CCAMP Working Group   Igor Bryskin (Ed)
Internet Draft   Wes Doonan
Intended status: Standards Track   ADVA Optical Networking
                                          Vishnu Pavan Beeram (Ed)
                                                   John Drake (Ed)
                                                     Gert Grammel
                                                 Juniper Networks
                                                      Manuel Paul
                                                   Ruediger Kunze
                                                 Deutsche Telekom
                                              Friedrich Armbruster
                                                    Cyril Margaria
                                                     Coriant GmbH
                                             Oscar Gonzalez de Dios
                                                       Telefonica
                                                Daniele Ceccarelli
                                                        Ericcsson

Expires: March 12, 2014   September 12, 2013

        Generalized Multiprotocol Label Switching (GMPLS) External Network
          Network Interface (E-NNI): Virtual Link Enhancements for the
                                  Overlay Model
                     draft-beeram-ccamp-gmpls-enni-03.txt

Status of this Memo

   This Internet-Draft will expire on March 12, 2014.

Copyright Notice

Abstract

   This memo is a companion document to [RFC4208]. It describes how the
   client domain networking in the overlay model can be enhanced via
   presenting to the client the network domain as an overlay topology
   made of Virtual TE Links.

Conventions used in this document

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

Table of Contents

1. Introduction

   [RFC4208] discusses how GMPLS can be applied to the overlay model,
   which it defines to be a client network that uses a server network
   to dynamically instantiate LSPs between the client network's nodes.
   In the client network such an LSP is a link between two adjacent
   client nodes, while in the server network the LSP may transit
   multiple links and nodes; the client network is unaware of the
   server network topology.

   While the client network is unaware of the server network topology,
   [RFC4208] does suggest that there may be an exchange of routing
   information between the server network and the client network.
   Building on this premise, this memo describes how introducing a
   representation of server network domain resources into a client
   network domain topology enhances client networking in the overlay
   model

   This document is designed to be a companion document to [RFC4208],
   but because routing is generally not considered to be part of the
   definition of a UNI, this document uses the term 'External Network
   Network Interface (E-NNI)'. 'E-NNI' is generally used to indicate a
   control plane (routing and signaling) reference point for exchange
   of information between two control plane instances. In this
   document, the term 'ENNI'is specifically used to describe the
   interface between two network domains that allows the exchange of
   routing and signaling information.

2. Hybrid Topology

   Two adjacent domains in the overlay model represent, generally
   speaking, regions of dissimilar transport technology. When an end-
   to-end service crosses a boundary between the domains, it is
   necessary to execute distinct forms of service activation within
   each domain.

```
                                     |        +---+
                                     |        |   |  - router node
           +---+                     |        +---+
           | B |                     |
           +---+                     |        /-\
            /                        |       (   )  - WDM node
           /                         |        \-/
          /                          |_____
         /
        /
      /-\           /-\            /-\         +---+
     ( E )-------( G )--------( J )--------| C |
      \-/           \-/            \-/         +---+
                   /   \          /   \
                  /     \        /     \
                 /       \      /       \
                /         \    /         \
               /           \  /           \
   +---+      /-\           /-\           /-\         +---+
   | A |--------( F )--------( H )--------( I )--------| D |
   +---+      \-/           \-/           \-/         +---+
```
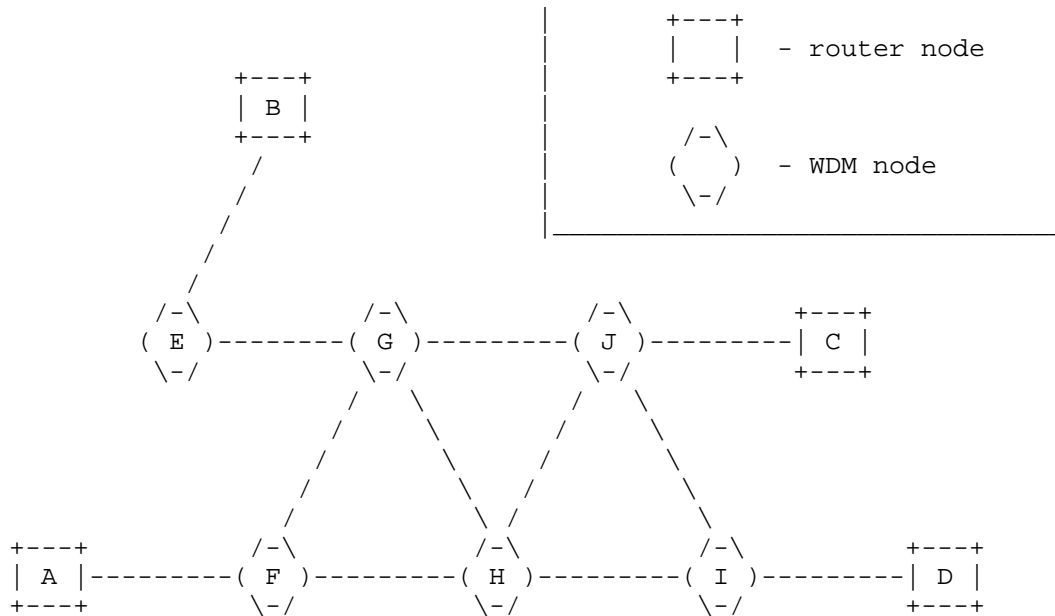
                    Figure 1: Sample hybrid topology

   For example, in the hybrid network illustrated in Fig 1,
   provisioning a transport service between two GMPLS-enabled IP
   routers (clients) on either side of the optical WDM transport
   topology (server network domain) requires operations in two distinct
   layer networks; the client layer network interconnecting the routers
   themselves, and the server layer network interconnecting the optical
   transport elements in between the routers.

   The activation of the end-to-end service begins with a path
   determination process, followed by the initiation of a signaling
   process from the ingress client network element along the determined
   path, per the example illustrated in Fig 2a-c.

```
          +---+
          | B |         |
          +---+         |   ##### - client-layer service
           /            |   ***** - server-layer WDM service
          /             |_____
         /
        /
       /
      /-\           /-\           /-\          +---+
     ( E )--------( G )---------( J )---------| C |
      \-/           \-/           \-/          +---+
                   /   \         /   \
                  /     \       /     \
                 /       \     /       \
                /         \   /         \
               /           \ /           \
  +---+ ######## /-\         /-\          /-\          +---+
  | A |--------( F )--------( H )--------( I )--------| D |
  +---+         \-/          \-/          \-/          +---+
```

                  Figure 2a: Hierarchical service activation -
                        Client-layer service setup is initiated.

```
                +---+
                | B |           |
                +---+           |   ##### - client-layer service
                 /              |   ***** - server-layer WDM service
                /               |_____
               /
              /
             /
          /-\             /-\             /-\          +---+
        ( E )--------( G )---------( J )---------| C |
          \-/           *\-/*         *\-/         +---+
                       */  \*       */   \
                      */    \*     */     \
                     */      \*   */       \
                    */        \* */         \
                   */          \*/           \
        +---+ ######## /-\          /-\            /-\          +---+
        | A |--------( F )--------( H )--------( I )--------| D |
        +---+         \-/          \-/            \-/          +---+
```
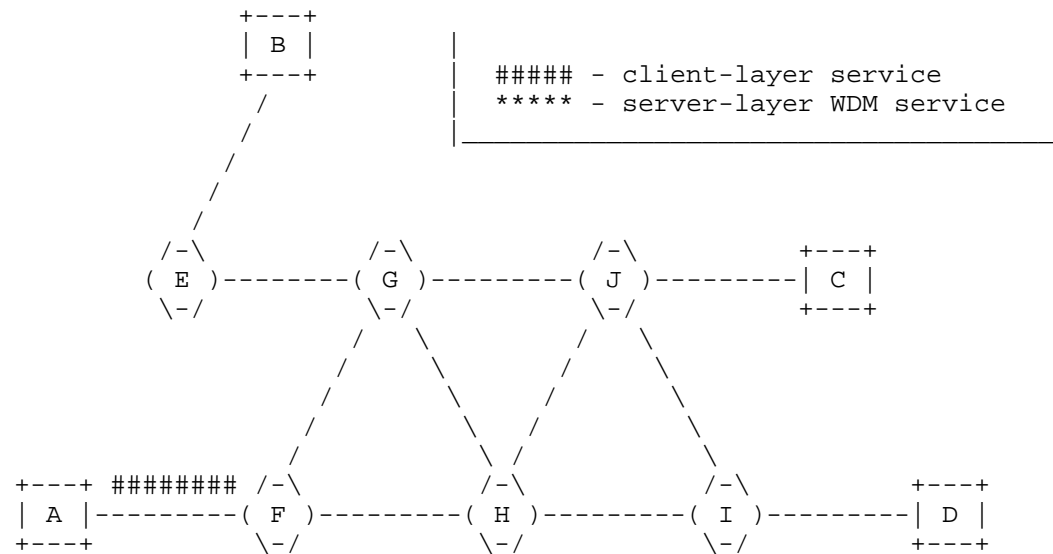
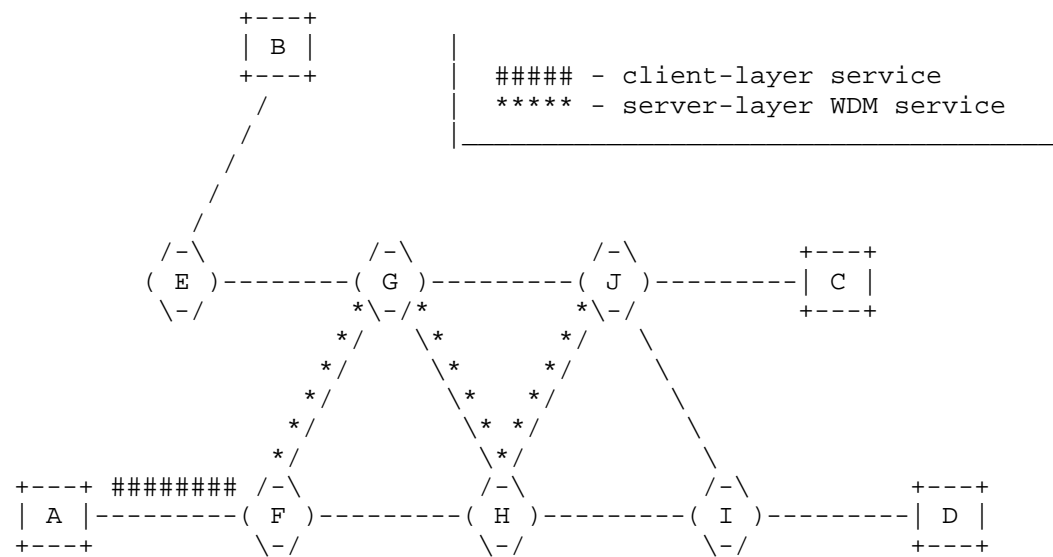   Figure 2b: Hierarchical service activation -
                        Server-layer WDM service that caters to the
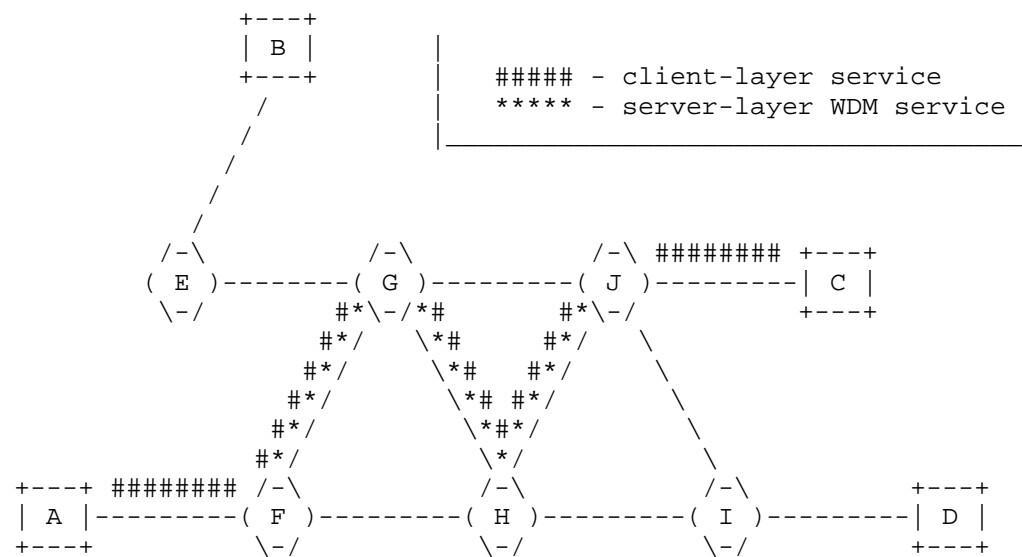   client-layer service is established within the
                        core.

```
                +---+
                | B |       |
                +---+       |    ##### - client-layer service
                 /          |    ***** - server-layer WDM service
                /           |_____
               /
              /
             /
           /-\           /-\               /-\ ######## +---+
          ( E )-------( G )---------( J )---------| C |
           \-/         #*\-/*#       #*\-/         +---+
                       #*/  \*#     #*/  \
                      #*/    \*#   #*/    \
                     #*/      \*# #*/      \
                    #*/        \*#*/        \
                   #*/          \*/          \
      +---+ ######## /-\           /-\           /-\          +---+
      | A |--------( F )---------( H )---------( I )---------| D |
      +---+         \-/           \-/           \-/          +---+
```

                Figure 2c: Hierarchical service activation –
                      Client-layer service setup is resumed and
                      the end-to-end connection is established.

3. Traffic Engineering

   The previous section outlines the basic method for activating end-
   to-end services across a multi-domain/multi-layer network. As a
   necessary part of that process an initial path selection process is
   to be performed, whereby an appropriate path between the desired
   endpoints is to be determined through some means. Further, per
   expectations set through current practices with regard to service
   provisioning in homogeneous networks, operators expect that the
   underlying control plane system provides automated mechanisms for
   computing the desired path(s) between network endpoints.

In particular, operators do not expect under normal circumstances to
be required to explicitly specify the end-to-end path; rather, they
expect to be able to specify just the endpoints of the path and rely
on an automated computational process to identify and qualify all
the elements and links on the path between them. Hence when
operating a hybrid multi-layer network such as that described in Fig
1, it is necessary to extend existing traffic engineering and path
computation mechanisms to operate in a similar manner.

Path computation and qualification operations occur at the path
computation element (PCE - RFC4655) selected by ingress network
element of an end-to-end service. In order to be able to compute and
qualify paths, the PCE should be provided with information regarding
the traffic engineering capabilities of the layer network to which
it is associated with, in particular, the topology of the layer
network and what layer-specific transport capabilities exist at the
various nodes and links in that topology.

It is important to note that topology information is layer-specific;
e.g. path computation and qualification operations occur within a
given layer, and hence information about topology and resource
availability are required for the specific layer to which the
connection belongs. The topology and resource availability
information required by a path computation element in the client
layer is quite distinct from that required by a path computation
element in the server layer network. Hence, the actual server layer
traffic engineering links are of no importance for the client layer
network. In fact, it can be desirable to block their advertisements
into the client TE domain by the border nodes.

For example, in the sample hybrid network (Fig 1) there are multiple
transport elements supporting client the connection (in this memo
terms "connection" and "LSP" are used interchangeably) between the
GMPLS-enabled clients A and C, the server layer topology between
them includes several nodes and links. However, in this example the
optical network elements are not capable of switching traffic with
the client layer granularity (i.e. IP/MPLS packets), as the optical
network elements are lambda switches, not IP/MPLS switches. Hence,
while the intervening server layer network elements may physically
exist along the path, they are not a part of the topology required
by the client layer nodes for the purposes of traffic engineering in
the client layer network.

An example of what the client layer Traffic Engineering topology
would look like for the sample hybrid network is shown in the top
half of Fig 3.

```
                                  |  +++++ - client-layer TE link
                                  |  ~~~~~ - server-layer TE link
                                  |  =====
                                  |  | N | - client-layer TE node (only)
  Client TE          =====        |  =====
  Database           | B |        |   {N}  - server-layer TE node (only)
                     =====        |  =====
                       +          |  |{N}| - server and client-layer
                        +         |  =====    TE node
                         +        |_____
                          +
                =====                    =====          =====
                |{E}|~~~~~~~~~{G}~~~~~~|{J}|++++++++| C |
                =====        ~  ~        =====          =====
                          ~     ~      ~    ~
                         ~       ~    ~      ~
                        ~         ~  ~        ~
    =====        =====           ~        =====          =====
    | A |+++++++ |{F}|~~~~~~~~~{H} ~~~~~|{I}|++++++++| D |
    =====        =====                   =====          =====


   Physical        +---+         |
   Topology        | B |         | ##### - client-layer service
                   +---+         | ***** - server-layer WDM service
                  /              |_____
                 /
                /
              /-\          /-\          /-\ ######## +---+
             ( E )--------( G )-----( J )---------| C |
              \-/          #*\-/*#    #*\-/          +---+
               #*/      \*# #*/    \
               #*/        \*#*/      \
               #*/          \*/        \
    +---+ ########## /-\          /-\          /-\          +---+
    | A |----------( F )-----( H )-----( I )---------| D |
    +---+           \-/          \-/          \-/          +---+
```
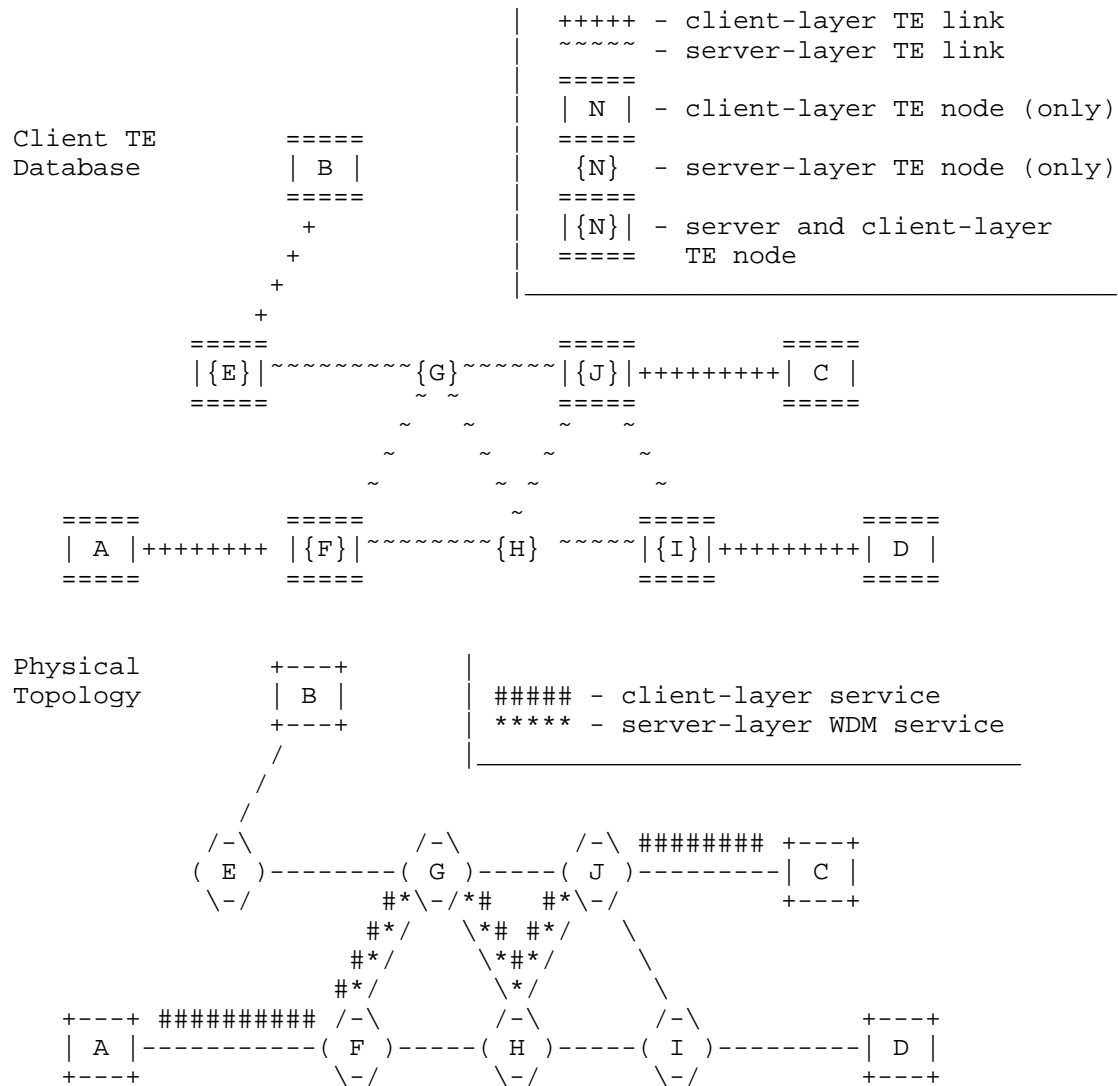
          Figure 3: Traffic engineering - ERO with "loose hop"
                   [Path = {A,F,J,C} (with J loose)].

In this example, the TE topology associated with the client layer
network is indicated by the links marked with '+' and nodes marked
without brackets, whereas the TE topology associated with the server
layer network is indicated by the links marked with '~' and nodes
marked in '{}'. The nodes at the edge of the server layer network
are visible in both the topologies. The client topology is capable
of switching traffic within the client layer, whereas the server
topology is capable of switching traffic within the server layer.

In this example, if the "B" router attempts to determine a path to
the "D" router it will be unable to do so, as the client topology to
which the B and D routers is connected does not include a full path
made of just client layer links between them. The only way to setup
an end-to-end path in this case is to use an ERO with a "loose hop"
across the server layer domain as illustrated in Fig 3. This would
cause the server layer to create the necessary link in the client
layer topology on the fly. However, this approach has a few
drawbacks - [a] the necessity for the operator to specify the ERO
with the "loose" hop; [b] potential sub-optimal usage of server
layer network resources; [c] unpredictability with regard to the
fate-sharing of the new link (that is created on the fly) with other
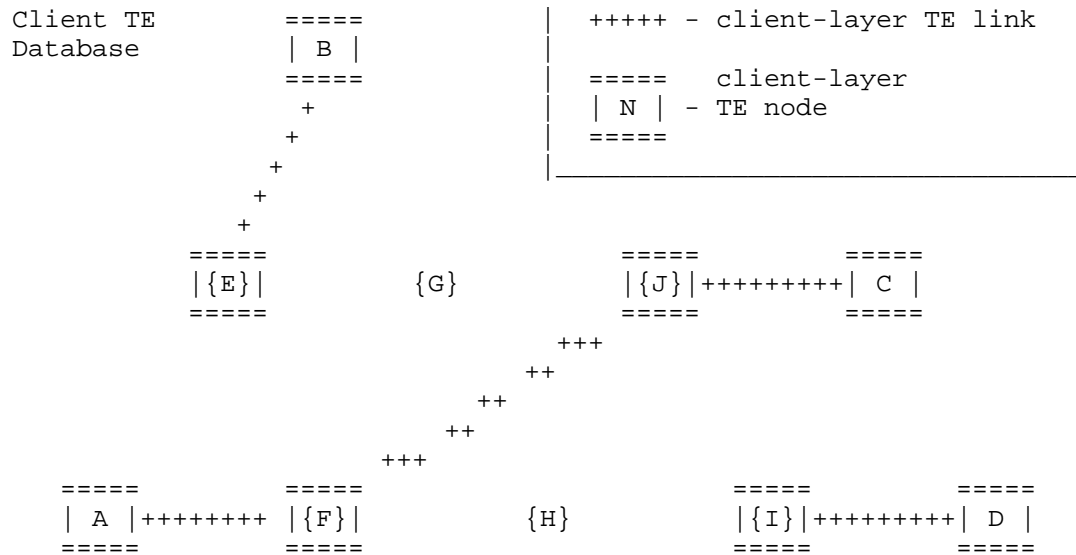links of the client layer topology.

In order to be able to compute an end-to-end path between the two
client layer endpoints, the client topology must be sufficiently
augmented to indicate where there are paths through the server
topology, which can provide connectivity between nodes in the client
topology. In other words, in order for a client to compute path(s)
across the server layer network to other clients, the feasible paths
across the server layer network should be made available (in terms
of TE links and nodes that exist in the client layer network) to all
the clients. This is discussed in detail in the next section.

As it is mentioned already, in the overlay model the client and
network domains, generally speaking, exist in separate layer-
networks. One important use case, however, is when the client and
network topologies belong to the same layer network. For example, IP
routers, connected via GMPLS ENNI to a WDM network, could be capable
of terminating optical trails being lambda switched by the network.
The method described in the following sections allows also
partitioning a single layer network into domains. Those domains do
not need to leak the full routing information to their neighboring
domains but rather provide sufficient information for a path

computation engine to route connections across a multi-domain
network.

3.1. Augmenting the Client layer Topology

In the example hybrid network, shown below in Fig 4, consider a
scenario, where each GMPLS-enabled IP router is connected to the
optical WDM transport network via a transponder. Further, consider
the situation, where the transponder on node F can be connected to
the transponder on node J via the optical path F-G-H-J. Suppose, a
lambda LSP is provisioned in the server layer along this path and
advertised (as a TE link) into the client layer network. With the
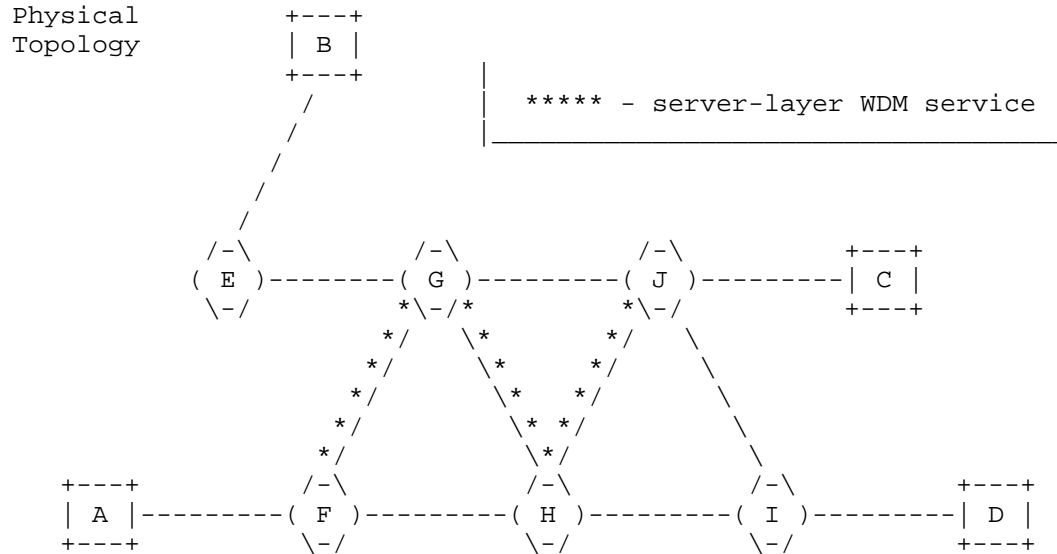availability of this TE link, the path computation function at node

```
 Client TE           =====              |   +++++ - client-layer TE link
 Database            | B |              |
                     =====              |   =====   client-layer
                      +                 |   | N | - TE node
                      +                 |   =====
                    +                   |_____
                   +
                  +
                 =====                        =====        =====
                 |{E}|          {G}          |{J}|+++++++++| C |
                 =====                        =====        =====
                                       +++
                                      ++
                                    ++
                                   ++
                                +++
      =====        =====                        =====        =====
     | A |++++++++ |{F}|          {H}          |{I}|+++++++++| D |
      =====        =====                        =====        =====
```

```
Physical           +---+
Topology           | B |
                   +---+          |
                    /             |  ***** - server-layer WDM service
                   /              |_____
                  /               |
                 /
                /
              /-\          /-\            /-\          +---+
             ( E )-------( G )--------( J )---------| C |
              \-/          *\-/*          *\-/         +---+
                     */    \*        */    \
                    */      \*      */      \
                   */        \*    */        \
                  */          \*  */          \
                 */            \*/            \
      +---+     /-\            /-\           /-\          +---+
      | A |---------( F )---------( H )---------( I )---------| D |
      +---+     \-/            \-/           \-/          +---+
```

            Figure 4: Traffic engineering - end-to-end path
                      computation.[The client layer "TE link" between F
                      and J is produced by creating the underlying
                      server-layer connection; Node A has visibility
                      to end-to-end (A to C) client-layer links and
                      can do CSPF]

   A is able to compute an end-to-end path from A to C. In this
   example, in order for the TE link to be made available in the client
   layer network topology, the network resources supporting the
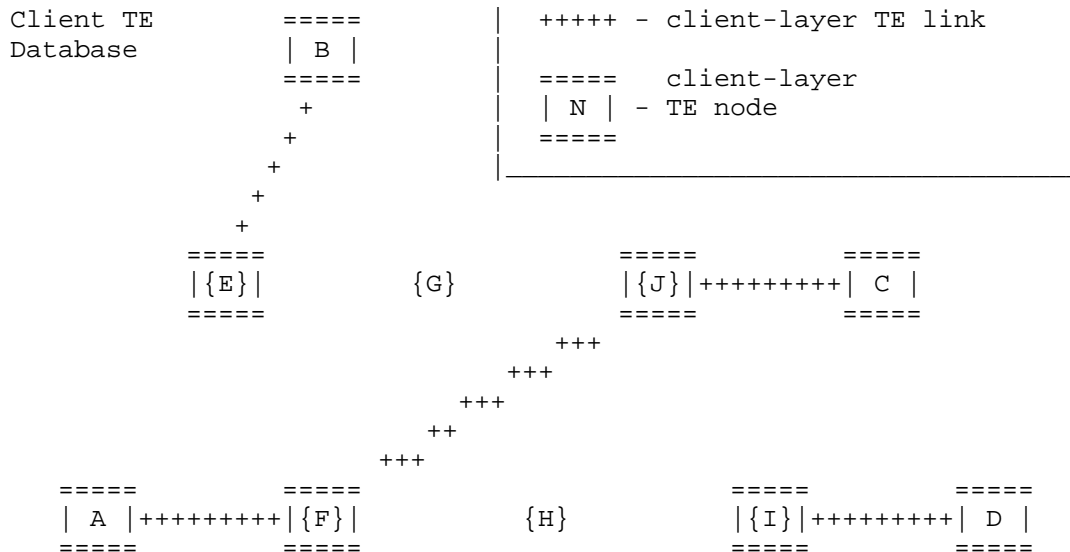   underlying server layer LSP are fully committed beforehand.

   As another scenario, consider a network configuration, where the
   transponders on nodes E, F, J and I are connected to each other via
   directionless ROADM technology. In this case it is physically
   possible to connect any transponder to any other transponder in the
   server layer network. As there are transport capabilities available
   in the server layer network between every pair of elements with an
   adaptation function to the client layer network, the operator in

this case would not wish to commit any network resources in the
server layer network until a client LSP is signaled. The next
section proposes a method to address this common operational
requirement.

3.1.1. Virtual TE Links

A "Virtual TE Link" as defined in section 7.3.3 of [RFC4847] is a TE
link that is advertised into the client layer network. The
advertisement includes information about available but not
necessarily reserved/committed resources in the server layer network
necessary to support that TE link. In other words, Virtual TE Links
represent specific transport capabilities available in the server
layer network, which can support the establishment of LSPs in the
client layer network.

The two fundamental properties of a Virtual TE Link are: [a] it is
advertised just like a real TE link and thus contributes to the
buildup of the client layer network topology; and [b] it does not
require allocation of resources at the server layer until used, thus
allowing the mutually exclusive sharing of server layer network
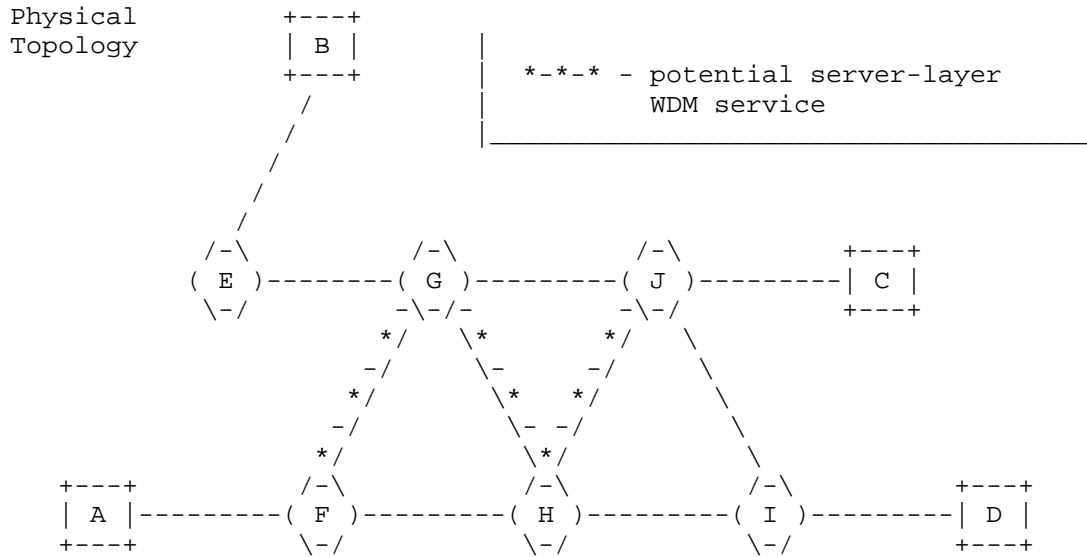resources with other Virtual TE Links.

```
 Client TE          =====        |   +++++ - client-layer TE link
 Database          | B |         |
                    =====        |   =====   client-layer
                     +           |  | N | - TE node
                     +           |   =====
                      +          |_____
                       +
                        +
                =====                       =====        =====
               |{E}|          {G}          |{J}|++++++++| C |
                =====                       =====        =====
                                   +++
                                  +++
                                 +++
                                ++
                               +++
     =====        =====                      =====        =====
    | A |++++++++|{F}|          {H}          |{I}|++++++++| D |
     =====        =====                      =====        =====
```

```
Physical              +---+
Topology              | B |         |
                      +---+         |   *-*-* - potential server-layer
                       /            |             WDM service
                      /             |_____
                     /
                    /
                   /
            /-\           /-\           /-\         +---+
          ( E )-------( G )--------( J )--------| C |
            \-/          -\-/-          -\-/        +---+
                       */  \*        */  \
                      -/    \-      -/       \
                     */      \*    */         \
                    -/        \-  -/            \
                   */          \*/               \
          +---+   /-\           /-\           /-\        +---+
          | A |--------( F )--------( H )--------( I )--------| D |
          +---+   \-/           \-/           \-/        +---+
```

                Figure 5: Traffic engineering - end-to-end path
                          computation with "Virtual TE Links". [The
                          "Virtual TE link" between F and J is created
                          in the client layer without actually
                          instantiating the underlying server-layer
                          connection; Node A has visibility to end-
                          to-end client-layer links and can do CSPF]

   In the example shown in Fig 5, the availability of a lambda channel
   along the path F-G-H-J results in the advertisement by nodes F and J
   of a Virtual TE Link between F and J into the client layer network
   topology (+++ line). With the advertisement of this Virtual TE
   Link, the path computation function at node A is able to compute an
   end-to-end path from A to C.

Whenever a Virtual TE Link gets selected and signaled in the ERO of
a client layer LSP, it ceases temporarily to be "virtual" and
transforms into a regular TE link. When this transformation takes
place, the clients will notice the change in the advertised
available bandwidth of this TE link. Also, all other Virtual TE
Links that share in a mutual exclusive way some of server layer
resources with the TE link in question SHOULD start advertising
"zero" available bandwidth. Likewise, the TE network image reverts
back to the original form as soon as the last client layer LSP,
going through the TE link in question, is released, i.e. Virtual TE
Link becomes "virtual" again.

The overlay topology, advertised into the client domain as a set of
Virtual TE Links, along with access TE links (the TE links
interconnecting client network elements with the network domain)
makes up the topology that in the overlay model allows for the
client domain path computation function to compute end-to-end paths
interconnecting client network elements across the network domain.

3.2. Macro SRLGs

The Virtual TE Links, which are advertised into the client layer
network topology, cannot be assumed to be independent. It is quite
possible for a given Virtual TE Link to share fate with one or more
other Virtual TE Link(s). This is because the underlying server
layer LSPs (established or potential) can traverse the same server
layer network link and/or node, and failure of any such shared
link/node would make all such LSPs inoperable (along with the
Virtual TE Links supported by the LSPs). If diverse end-to-end paths
for client layer LSPs are to be computed, the fate sharing
information of the Virtual TE Links needs to be taken into account.
The standard way of addressing this problem is via the concept of
Shared Risk Link Group (SRLG). Specifically, a network resource
shared by two or more TE links is identified via a network scope
unique number (SRLG ID) and advertised within each such TE link
advertisement.

A "traditional" SRLG (per [RFC4202]) represents a shared physical
network resource, upon which normal function of a link depends. Such
SRLGs can also be referred to as physical SRLGs. Zero, one or more
physical SRLGs could be identified and advertised for every TE link
in a given layer network. There is a scalability issue with physical
SRLGs in multi-layer environments. For example, if a server layer
LSP serves a client layer link, every server layer link and node
traversed by the LSP must be considered as a separate SRLG. The
number of server layer SRLGs to be advertised to client layer per

TE link is directly proportional to the number of hops traversed by
the underlying server layer LSP.

This document introduces a notion of Macro SRLGs, which addresses
this scaling problem. Macro SRLGs have the same protocol format as
their physical counterparts and can be assigned automatically for
each TE link that is advertised into the client layer network
supported by an underlying server layer LSP (instantiated or
otherwise). A Macro SRLG represents a shared path segment that is
traversed by two or more of the underlying server layer LSPs. Each
shared path segment can be viewed as a set of shared server layer
resources. The actual procedure for deriving the Macro SRLGs is
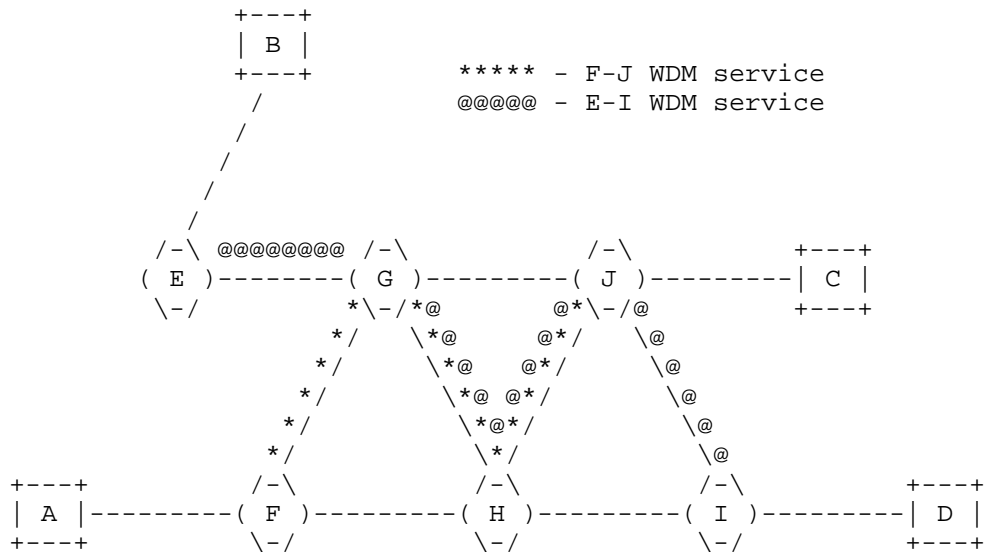beyond the scope of this document.

```
Client TE          =====          |    +25+ - client-layer TE link
Database          | B |           |          with SRLG ID "25"
                   =====           |_____
                    +
                     +
                      +
                       +
                        +
             =====                        =====          =====
            |{E}|            {G}         |{J}|++++++++| C |
             =====                        =====          =====
                ++25++            +++
                      +++++    +++
                          ++++
                      ++        +++++
                    +25+             ++++++
     =====          =====                      =====          =====
    | A |++++++++|{F}|           {H}          |{I}|++++++++| D |
     =====          =====                      =====          =====
```

```
                    +---+
                    | B |
                    +---+              ***** - F-J WDM service
                     /                 @@@@@ - E-I WDM service
                    /
                   /
                  /
                 /
             /-\ @@@@@@@@ /-\               /-\               +---+
            ( E )-------( G )---------( J )---------| C |
             \-/          *\-/*@        @*\-/@          +---+
               */      \*@      @*/    \@
                */       \*@    @*/      \@
                 */        \*@  @*/        \@
                  */         \*@*/          \@
                   */          \*/            \@
    +---+          /-\          /-\          /-\          +---+
    | A |---------( F )---------( H )---------( I )---------| D |
    +---+          \-/          \-/          \-/          +---+
```
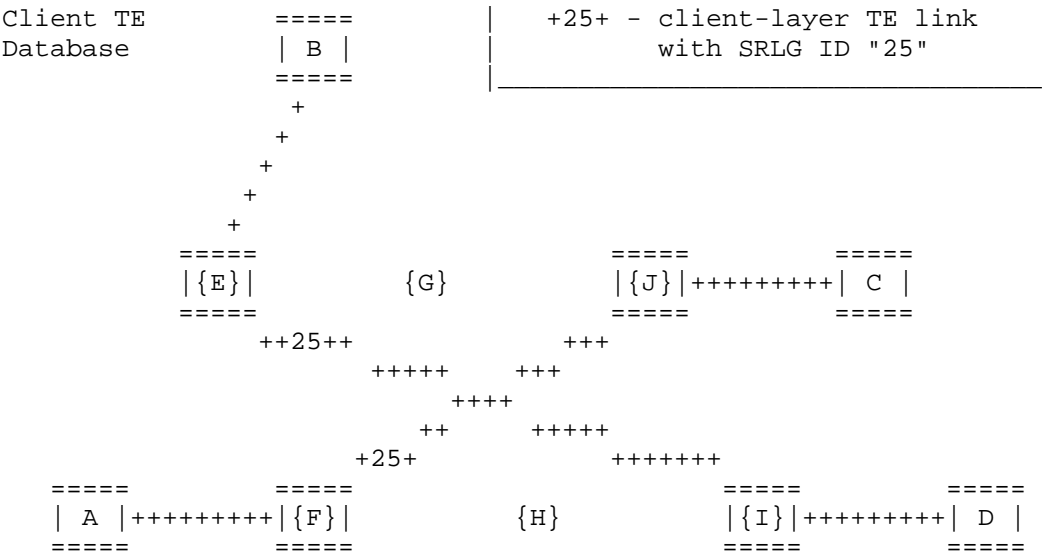
          Figure 6: Macro SRLGs - ["TE links" E-I and F-J share fate
                   since the underlying server-layer connections
                   traverse the same path segments [G-H][H-I]. Macro
                   SRLG-ID "25" is assigned to both "TE links"]

3.3. MELGs

   If two or more Virtual TE Links share fate, it means that the links
   could be concurrently activated and used by client LSPs with a
   caveat that the links could be taken out of service by a single
   network failure, and, thus, cannot be used in the same protection
   scheme. There could be a stronger (than fate sharing) relationship
   between two or more Virtual TE Links. Because a set of Virtual TE
   Links can depend on the same uncommitted network resources, the
   situation can arise, when only one Virtual TE Link from the set
   could be activated at any given time. In other words, two or more
   Virtual TE Links can be mutually exclusive.

   One example of the mutually exclusive relationship of Virtual TE
   Links is when the paths for the server layer network LSPs supporting
   the Virtual TE Links not only intersect, but also require usage of
   the same resource (e.g. lambda channel) on the intersection. Another
   example is when the said paths depend on a common physical resource

(e.g. transponder, regenerator, wavelength converter, etc.) that
could be used only by one LSP at a time.

For a client path computation function (especially a centralized one
capable of concurrent computation of multiple paths) it is important
to know about such mutually exclusive relationship between Virtual
TE Links. This document recommends the use of the extensions defined
in [MELG] to address this requirement.

3.4. Switching Constraints

Generally speaking, it SHOULD NOT be assumed that a Virtual TE Link
advertised by a given network domain border node can be cross-
connected within a client LSP with every access TE link advertised
by the said node. This circumstance necessitates the specification
of connectivity constraints by network domain border nodes. If such
information is not available for client domain path computers, there
is a significant risk of provisioning failures of client LSPs,
if/when they are signaled with the computed paths (see, Figure 7).
This document recommends the use of the advertisements specified in
[GEN_CNSTR] and [OSPF_GEN_CONSTR] to address the network element
switching limitations problem.

```
   +---+-a1------b1--/-\--b3-------------c1--/-\--c3-----d1-+---+
   | A |            ( B )                   ( C )          | D |
   +---+-a2------b2--\-/--b4-------------c2--\-/--c4-----d2-+---+
```
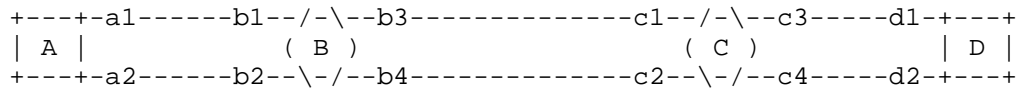

```
Access TE-links:        TE links served          Valid paths:
                        By the server domain:
a1-b1, c3-d1            b3-c1                     [a1-b1][b3-c1][c3-d1]
a2-b2, c4-d2            b4-c2                     [a2-b2][b4-c2][c4-d2]


Binding constraints:                             Invalid paths:
b1<->b3                                           [a1-b1][b4-c2]...
b2<->b4                                           [a2-b2][b3-c1]...
c1<->c3                                           [a1-b1][b3-c1][c4-d2]
c2<->c4                                           [a2-b2][b4-c2][c3-d1]
```
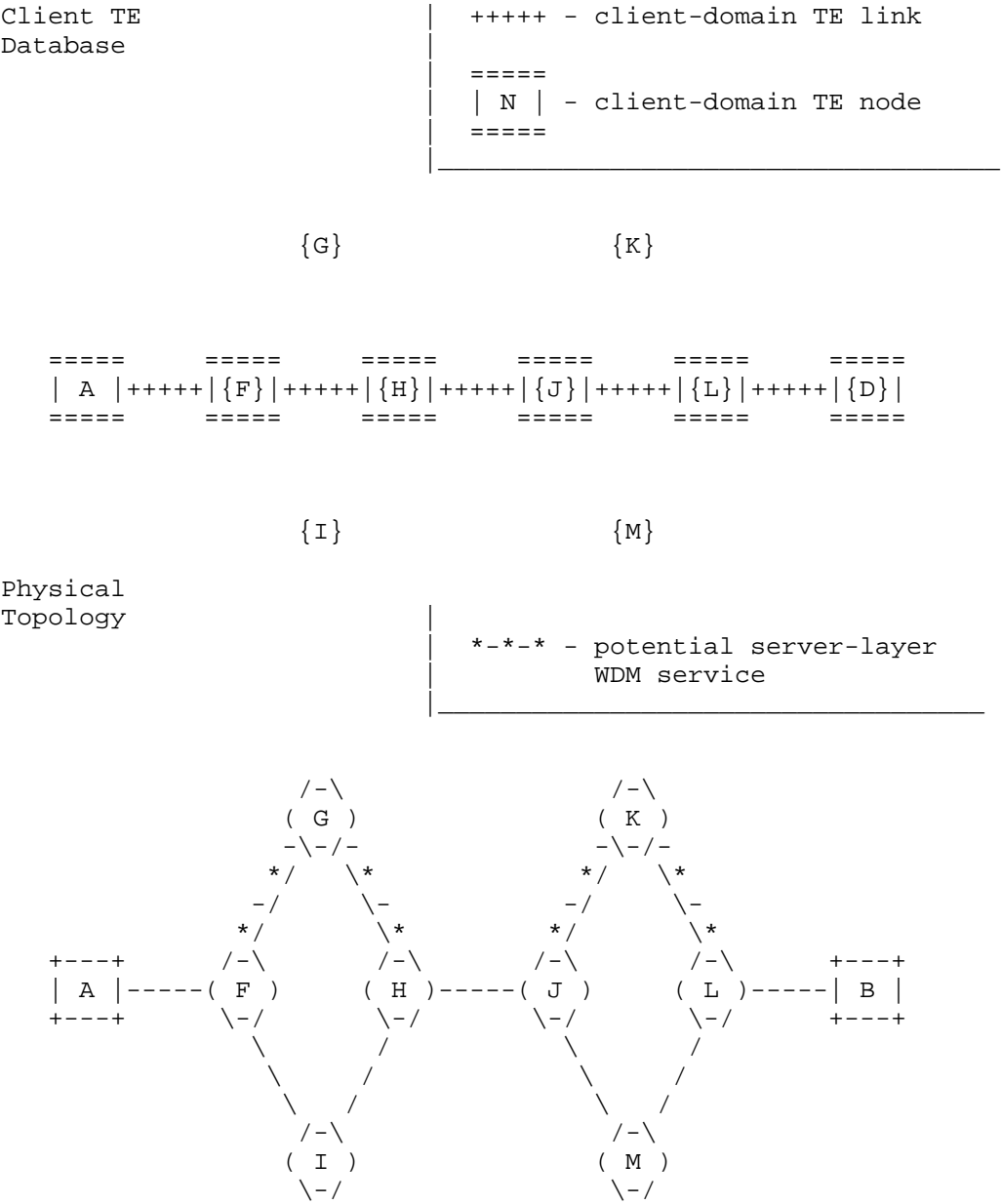
Figure 7: Switching Constraints

4. GMPLS ENNI and Multiple Server Network Domains

   In the previous sections a single server network domain GMPLS ENNI
   configuration was considered. The said configuration is modeled as a
   set of client nodes, belonging to one or more client domains,
   connected to a single server network domain. The connectivity is
   realized via access links in the data plane and GMPLS ENNI
   interfaces in the control plane. The server domain is independent
   from the client domain(s) (administratively and from the Traffic
   Engineering and control/management plane point of view). The network
   domain exposes its resources to the clients in a form of Virtual TE
   Links, thus, enabling the clients to influence the way their LSPs
   are routed across the network domain.

   There are important use cases that require client LSPs to traverse
   more than one server network domains. In such use cases the server
   domains, generally speaking, are independent from each other and
   from each of the client domains. In such configurations the clients
   would still want to control the routing of their LSPs in each of the
   server domains, the LSPs are going through, for the same reasons it
   is necessary to do so in the single server domain configuration
   (e.g. diversity, fate sharing, MELG considerations, etc.).
   Fortunately, the Virtual TE Link approach allows for exposing of the
   resources of multiple network domains in the same way as in the
   single server domain case, and, thus, provides the same tools for
   dynamic provisioning of client LSPs across either single or multiple
   server network domains.

   Multiple server network domains are modeled as separate independent
   networks interconnected with each other on their respective border
   nodes via inter-domain links in the data plane and GMPLS ENNI
   interfaces in the control plane. A network border node sees no
   difference between an access link and an inter-domain link
   terminated on the node (nor can it tell whether it is connected via
   a given link to a client node or a border node of a neighboring
   server network domain). Just like in the single-domain case, each
   server domain exposes its resources to other server and client
   network domains via Virtual TE Links configured in accordance with
   local domain policies. It is responsibility of server domain border
   nodes to advertise into the neighboring domains all access, inter-
   domain and Virtual TE Links it locally terminates, as well as
   imposed on them switching limitations. The said advertisements are
   flooded into the client domains and populate the client path
   computer's TEDs. Successful path computations produce end-to-end
   paths in the form of access, Virtual and inter-domain TE link
   chains.

```
    Client TE                  | +++++ - client-domain TE link
    Database                   |
                               | =====
                               | | N | - client-domain TE node
                               | =====
                               |_____


            {G}                        {K}



     =====     =====     =====     =====     =====     =====
    | A |+++++|{F}|+++++|{H}|+++++|{J}|+++++|{L}|+++++|{D}|
     =====     =====     =====     =====     =====     =====



            {I}                        {M}

    Physical
    Topology                   |
                               |  *-*-* - potential server-layer
                               |            WDM service
                               |_____


                 /-\                        /-\
                ( G )                       ( K )
                -\-/-                       -\-/-
              */    \*                    */    \*
             -/       \-                 -/       \-
            */          \*              */          \*
  +---+       /-\          /-\          /-\          /-\       +---+
  | A |-----( F )        ( H )-----( J )        ( L )-----| B |
  +---+       \-/          \-/          \-/          \-/       +---+
                \          /              \          /
                 \        /                \        /
                  \      /                  \      /
                   /-\                       /-\
                  ( I )                     ( M )
                   \-/                       \-/
```

    Figure 8: Multiple Network Domains ([F,G,H,I] belong to Server
    Network Domain 1;[J,K,L,M] belong to Server Network domain 2)

5. Path computation aspects

   It is assumed that a client domain path computation function makes
   use of advertised access TE links as well as Virtual TE Links, while
   computing end-to-end paths for client LSPs. The said path
   computation function could be local (i.e. located on client LSP
   ingress nodes, as stipulated by [RFC4655] Composite PCE node) or
   remote (i.e. on network PCEs). Path computations could be triggered
   by client nodes or NMS. Generally speaking, the responsibility of
   the client domain path computation function is to (concurrently)
   compute one or several paths for each source-destination pair
   (potential client LSP termination points) specified in a single path
   computation request. The path computation SHOULD be subject to one
   or more path optimization criterions (such as minimal cost, minimal
   latency, etc.) and a set of path computation constraints (such as
   link unreserved bandwidth, link colors, layer-specific constraints,
   explicit inclusions and exclusions, etc.)

   As the overlay topology hides actual server domain/layer links and
   nodes, it is RECOMMENDED to support SRLG diverse computation of two
   or more paths.

   Furthermore, the path computation SHOULD consider the
   connectivity/switching limitation constraint (when available) in
   addition to all other path computation constraints.

   The use of the PCE architecture and PCEP protocol is governed by
   [RFC5440], [RFC5521] and [RFC5541].

   As described in section 3.3., two or more Virtual TE Links may not
   only share risk, but also may exclusively depend on the same server
   layer resources. Therefore, paths, computed on network topologies
   containing Virtual TE Links, have an increased probability of LSP
   setup failures (two LSPs, for example, could be routed over two
   Virtual TE Links that exclusively depend on the same server layer
   resource). In such cases concurrent path computation, taking in
   consideration MELG information, will address this problem. PCEP
   supports concurrent path computation per [RFC5440]. Specifying MELG
   diversity constraint in path computation requests is out of scope of
   this document.

   In addition MELG may carry information on the establishment of
   server-layer resources. A Path computation request MAY constraint
   the path computation to TE-Links that are fully provisioned only.
   This information MAY also be used in PCE path computation policies.

6. Access and Virtual TE link addressing

   [RFC4208] implies that access TE links could be named from the same
   address space as network domain TE links or from a separate address
   space. This memo requires the following:

   - It MAY be possible to assign addresses for access TE links from
     the same address space as the one used for naming network internal
     TE links (i.e. TE links interconnecting network domain devices);
   - It MUST be possible to assign addresses for access TE links from a
     separate address space, independent from the space used for
     addressing network internal TE links;
   - Virtual TE Links MUST share the address space with any access TE
     links they are allowed to be cross-connected within a client LSP.

7. Use cases

7.1. Service Optimization and Restoration in Multi-layer networks

   Multi-layer networks are a reality today, and they are operated by
   different groups of people, following different operational
   procedures. This requires an independent optimization of the client
   and server layer networks. Such independence may cause a situation,
   where the re-routing of a client layer LSP fails, because some of
   resources on the selected alternate path share fate with some of
   resources on the LSP's failed path. This usually happens due to
   lack of knowledge of the server layer network by a client layer path
   computation function at the time when the alternative path is
   selected.

   The high volume and importance of IP traffic in provider networks
   today requires the client and server layer networks to share
   sufficient information in order to enable an optimized transport for
   IP/MPLS services and address existing inefficiencies. From the
   carrier perspective it is very important that the SRLG information
   is provided by the server layer TE application and is used by the
   client layer path computation.

   In a typical multi-layer network, where IP/MPLS is the client layer
   network and WDM/OTN is the server layer network, the client layer
   network is responsible for the protection of the IP/MPLS traffic
   from networks failures. This is normally achieved via using

protection schemes, such as FRR and/or LFA. Regardless of the used
mechanism, the SRLG information, provided by the server layer
network, helps to optimize the client layer network with respect to
reduced link utilization and reliable and efficient protection of
the user traffic.

Today the SRLGs information is used mainly when calculating diverse
alternative paths for the IP/MPLS LSPs. Therefore, the following
procedures are performed periodically:

- Building traffic matrix for the server layer network
  (based on IP links)
- Solving traffic engineering problems in the server layer network
- (Re-)Calculating SRLGs to be propagated into the client layer
  network
- Simulating failure scenarios
- Making sure that the affected IP/MPLS LSPs function properly after
  they are replaced onto SRLG diverse alternative paths

GMPLS ENNI reduces the OPEX costs of performing these procedures via
the automation as follows:

- server layer network automatically discovers and advertises the
  SRLG information into client layer network via a common routing
  protocol;

- client layer network path computer uses the SRLG information when
  selecting diverse paths.

7.2. IP/MPLS Offloading with ENNI automation

A typical application in multi-layer (IP/MPLS over optical) networks
is termed 'IP Offloading', in which the network responds to the
increase in traffic of a particular service or across a segment in
the IP network by dynamically creating additional IP/MPLS links
served by GMPLS LSPs provisioned in the server layer network, and
placing the extra IP/MPLS traffic onto said links. Likewise, when
the IP/MPLS traffic decreases to a normal pattern, the said GMPLS
LSPs are torn down, and the extra IP/MPLS links are removed from the
client layer network TE domain. The increase in traffic is typically
caused by an elevated number of high traffic flows/services
traversing an IP network segment.

The decision process driving IP offloading is complex, and is
governed by a set of rules. These rules reduce the cost of running
the multi-layer network, while ensuring that it remains stable.

Automation of IP Offloading poses a number of challenges. It
includes dynamic provisioning, release and maintenance of GMPLS LSPs
in the server layer (e.g. WDM) network as well as automatic
advertising/withdrawing them as (numbered or/and unnumbered) TE
links into/from the client layer network. In order to pre-plan and
manage properly the said dynamic IP/MPLS TE links, it is important
to know in advance (and also in real time) the capabilities and
resource availability of server layer network. The network
domain/layer virtualization procedures described in this document
helps to solve this complex operational issue.

7.3. Use of PCE and VNTM in Multi-layer Network Operation

   Two key elements have been proposed to help in the management and
   coordination of multi-layer networks: the Path Computation Element
   (PCE) and the Virtual Network Topology Manager (VNTM). PCE is
   responsible for the calculation of paths between endpoints,
   particularly in complex scenarios involving, for example, WDM layer
   physical impairments. VNTM is in charge of maintaining the topology
   of the client layer network by instantiating virtual links, in the
   server layer network. I.e., it can be used to provide TE links to
   the client layer network dynamically.

   Several cooperation modes between PCE, VNTM and the NMS have been
   proposed in [RFC5623]. For instance, the operator can request a new
   MPLS tunnel via the NMS, which communicates with a PCE with
   information of the multi-layer network. The PCE, in case there are
   enough resources in the IP/MPLS layer, normally returns a path for
   the tunnel made of real TE links. On the other hand, if there is a
   lack of resources in the IP/MPLS layer, the response may contain a
   path with one or more Virtual TE Links. In this case, the NMS can
   cooperate with the VNTM to suggest the set-up of a GMPLS LSP(s) in
   the server layer network. The VNTM, based on the local policies, can
   accept the suggestion and cause the set-up of the GMPLS LSPs in the
   server layer network.

   In order for the computation to be effective, the PCE needs
   knowledge of the overlay topology (SRLGs, MELGs, TE metrics of the
   Virtual TE links), which can be provided via GMPLS ENNI.

8. Security Considerations

    TBD

9. IANA Considerations

    TBD.

10. References

10.1. Normative References

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

    [RFC4202]   K. Kompella, Y.Rekhter
                      "Routing Extensions in Support of Generalized
                      Multi-Protocol Label Switching (GMPLS)",
                      RFC 4202, October 2005.

    [RFC4208]   G. Swallow, J.Drake, H. Ishimatsu, and Y. Rekhter,
                      "GMPLS UNI: RSVP-TE Support for the Overlay Model",
                      RFC 4208, October 2005.

    [GEN_CNSTR]   G.Bernstein, Y.Lee, D.Li, W.Imajuku, "General
                      Network Element Constraint Encoding for GMPLS
                      Controlled Networks"
                      [draft-general-constraint-encode-10.txt]

    [OSPF_GEN_CNSTR] F.Zhang, J.Han, Y.Lee, D.Li, G.Bernstein, Y.Hu
                      "OSPF-TE Extensions for General Network Element
                      Constraints"
                      [draft-general-constraints-ospf-te-04.txt]

    [MELG]   V.Beeram, I.Bryskin, et al, "Mutually Exclusive
                      Link Groups", [draft-beeram-ccamp-melg-01.txt]

    [TE INFO XCHG]   A.Farrel, N.Bitar, G.Swallow, D.Ceccarelli,
                      "Problem Statement and Architecture for Information
                      Exchange Between Interconnected Traffic Engineered
                      Networks", [draft-farrel-interconnected-te-info-
                      exchange-01.txt]

10.2. Informative References

    [RFC4847]   T. Takeda, "Framework and Requirements for Layer 1

                    VPNs", RFC 4847, April 2007.

   [RFC4655]   A. Farrel, J.-P. Vasseur, J. Ash, "A Path
                    Computation Element (PCE)-Based Architecture", RFC
                    4655, August 2006.

11. Acknowledgments

   Chris Bowers [cbowers@juniper.net]

Authors' Addresses
   Igor Bryskin
   ADVA Optical Networking

   Email: ibryskin@advaoptical.com

   Wes Doonan
   ADVA Optical Networking

   Email: wdoonan@advaoptical.com

   Vishnu Pavan Beeram
   Juniper Networks

   Email:vbeeram@juniper.net

   John Drake
   Juniper Networks

   Email: jdrake@juniper.net

   Gert Grammel
   Juniper Networks

   Email: ggrammel@juniper.net

   Manuel Paul
   Deutsche Telekom

   Email: Manuel.Paul@telekom.de

Ruediger Kunze
Deutsche Telekom

Email: Ruediger.Kunze@telekom.de

Oscar Gonzalez de Dios
Telefonica

Email: ogondio@tid.es

Cyril Margaria
Coriant GmbH

Email: cyril.margaria@coriant.com

Friedrich Armbruster
Coriant GmbH

Email: friedrich.armbruster@coriant.com

Daniele Ceccarelli
Ericsson

Email: daniele.ceccarelli@ericsson.com