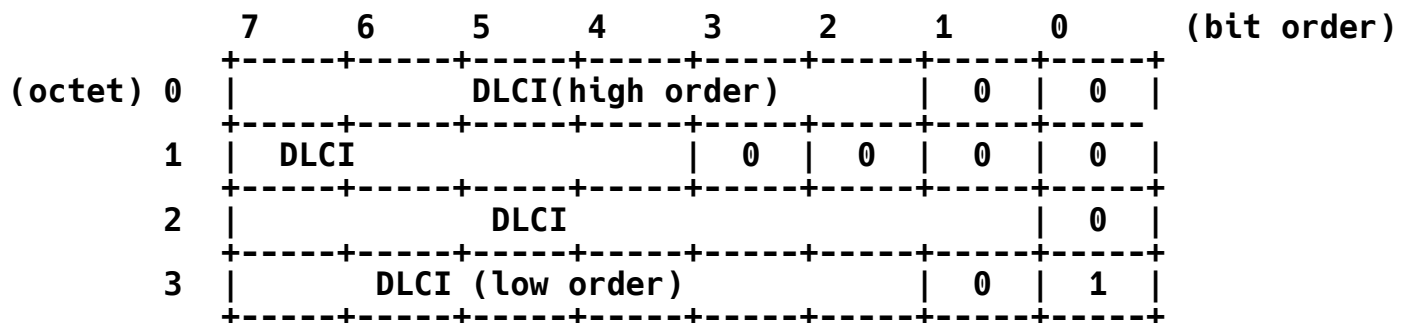
        23 bits DLCI

The encapsulation of data or control messages exchanged by various
protocols that use SNAP encapsulation (with their own PIDs) is not
affected. The encoding of the IPv6 protocol identifier in such
messages MUST be done according to the specifications of those
protocols, and [ASSNUM].

4. Stateless Autoconfiguration

An interface identifier [AARCH] for an IPv6 Frame Relay interface
must be unique on a Frame Relay link [AARCH], and must be unique on
each of the virtual links represented by the VCs terminated on the
interface.

The interface identifier for the Frame Relay interface is locally
generated by the IPv6 module.

Each virtual circuit in a Frame Relay network is uniquely identified
on a Frame Relay interface by a DLCI. Furthermore, a DLCI can be seen
as an identification of the end point of a virtual circuit on a Frame
Relay interface. Since each Frame Relay VC is configured or
established separately, and acts like an independent virtual-link
from other VCs in the network, or on the interface, link, wire or

fiber, it seems beneficial to view each VC's termination point on the Frame Relay interface as a "pseudo-interface" or "logical-interface" overlaid on the Frame Relay interface. Furthermore, it seems beneficial to be able to generate and associate an IPv6 autoconfigured address (including an IPv6 link local address) to each "pseudo-interface", i.e. end-point of a VC, i.e. to each DLCI on a Frame Relay interface.

In order to achieve the benefits described above, the mechanisms specified in this document suggest constructing the Frame Relay interface identifier from 3 distinct fields (Fig.1):

(a)   The "EUI bits" field. Bits 6 and 7 of the first octet, representing the EUI-64 "universal/local" and respectively "individual/group" bits converted to IPv6 use. The former is set to zero to reflect that the 64 bit interface identifier value has local significance [AARCH]. The latter is set to 0 to reflect the unicast address [AARCH].

(b)   The "Mid" field. A 38 bit field which is generated with the purpose of adding uniqueness to the interface identifier.

(c)   The "DLCI" field. A 24 bit field that MAY hold a 10, 17, or 23 bit DLCI value which MUST be extended with 0's to 24 bits. A DLCI based interface identifier -- which contains a valid DLCI -- SHOULD be generated as a result of successfully establishing a VC -- PVC or SVC.

      If a DLCI is not known, the field MUST be set to the "unspecified DLCI" value which consists of setting each of the 24 bits to 1.

Since DLCIs are local to a Frame Relay node, it is possible to have Frame Relay distinct virtual circuits within a Frame Relay network identified with the same DLCI values.

```
                  7     6     5     4     3     2     1     0  (bit order)
                +-----+-----+-----+-----+-----+-----+-----+-----+
 (Octets) 0     |                                   |"EUI bits" |
                +                                   +-----+-----+
          1     |                                               |
                +                                               +
          2     |                   "Mid"                       |
                +                                               +
          3     |                                               |
                +                                               +
          4     |                                               |
                +-----+-----+-----+-----+-----+-----+-----+-----+
          5     |                                               |
                +                                               +
          6     |                   "DLCI"                      |
                +                                               +
          7     |                                               |
                +-----+-----+-----+-----+-----+-----+-----+-----+
```
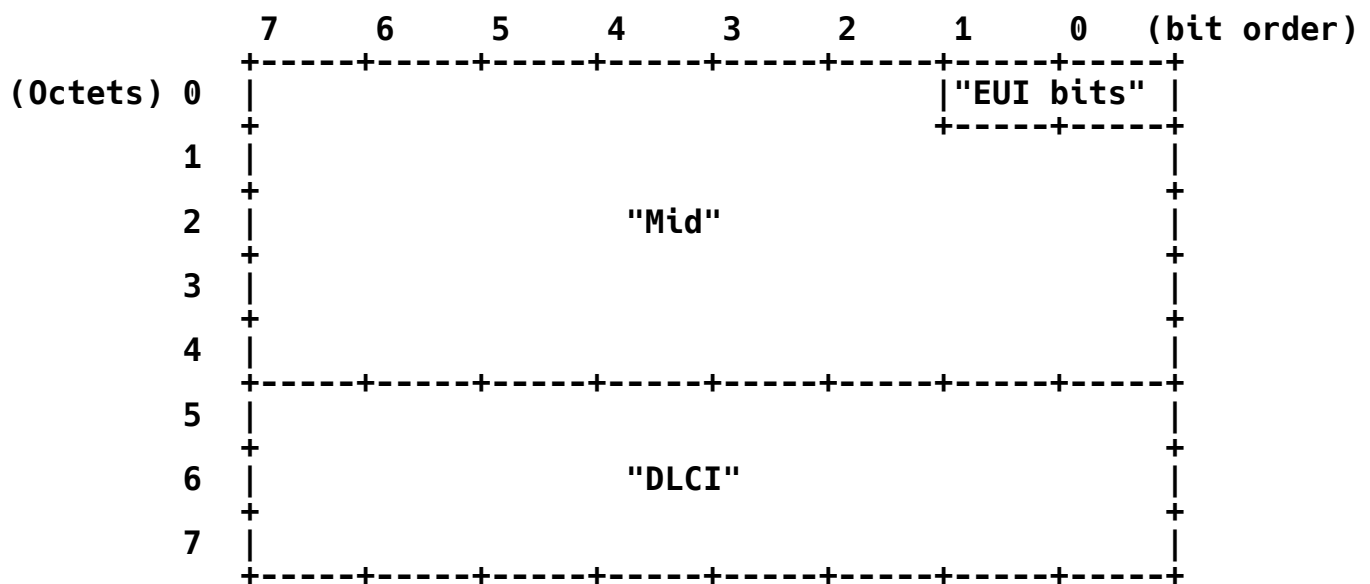
           Fig.1 Frame Relay Pseudo-Interface Identifier

   The Duplicate Address Detection specified in [AUTOCONF] is used
   repeatedly during the interface identifier and local-link address
   generation process, until the generated identifier and consequently
   the link-local address on the link -- VC -- are unique.

4.1 Generating the "Mid" field.

   The "Mid" can be generated in multiple ways. This specification
   suggests two mechanisms:

 (b.1) "Use of Local Administrative Numbers"

       The "Mid" is filled with the result of merging:

   (b.1.1) A random number of 6 bits in length (Fig.2).

   (b.1.2) The Frame Relay Node Identifier -- 16 bits -- is a user
            administered value used to locally identify a Frame Relay
            node (Fig.2).

   (b.1.3) The Frame Relay Link Identifier -- 16 bits -- is a numerical
            representation of the Frame Relay interface or link (Fig.2).

```
                7      6     5     4     3     2     1     0  (bit order)
                +-----+-----+-----+-----+-----+-----+-----+-----+
   (Octets) 0  |          Random Number              |    MBZ    |
                +-----------------------------------------+-----+-----+
            1  |                                                     |
                +            Frame Relay Node Identifier             +
            2  |                                                     |
                +-----+-----+-----+-----+-----+-----+-----+-----+
            3  |                                                     |
                +            Frame Relay Link Identifier             +
            4  |                                                     |
                +-----+-----+-----+-----+-----+-----+-----+-----+
            5  |                                                     |
                +                                                     +
            6  |                     "DLCI"                           |
                +                                                     +
            7  |                                                     |
                +-----+-----+-----+-----+-----+-----+-----+-----+
```
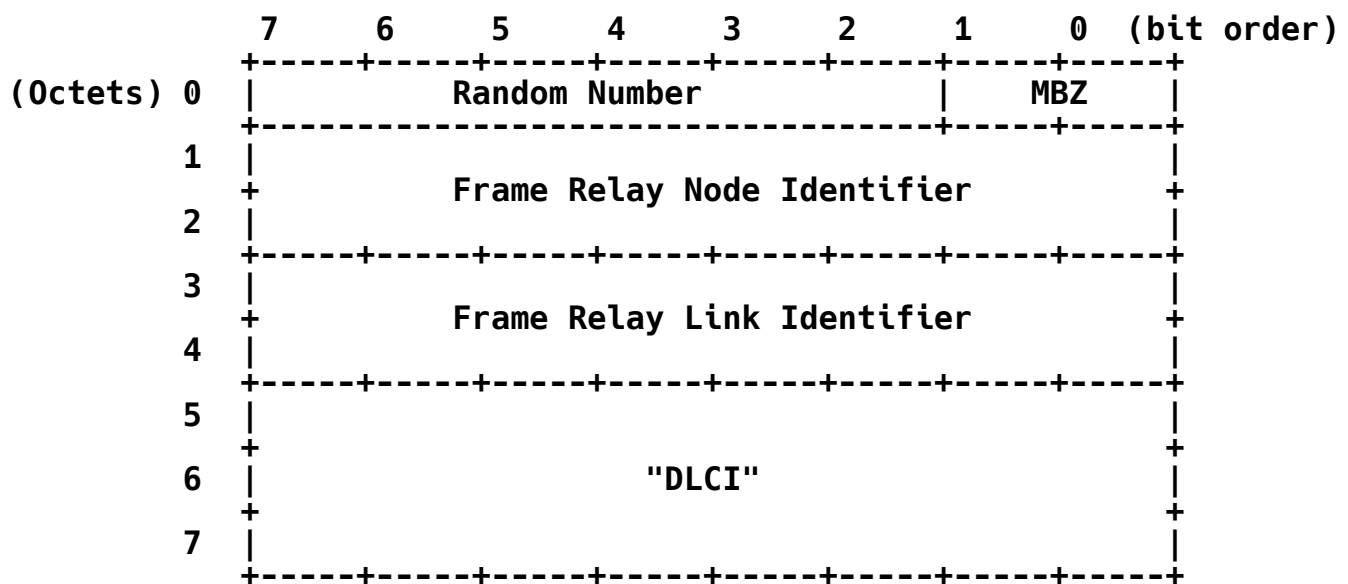
             Fig.2   Frame Relay Pseudo-Interface Identifier

   or,

   (b.2) "Use of The Frame Relay address - E.164 [E164], X.121
          [X25] numbers, or NSAP [NSAP] address"

          If a Frame Relay interface has an E.164 or a X.121 number, or an
          NSAP address, the "Mid" field MUST be filled in with a number
          resulted from it as follows: the number represented by the BCD
          encoding of the E.164 or X.121 number, or the binary encoding of
          the NSAP address is truncated to 38 bits (Fig.3). Since the Frame
          Relay interface identifier has a "local" significance, the use of
          such a value has no real practical purposes other than adding to
          the uniqueness of the interface identifier on the link. Therefore
          the truncation can be performed on the high order or low order
          bits. If the high order bits truncation does not provide
          uniqueness on the link -- perhaps the DLCI value is not unique --
          this most likely means that the VC spans more for instance than a
          national and/or international destination area for an E.164
          number, and therefore the truncation of the low order bits should
          be performed next, which most likely will provide the desired
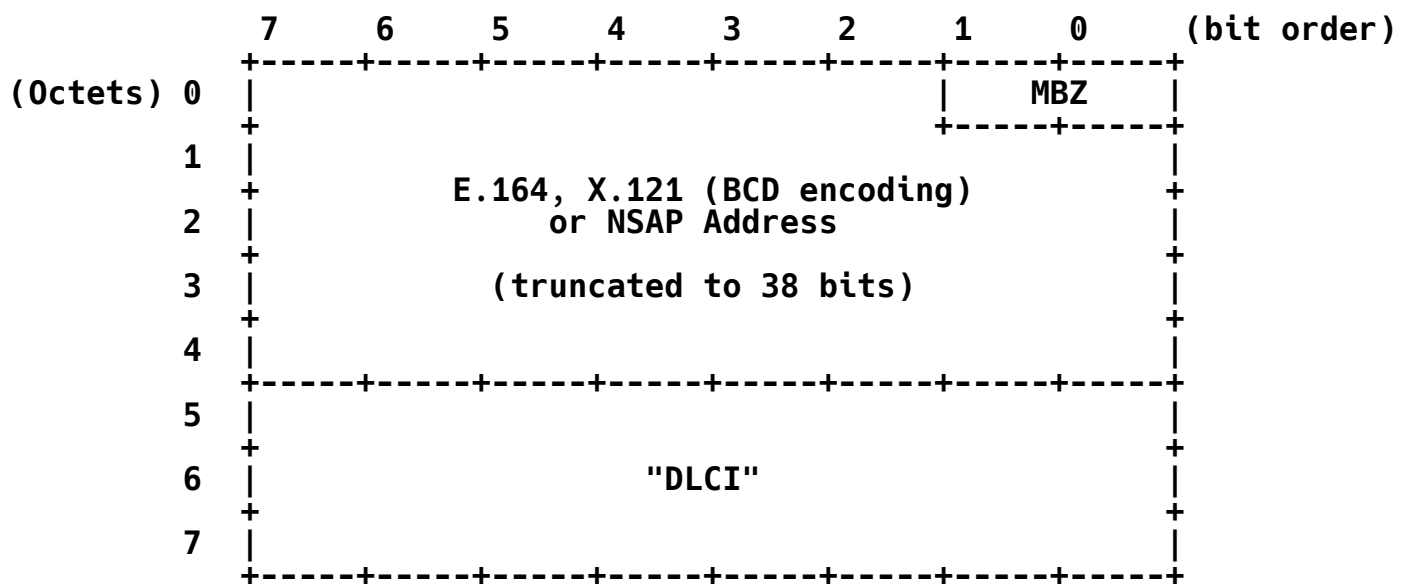   uniqueness.

```
                     7     6     5     4     3     2     1     0   (bit order)
                  +-----+-----+-----+-----+-----+-----+-----+-----+
     (Octets) 0  |                                   |     MBZ   |
                 +                                    +-----+-----+
             1  |                                                 |
                 +            E.164, X.121 (BCD encoding)         +
             2  |                 or NSAP Address                 |
                 +                                                 +
             3  |            (truncated to 38 bits)               |
                 +                                                 +
             4  |                                                 |
                 +-----+-----+-----+-----+-----+-----+-----+-----+
             5  |                                                 |
                 +                                                 +
             6  |                     "DLCI"                       |
                 +                                                 +
             7  |                                                 |
                 +-----+-----+-----+-----+-----+-----+-----+-----+
```

            Fig.3   Frame Relay (Pseudo) Interface Identifier

5. Link-Local Addresses

The IPv6 link-local address [AARCH] for an IPv6 Frame Relay interface
is formed by appending the interface identifier, formed as defined
above, to the prefix FE80::/64 [AARCH].

```
      10 bits            54 bits                    64 bits
   +----------+---------------------+---------------------------+
   |1111111010|       (zeros)       |Frame Relay Interface Ident.|
   +----------+---------------------+---------------------------+
```
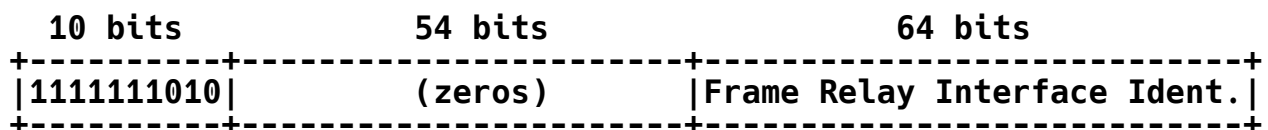
6. Address Mapping -- Unicast, Multicast

The procedure for mapping IPv6 addresses to link-layer addresses is
described in [IPv6-ND]. Additionally, extensions to Neighbor
Discovery (ND) that allow the mapping of link-layer addresses to IPv6
addresses are defined as Inverse Neighbor Discovery (IND) in [IND].
This document defines the formats of the link-layer address fields
used by ND and IND. This specification does not define an algorithmic
mapping of IPv6 multicast addresses to Frame Relay link-layer
addresses.

The Source/Target Link-layer Address option used in Neighbor
Discovery and Inverse Neighbor Discovery messages for a Frame Relay
link follows the general rules defined by [IPv6-ND]. IPv6 addresses
can map two type of identifiers equivalent to link-layer addresses:

   DLCIs, and Frame Relay Addresses. Therefore, for Frame Relay, this
   document defines two distinct formats for the ND and IND messages
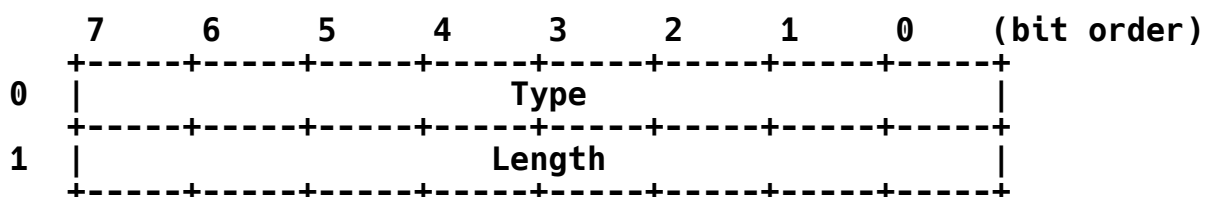   Link-Layer Address field:

   (a) DLCI Format -- used in ND and/or IND messages on VCs that were
        established prior to the ND or IND message exchange -- mostly
        PVCs. The use on SVCs makes sense with Inverse Neighbor
        Discovery [IND] messages if IND is employed after the successful
        establishing of an SVC to gather information about other IPv6
        addresses assigned to the remote node and that SVC.

   (b) Frame Relay Address Format -- used mostly prior to establishing
        a new SVC, to get the Frame Relay remote node identifier
        (link-layer address) mapping to a certain IPv6 address.

        Note: An implementation may hold both types of link layer
        identifiers in the Neighbor Discovery cache. Additionally, in
        case of multiple VCs between two nodes, one node's Neighbor
        Discovery cache may hold a mapping of one of the remote node's
        IPv6 addresses to each and every DLCI identifying the VCs.

        The mechanisms which in such an implementation would make the
        distinction between the Neighbor Discovery Cache mapping of an
        IPv6 address to a "Frame Relay Address Format" and a "DLCI
        Format" link-layer address, or among several mappings to a "DLCI
        Format" addresses are beyond the scope of this specification.

        The use of the override "O" bit in the advertisement messages
        that contain the above Link-Layer Address formats SHOULD be
        consistent with the [ND] specifications. Additionally, there
        should be consistency related to the type of Link-Layer Address
        format: an implementation should override one address format in
        its Neighbor Discovery cache with the same type of address
        format.
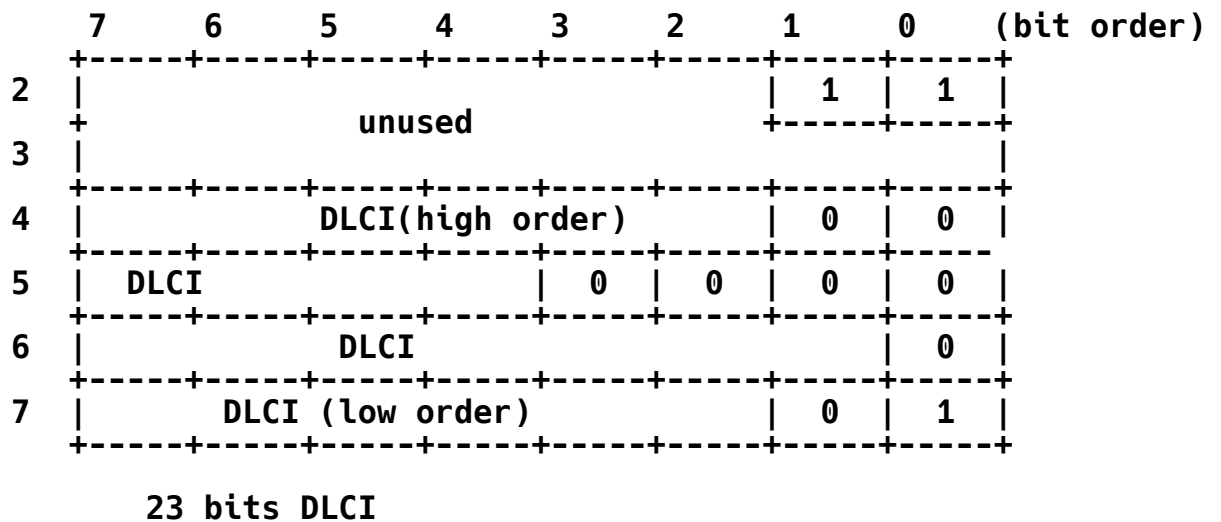
   The "DLCI Format" is defined as follows:

```
          7     6     5     4     3     2     1     0   (bit order)
        +-----+-----+-----+-----+-----+-----+-----+-----+
     0  |                     Type                      |
        +-----+-----+-----+-----+-----+-----+-----+-----+
     1  |                    Length                     |
        +-----+-----+-----+-----+-----+-----+-----+-----+
```

   with a DLCI (Q.922 address) encoded as option value:

```
        7     6     5     4     3     2     1     0   (bit order)
      +-----+-----+-----+-----+-----+-----+-----+-----+
   2  |                             |  1  |  1  |
      +                 unused      +-----+-----+
   3  |                             |
      +-----+-----+-----+-----+-----+-----+-----+-----+
   4  |        DLCI(high order)     |  0  |  0  |
      +-----+-----+-----+-----+-----+-----+-----+-----+
   5  |  DLCI(low order)     |  0  |  0  |  0  |  1  |
      +-----+-----+-----+-----+-----+-----+-----+-----+
   6  |                                               |
      +                 Padding                       +
   7  |                 (zeros)                        |
      +-----+-----+-----+-----+-----+-----+-----+-----+

        10 bits DLCI

        7     6     5     4     3     2     1     0   (bit order)
      +-----+-----+-----+-----+-----+-----+-----+-----+
   2  |                             |  1  |  1  |
      +                 unused      +-----+-----+
   3  |                             |
      +-----+-----+-----+-----+-----+-----+-----+-----+
   4  |        DLCI(high order)     |  0  |  0  |
      +-----+-----+-----+-----+-----+-----+-----+-----+
   5  |  DLCI               |  0  |  0  |  0  |  0  |
      +-----+-----+-----+-----+-----+-----+-----+-----+
   6  |        DLCI(low order)            |  0  |
      +-----+-----+-----+-----+-----+-----+-----+-----+
   7  |    unused (set to 0)        |  1  |  1  |
      +-----+-----+-----+-----+-----+-----+-----+-----+

        17 bits DLCI
```

```
            7     6     5     4     3     2     1     0   (bit order)
         +-----+-----+-----+-----+-----+-----+-----+-----+
     2   |                             | 1   | 1   |
         +                   unused    +-----+-----+     |
     3   |                                               |
         +-----+-----+-----+-----+-----+-----+-----+-----+
     4   |           DLCI(high order)       | 0   | 0   |
         +-----+-----+-----+-----+-----+-----+-----+-----
     5   | DLCI                 | 0   | 0   | 0   | 0   |
         +-----+-----+-----+-----+-----+-----+-----+-----+
     6   |              DLCI                      | 0   |
         +-----+-----+-----+-----+-----+-----+-----+-----+
     7   |         DLCI (low order)          | 0   | 1   |
         +-----+-----+-----+-----+-----+-----+-----+-----+

          23 bits DLCI
```

   Option fields:

      Type   1 for Source Link-layer address.
                    2 for Target Link-layer address.

      Length   The Length of the Option (including the Type
                    and Length fields) in units of 8 octets.
                    It has the value 1.

      Link-Layer Address   The DLCI encoded as a Q.922 address.

   Description

      The "DLCI Format" option value field has two components:

      (a) Address Type -- encoded in the first two bits of the first
           two octets. Both bits are set to 1 to indicate the DLCI
           format. The rest of the bits in the two first octets are
           not used -- they MUST be set to zero on transmit and MUST
           be ignored by the receiver.

      (b) DLCI -- encoded as a Q.922 address padded with zeros to the
           last octet of the 6 octets available for the entire Link-
           Layer Address field of this format.

The "Frame Relay Address Format" is defined as follows:

```
          7     6     5     4     3     2     1     0    (bit order)
        +-----+-----+-----+-----+-----+-----+-----+-----+
    0   |                      Type                     |
        +-----+-----+-----+-----+-----+-----+-----+-----+
    1   |                     Length                    |
        +-----+-----+-----+-----+-----+-----+-----+-----+
```
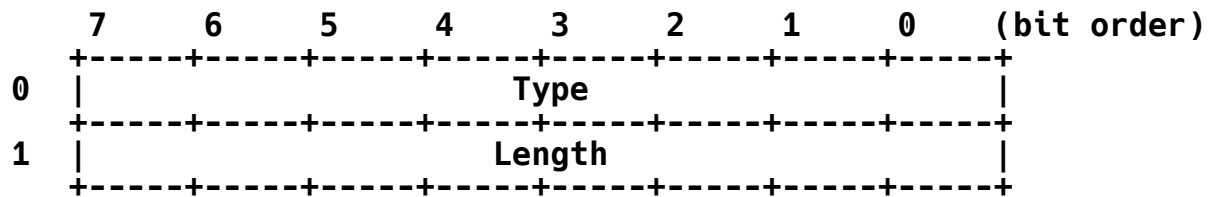
with an E.164, X.121, number or NSAP address encoded as option
value:
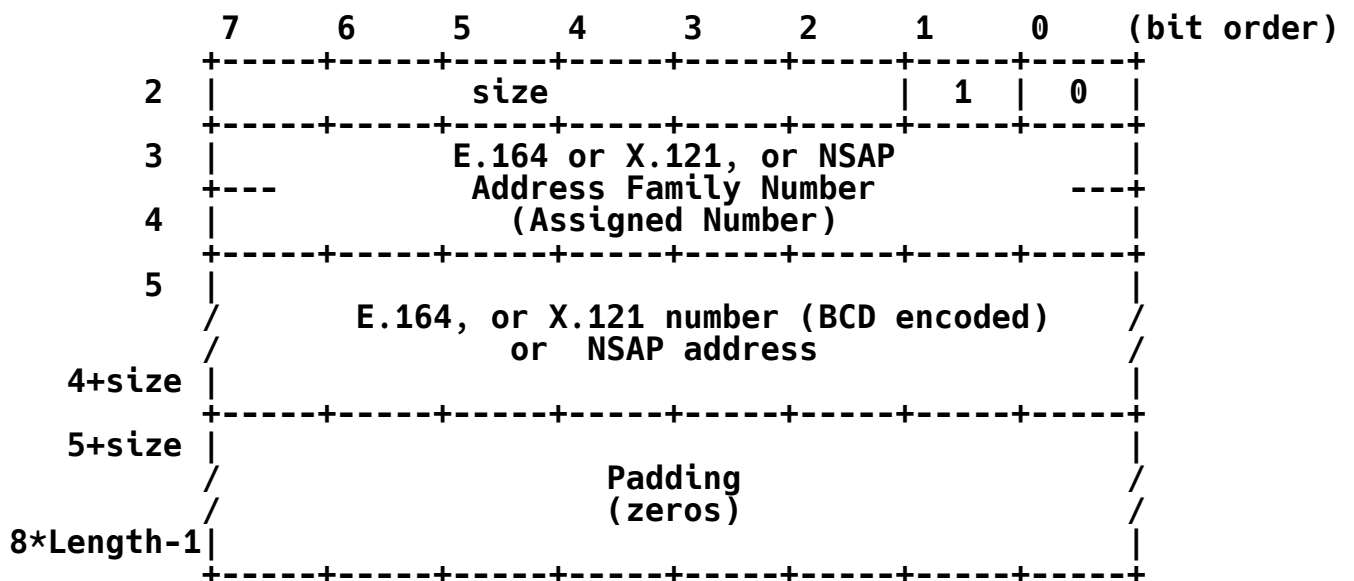
```
          7     6     5     4     3     2     1     0    (bit order)
        +-----+-----+-----+-----+-----+-----+-----+-----+
    2   |              size               |  1  |  0  |
        +-----+-----+-----+-----+-----+-----+-----+-----+
    3   |          E.164 or X.121, or NSAP              |
        +---      Address Family Number            ---+
    4   |             (Assigned Number)                 |
        +-----+-----+-----+-----+-----+-----+-----+-----+
    5   |                                               |
        /      E.164, or X.121 number (BCD encoded)     /
        /            or   NSAP address                  /
 4+size |                                               |
        +-----+-----+-----+-----+-----+-----+-----+-----+
 5+size |                                               |
        /                   Padding                     /
        /                   (zeros)                     /
8*Length-1|                                             |
        +-----+-----+-----+-----+-----+-----+-----+-----+
```

Option fields:

Type   1 for Source Link-layer address.
            2 for Target Link-layer address.

Length   The length of the Option (including the
            Type and Length fields) in units of 8 octet.
            It may have the value:

            2 -- for E.164, or X.121 numbers or NSAP
                 addresses not longer than 11 octets
                 [E164], [X25], [NSAP].

            3 -- for NSAP addresses longer than 11 but
                 not longer than 19 octets.

                         4 -- for NSAP addresses longer than 19 octets
                              (not longer than the maximum NSAP address
                              length) [NSAP].

          Link-Layer Address   The E.164, X.121, number encoded in
                                       Binary Coded Decimal (BCD), or the NSAP
                                       address.

   Description

      The "Frame Relay Address" option value has three components:

      (a)   Address Type -- encoded in the first two bits of the first
            octet. The first bit is set to 0, the second bit is set to 1.

      (b)   Size -- encoded in the last (high order) 6 bits of the first
            octet. The maximum value of the field is the maximum size of
            the E.164, X.121, or NSAP addresses.

      (c)   Address Family Number -- the number assigned for the E.164,
            X.121, or NSAP address family [ASSNUM].

      (d)   E.164, X.121, number -- encoded in BCD (two digits per octet).
            If the E.164, or X.121 has an even number of digits the
            encoding will fill all encoding octets -- half the number of
            digits. If the E.164, or X.121 number has an odd number of
            digits, the lowest order digit fills only half of an octet --
            it is placed in the first 4 bits of the last octet of the
            E.164, or X.121 BCD encoding. The rest of the field up to the
            last octet of the 11 octets available is padded with zeros.

            NSAP address -- the NSAP address. It is padded with zeros if
            the NSAP address does not fit in a number of octets that makes
            the length of the option an even number of 8 octets.

7. Sending Neighbor Discovery Messages

   Frame Relay networks do not provide link-layer native multicasting
   mechanisms. For the correct functioning of the Neighbor Discovery
   mechanisms, link-layer multicasting must be emulated.

   To emulate multicasting for Neighbor Discovery (ND) the node MUST
   send frames carrying ND multicast packets to all VCs on a Frame Relay
   interface. This applies to ND messages addressed to both all-node and
   solicited-node multicast addresses. This method works well with PVCs.
   A mesh of PVCs MAY be configured and dedicated to multicast traffic
   only. An alternative to a mesh of PVCs is a set of point-to-
   multipoint PVCs.

8. Receiving Neighbor Discovery Messages

   If a Neighbor Discovery Solicitation message received by a node
   contains the Source link-layer address option with a DLCI, the
   message MUST undergo Frame Relay specific preprocessing required for
   the correct interpretation of the field during the ND protocol engine
   processing. This processing is done before the Neighbor Discovery
   message is processed by the Neighbor Discovery (ND) protocol engine.

   The motivation for this processing is the local significance of the
   DLCI fields in the Neighbor Discovery message: the DLCI significance
   at the sender node is different than the DLCI significance at the
   receiver node. In other words, the DLCI that identifies the Frame
   Relay virtual circuit at the sender may be different than the DLCI
   that identifies the virtual circuit at the receiver node.
   Furthermore, the sender node may not be aware of the DLCI value at
   the receiver. Therefore, the Frame Relay specific preprocessing
   consists in modifying the Neighbor Discovery Solicitation message
   received, by storing into the Source link-layer address option the
   DLCI value of the virtual circuit on which the frame was received, as
   known to the receiver node. The DLCI value being stored must be
   encoded in the appropriate format (see previous sections). The
   passing of the DLCI value from the Frame Relay module to the Neighbor
   Discovery preprocessing module is an implementation choice.

9. Security Considerations

   The mechanisms defined in this document for generating an IPv6 Frame
   Relay interface identifier are intended to provide uniqueness at link
   level -- virtual circuit. The protection against duplication is
   achieved by way of IPv6 Stateless Autoconfiguration Duplicate Address
   Detection mechanisms. Security protection against forgery or accident
   at the level of the mechanisms described here is provided by the IPv6
   security mechanisms [IPSEC], [IPSEC-Auth], [IPSEC-ESP] applied to
   Neighbor Discovery [IPv6-ND] or Inverse Neighbor Discovery [IND]
   messages.

   To avoid an IPsec Authentication verification failure, the Frame
   Relay specific preprocessing of a Neighbor Discovery Solicitation
   message that contains a DLCI format Source link-layer address option,
   MUST be done by the receiver node after it completed IP Security
   processing.

10. Acknowledgments

Thanks to D. Harrington, and M. Merhar for reviewing this document and providing useful suggestions. Also thanks to G. Armitage for his reviewing and suggestions. Many thanks also to Thomas Narten for suggestions on improving the document.

11. References

[AARCH]   Hinden, R. and S. Deering, "IPv6 Addressing
             Architecture", RFC 2373, July 1998.

[ASSNUM]   Reynolds, J. and J. Postel, "Assigned Numbers", STD 2,
             RFC 1700, October 1994. See also:
             http://www.iana.org/numbers.html

[AUTOCONF]   Thomson, S. and T. Narten, "IPv6 Stateless
             Autoconfiguration", RFC 2462, December 1998.

[CANON]   Narten, T. and C. Burton, "A Caution on the Canonical
             Ordering of Link-Layer Addresses", RFC 2469, December
             1998.

[ENCAPS]   Brown, C. and A. Malis, "Multiprotocol Interconnect over
             Frame Relay", STD 55, RFC 2427, November 1998.

[IND]   Conta, A., "Extensions to IPv6 Neighbor Discovery for
             Inverse Discovery", Work in Progress, December 1998.

[IPv6]   Deering, S. and R. Hinden, "Internet Protocol Version 6
             Specification", RFC 2460, December 1998.

[IPv6-ATM]   Armitage, G., Schulter, P. and M. Jork, "IPv6 over ATM
             Networks", RFC 2492, January 1999.

[IPv6-ETH]   Crawford, M., "Transmission of IPv6 packets over
             Ethernet Networks", RFC 2464, December 1998.

[IPv6-FDDI] Crawford, M., "Transmission of IPv6 packets over FDDI
             Networks", RFC 2467, December 1998.

[IPv6-NBMA] Armitage, G., Schulter, P., Jork, M. and G. Harter,
             "IPv6 over Non-Broadcast Multiple Access (NBMA)
             networks", RFC 2491, January 1999.

[IPv6-ND]   Narten, T., Nordmark, E. and W. Simpson, "Neighbor
             Discovery for IP Version 6 (IPv6)", RFC 2461, December
             1998.

   [IPv6-PPP]   Haskin, D. and E. Allen, "IP Version 6 over PPP", RFC
                 2472, December 1998.

   [IPv6-TR]   Narten, T., Crawford, M. and M. Thomas, "Transmission
                 of IPv6 packets over Token Ring Networks", RFC 2470,
                 December 1998.

   [IPSEC]   Atkinson, R. and S. Kent, "Security Architecture for the
                 Internet Protocol", RFC 2401, November 1998.

   [IPSEC-Auth] Atkinson, R. and S. Kent, "IP Authentication Header",
                 RFC 2402, December 1998.

   [IPSEC-ESP]   Atkinson, R. and S. Kent, "IP Encapsulating Security
                 Protocol (ESP)", RFC 2406, November 1998.

   [RFC2119]   Bradner, S., "Key words for use in RFCs to indicate
                 Requirement Levels", BCP 14, RFC 2119, March 1997.

   [E164]   International Telecommunication Union - "Telephone
                 Network and ISDN Operation, Numbering, Routing, amd
                 Mobile Service", ITU-T Recommendation E.164, 1991.

   [NSAP]   ISO/IEC, "Information Processing Systems -- Data
                 Communications -- Network Service Definition Addendum 2:
                 Network Layer Addressing". International Standard
                 8348/Addendum 2, ISO/IEC JTC 1, Switzerland 1988.

   [X25]   "Information Technology -- Data Communications -- X.25
                 Packet Layer Protocol for Data Terminal Equipment",
                 International Standard 8208, March 1988.

12. Authors' Addresses

Alex Conta
Lucent Technologies Inc.
300 Baker Ave, Suite 100
Concord, MA 01742

Phone: +1-978-287-2842
EMail: aconta@lucent.com

Andrew Malis
Ascend Communications
1 Robbins Rd
Westford, MA 01886

Phone: +1-978-952-7414
EMail: malis@ascend.com

Martin Mueller
Lucent Technologies Inc.
300 Baker Ave, Suite 100
Concord, MA 01742

PHone: +1-978-287-2833
EMail: memueller@lucent.com

13. Full Copyright Statement

Acknowledgement