Teas Working Group
Internet Draft

Intended status: Informational

Expires: September 2016

Young Lee
Huawei

Sergio Belotti
Alcatel-Lucent

Dhruv Dhody
Huawei

Daniele Ceccarelli
Ericsson

Bin Young Yun
ETRI

March 9, 2016

Information Model for Abstraction and Control of TE Networks (ACTN)

draft-leebelotti-teas-actn-info-02.txt

Abstract

This draft provides an information model for abstraction and control of Traffic Engineered (TE) networks (ACTN).

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

This draft provides an information model for the requirements identified in the ACTN requirements [ACTN-Req] and the ACTN interfaces identified in the ACTN architecture and framework document [ACTN-Frame].

The purpose of this draft is to put all information elements of ACTN in one place before proceeding to development work necessary for protocol extensions and data models.

The ACTN reference architecture identified a three-tier control hierarchy as depicted in Figure 1:

- Customer Network Controllers (CNC)
- Multi-Domain Service Coordinator (MDSC)
- Physical Network Controllers (PNC).

```
   +-------+                 +-------+                   +-------+
   | CNC-A |                 | CNC-B |                   | CNC-C |
   +-------+                 +-------+                   +-------+
         \___________            |             ____________ _/
          ----------             | CMI          ------------
                     \           |            /
                      +-----------------------+
                      |         MDSC          |
                      +-----------------------+
            _________/           |            \_________
            --------             | MPI          ------------____
           /                     |                          \
   +-------+                 +-------+                   +-------+
   |  PNC  |                 |  PNC  |                   |  PNC  |
   +-------+                 +-------+                   +-------+
```

Figure 1: A Three-tier ACTN control hierarchy

The two interfaces with respect to the MDSC, one north of the MDSC and the other south of the MDSC are referred to as CMI (CNC-MDSC Interface) and MPI (MDSC-PNC Interface), respectively. It is intended to model these two interfaces and derivative interfaces thereof (e.g., MDSC to MSDC in a hierarchy of MDSCs) with one common model.

Appendix A provides some relevant ACTN use-cases extracted from [ACTN-Req]. Appendix A is information only and may help readers understand the context of key use-cases addressed in [ACTN-Req].

2. ACTN Common Interfaces Information Model

This section provides ACTN common interface information model to describe in terms of primitives, objects, their properties (represented as attributes), their relationships, and the resources for the service applications needed in the ACTN context.

Basic primitives (messages) are required between the CNC-MDSC and MDSC-PNC controllers. These primitives can then be used to support different ACTN network control functions like network topology

request/query, VN service request, path computation and connection control, VN service policy negotiation, enforcement, routing options, etc.

The standard interface is described between a client controller and a server controller. A client-server relationship is recursive between a CNC and a MDSC and between a MDSC and a PNC. In the CMI, the client is a CNC while the server is a MDSC. In the MPI, the client is a MDSC and the server is a PNC. There may also be MDSC-MDSC interface(s) that need to be supported. This may arise in a hierarchy of MDSCs in which workloads may need to be partitioned to multiple MDSCs.

A Virtual Network is a client view of the transport network. It is composed by a set of physical resources sliced in the provider network and presented to the customer as a set of abstract resources i.e. virtual nodes and virtual links. Depending on the agreement between client and provider a VN can be just represented by how the end points can be connected with given SLA attributes(e.g., re satisfying the customer's objectives), or a pre-configured set of physical resources or can be created as outcome of a dynamic request from customer.

1. In the first case can be seen as an (or set of) e2e connection(s) that can be formed by recursive aggregation of lower level connections at provider level. Such end to end connections include: customer end points, access links (physical or virtual), intra domain tunnels and inter-domain link (physical or virtual).
2. When VN is pre-configured is provided after a static negotiation between customer and provider while in the third case VN can be dynamically created, deleted, or modified in response to requests from the customer. This implies dynamic changes of network resources reserved for the customer.
3. In the second and third case, once that customer has obtained his VN, can act upon the virtual network resources to perform connection management (set-up/release/modify connections).

Abstract topology: Every lower controller in the provider network, when is representing its network topology to an higher layer, it may want to hide details of the actual network topology. In such case, an abstract topology may be used for this purpose. Abstract topology enhances scalability for the MDSC to operate multi-domain networks.

Basic primitives (messages) are required between the CNC-MDSC and MDSC-PNC controllers. These primitives can then be used to support different ACTN network control functions like network topology request/query, VN service request, path computation and connection control, VN service policy negotiation, enforcement, routing options, etc.

At a minimum, the following VN action primitives should be supported:

- VN Instantiate (See Section 2.1.1. for the description)

- VN Modify (See Section 2.1.2. for the description)

- VN Delete (See Section 2.1.3. for the description)

- VN Update ((See Section 2.1.4. for the description)

- VN Path Compute (See Section 2.1.5. for the description)

- VN Query (See Section 2.1.6. for the description)

In addition to VN action primitives, TE Update primitive should also be supported (See Section 2.1.7. for the description).

2.1. VN Action Primitives

This section provides a list of main primitives necessary to satisfy ACTN requirements specified in [ACTN-REQ].

<VN Action> describes main primitives. VN Action can be one of the following primitives: (i) Instantiate; (ii) Modify; (iii) Delete; (iv) Update; (v) Path Compute; (vi) Query.

```
<VN Action> ::= <VN Instantiate> |

                <VN Modify> |

                <VN Delete> |

                <VN Update> |

                <VN Path Compute> |

                <VN Query>
```

2.1.1. VN Instantiate

<VN Instantiate> refers to an action from customers/applications to request their VNs. This primitive can also be applied from an MDSC to a PNC requesting a VN (if the domain the PNC supports can instantiate the entire VN) or a part of VN elements. Please see the definition of VN in the introduction section.

1. In the first case can be seen as an (or set of) e2e connection(s) that can be formed by recursive aggregation of lower level connections at provider level. Such end to end connections include: customer end points, access links (physical or virtual), intra domain tunnels and inter-domain link (physical or virtual).
2. When VN is pre-configured is provided after a static negotiation between customer and provider while in the third case VN can be dynamically created, deleted, or modified in response to requests from the customer. This implies dynamic changes of network resources reserved for the customer.
3. In the second and third case, once that customer has obtained his VN, can act upon the virtual network resources to perform connection management (set-up/release/modify connections).

2.1.2. VN Modify

<VN Modify> refers to an action from customers/applications to modify an existing VN (i.e., instantiated VN). This primitive can also be applied from an MDSC to a PNC requesting a VN (if the domain the PNC supports can instantiate the entire VN) or a part of VN elements.

2.1.3. VN Delete

<VN Delete> refers to an action from customers/applications to delete an existing VN. This primitive can also be applied from an MDSC to a PNC requesting a VN (if the domain the PNC supports can instantiate the entire VN) or a part of VN elements.

2.1.4. VN Update

<VN Update> refers to any update to the VN that need to be updated to the subscribers. VN Update fulfills a push model.

Note the VN Update means the connection-related information (e.g., LSPs) update that has association with VNs. See Section 2.2.6 for further details.

There are other existing and upcoming TE mechanisms to fulfill the same function as VN Update. VN Update can be built on these other existing TE mechanisms. The details are TDB.

2.1.5. VN Path Compute

<VN Path Compute> consists of Request and Reply. Request refers to an action from customers/applications to request a VN path computation. This primitive can also be applied from an MDSC to a PNC requesting a VN (if the domain the PNC supports can instantiate the entire VN) or a part of VN elements.

<VN Path Compute> Reply refers to the reply in response to <VN Path Compute> Request.

<VN Path Compute> Request/Reply is to be differentiated from a VN Instantiate. VNS connectivity. The purpose of VN Path Compute is a priori exploration to estimate network resources availability and getting a list of possible paths matching customer/applications constraints. To make this type of request Customer/application controller can have a shared (with lower controller) view of an abstract network topology on which to get the constraints used as input in Path Computation request. The list of paths obtained by the request can be used by customer/applications to give path constrains during VNS connectivity request and to compel the lower level controller (e.g. MDSC)to select the path that Client/application controller has chosen among the set of paths returned by the Path Computation primitives. The importance of this primitives is for example in a scenario like multi-domain in which the optimal path obtained by an orchestrator as sum of optimal paths for different domain controller cannot be the optial optimal path in the Client/application controller prospective. This only applies between CNC and MDSC.

2.1.6. VN Query

<VN Query> refers to any query pertaining to the VN that has been already instantiated. VN Query fulfills a pull model and permit to get topology view.

<VN Query Reply> refers to the reply in response to <VN Query>.

2.1.7. TE Update (for TE resources)

<TE Update> it is a primitives specifically related to MPI interface to provide TE resource update between any domain controller towards MDSC regarding the entire content of any "domain controller" TE topology or an abstracted filtered view of TE topology depending on negotiated policy.

```
<TE Update> ::= [<Abstraction>]<TE-topology>

<TE-topology> ::= <TE-Topology-reference> <Node-list> <Link-list>

<Node-list> ::= <Node>[<Node-list>]

<Node> ::= <Node> [<TE-Termination Points>]

<Link-list> ::= <Link>[<Link-list>]
```

Where

<Abstraction> provides information on level of abstraction (as determined a priori).

<TE-topology-reference> ::= information related to the specific te-topology related to nodes and links present in this TE-topology.

<Node-list> ::= detailed information related to a specific node belonging to a te-topology e.g. te-node-attributes.

<Link-list> ::= information related to the specific link related belonging to a te-topology e.g. te-link-attributes.

<TE-Termination Points> ::= information details associated to the termination point of te-link related to a specific node.

2.2. VN Objects

This section provides a list of objects associated with VN action primitives.

2.2.1. VN Identifier

<VN Identifier> is an identifier that identifies a unique VN.

2.2.2. VN Objective Function

<VN Objective Function> describes the requirements/preferences of VNs that customers/applications want to instantiate.

```
<VN Objective Function> ::= <VN Connectivity Type>

                (<VN Connectivity Matrix>...)
```

Note that <VN Connectivity Matrix> needs to be defined for each s-d pair being defined under a VN. Note that this is only applicable for P2P or Multi-destination type. For other types, one VN Connectivity Matrix suffices.

Where

```
      <VN Connectivity Type> ::= <P2P> | <P2MP> | <MP2MP> | <MP2P>

                                <Multi-destination>

      <VN Connectivity Matrix> ::= <Bandwidth>

                               [<Latency>]

                               [<Latency-Variation>]

                               [<Packet-Loss Objective>]
```

2.2.3. VN End-Point

<VN End-Point> Object describes the VN's customer end-point characteristics.

```
<VN End-Point> ::= (<Access Point Interface Identifier>

                [<Client Interface Capability>]

                [<Source Indicator>])...
```

It is assumed that a list of interface identifiers has been known to the server prior to any VN actions.

The Client Capability comprises the client interface capability (e.g., maximum interface bandwidth, etc.).

<Source Indicator> indicates if an End-point is source or not.

2.2.4. VN Survivability

<VN Survivability> describes all attributes related with the VN protection level and its survivability policy enforced by the customers/applications.

```
<VN Survivability> ::= <VN Protection Level>

                       <VN Survivability Policy>
```

Where

```
<VN Protection Level> ::= <No Protection> | <1+1> | <1:N>

<VN Survivability Policy> ::= <Local Reroute Allowed>

                              [<Domain Preference>]

                              <Push Allowed>

                              <Incremental Update>
```

Where

<Local Reroute Allowed> is a delegation policy to the Server to allow or not a local reroute fix upon a failure of the primary LSP.

<Domain Preference> is only applied on the MPI where the MDSC (client) provides a domain preference to each PNC (server).

<Push Allowed> is a policy that allows a server to trigger an updated VN topology upon failure without an explicit request from the client. Push action can be set as default unless otherwise specified.

<Incremental Update> is another policy that triggers an incremental update from the server since the last period of update. Incremental update can be set as default unless otherwise specified.

2.2.5. VN Action Status

<VN Action Status> is the status indicator whether the VN has been successfully instantiated, modified, or deleted in the server network or not in response to a particular VN action.

2.2.6. VN Associated LSP

<VN Associated LSP> describes the instantiated LSPs that is associated with the VN. <VN Associated LSP> is used between each domain PNC and the MDSC as part of VN Update once the VN is instantiated in each domain network.

```
<VN Associated LSP> ::= <VN Identifier> (<LSP>...)
```

2.2.7. VN Service Preference

This section provides VN Service preference. VN Service is defined in Section 2.

```
<VN Service Preference> ::= [<Location Service Preference >]

                            [<Client-specific Preference >]
```

```
                           [<End-Point Dynamic Selection Preference >]
```

Where

<Location Service Preference> describes the End-Point Location's support for certain Virtual Network Functions (VNFs) (e.g., security function, firewall capability, etc.).

<Client-specific Preference> describes any preference related to Virtual Network Service (VNS) that application/client can enforce via CNC towards lower level controllers. For example, permission the correct selection from the network of the destination related to the indicated VNF It is e.g. the case of VM migration among data center and CNC can enforce specific policy that can permit MDSC/PNC to calculate the correct path for the connectivity supporting the data center interconnection required by application.

<End-Point Dynamic Selection Preference> describes if the End-Point can support load balancing, disaster recovery or VM migration and so can be part of the selection by MDSC following service Preference enforcement by CNC.

2.3. Mapping of VN Primitives with VN Objects

This section describes the mapping of VN Primitives with VN Objects based on Section 2.2.

```
   <VN Instantiate> ::= <VN Identifier>

                        [<VN Objective Function>]

                        <VN End-Point>

                        [<VN Survivability>]

                        [<VN Service Preference>]



   <VN Modify> ::= <VN identifier>

                   [<VN Objective Function>]
                   <VN End-Point>
                   [<VN Survivability>]
                   [<VN Service Preference>]

   <VN Delete> ::= <VN Identifier>

   <VN Update> :: = <VN Identifier>
                    <VN Associated LSP>

   <VN Path Compute Request> ::= <VN Identifier>
                         [<VN Objective Function>]
                         <VN End-Point>
                         [<VN Survivability>]
                         [<VN Service Preference>]

   <VN Path Compute Reply> ::= <VN Identifier>
                               <VN Associated LSP>

   <VN Query> ::= <VN Identifier>

   <VN Query Reply> ::= <VN Identifier>
                        <VN Associated LSP>
```

3. References

3.1. Informative References

[ACTN-Req] Y. Lee, et al., "Requirements for Abstraction and Control of Transport Networks", draft-lee-teas-actn-requirements, work in progress.

[ACTN-Frame] D. Ceccarelli, et al., "Framework for Abstraction and Control of Transport Networks", draft-ceccarelli-teas-actn-framework, work in progress.

[Stateful-PCE] E. Crabbe, et al., "PCEP Extensions for Stateful PCE", draft-ietf-pce-stateful-pce, work in progress.

4. Contributors

Contributors' Addresses

Authors' Addresses

Young Lee (Editor)
Huawei Technologies
5340 Legacy Drive
Plano, TX 75023, USA
Phone: (469)277-5838
Email: leeyoung@huawei.com

Sergio Belotti (Editor)
Alcatel Lucent
Via Trento, 30
Vimercate, Italy
Email: sergio.belotti@alcatel-lucent.com

Dhruv Dhoddy
Huawei Technologies,
Divyashree Technopark, Whitefield
Bangalore, India
Email: dhruv.ietf@gmail.com

Daniele Ceccarelli
Ericsson
Torshamnsgatan,48
Stockholm, Sweden
Email: daniele.ceccarelli@ericsson.com

Bin Young Yun
ETRI
Email: byyun@etri.re.kr

Haomian Zheng
Huawei Technologies
Email: zhenghaomian@huawei.com

Xian Zhang
Huawei Technologies
Email: zhang.xian@huawei.com

Appendix A: ACTN Applications

A.1. Coordination of Multi-destination Service Requirement/Policy

```
                             +----------------+
                             |       CNC      |
                             |   (Global DC   |
                             |   Operation    |
                             |    Control)    |
                             +--------+-------+
                                      | |  Service Requirement/Policy:
                                      | |  - Endpoint/DC location info
                                      | |  - Endpoint/DC dynamic
                                      | |    selection policy
                                      | |    (for VM migration, DR, LB)
                                      | v
                            +---------+---------+
                            |  Multi-domain     | Service policy-driven
                            |Service Coordinator| dynamic DC selection
                            +-----+---+---+-----+
                                  |   |   |
                                  |   |   |
                 +----------------+   |   +----------------+
                 |                    |                    |
           +-----+-----+       +-----+------+      +------+-----+
           |   PNC for |       |  PNC for   |      |  PNC for   |
           | Transport |       | Transport  |      | Transport  |
           | Network A |       | Network B  |      | network C  |
           +-----------+       +------------+      +------------+
                 |                    |                   |
+---+      ------               ------              ------       +---+
|DC1|--////      \\\\       ////      \\\\      ////      \\\\---+DC5|
+---+ |              |     |              |    |              |  +---+
      |     TN A     +-----+     TN B     +----+      TN C    |
      /              |     |              |    |              |
     / \\\\      ////     / \\\\      ////      \\\\      ////
   +---+   ------        /      ------    \         ------ \
   |DC2|                /                  \                \+---+
   +---+               /                    \                |DC6|
                     +---+                   \ +---+         +---+
                     |DC3|                    \|DC4|
                     +---+                     +---+

                                                DR: Disaster Recovery
                                                LB: Load Balancing
```

Figure A.1: Service Policy-driven Data Center Selection

Figure A.1 shows how VN service policies from the CNC are incorporated by the MDSC to support multi-destination applications. Multi-destination applications refer to applications in which the selection of the destination of a network path for a given source needs to be decided dynamically to support such applications.

Data Center selection problems arise for VM mobility, disaster recovery and load balancing cases. VN's service policy plays an important role for virtual network operation. Service policy can be static or dynamic. Dynamic service policy for data center selection may be placed as a result of utilization of data center resources supporting VNs. The MDSC would then incorporate this information to meet the service objective of this application.

A.2. Application Service Policy-aware Network Operation

```
                           +----------------+
                           |       CNC      |
                           |   (Global DC   |
                           |   Operation    |
                           |    Control)    |
                           +--------+-------+
                                    | | Application Service Policy
                                    | | - VNF requirement (e.g.
                                    | |   security function, etc.)
                                    | | - Location profile for each VNF
                                    | v
                          +---------+---------+
                          |  Multi-domain     |  Dynamically select the
                          |Service Coordinator|  network destination to
                          +-----+---+---+-----+  meet VNF requirement.
                                |   |   |
                                |   |   |
                +---------------+   |   +----------------+
                |                   |                    |
         +------+-----+       +-----+------+      +------+-----+
         |   PNC for  |       |  PNC for   |      |  PNC for   |
         | Transport  |       | Transport  |      | Transport  |
         | Network A  |       | Network B  |      | network C  |
         |            |       |            |      |            |
         +------------+       +------------+      +------------+
                |                   |                    |
{VNF b}         |                   |                    |  {VNF b,c}
+---+      ------               ------              ------     +---+
|DC1|--////      \\\\       ////      \\\\      ////      \\\\-|DC5|
+---+ |              |     |              |    |              |+---+
     |      TN A      +---+     TN B       +--+      TN C      |
      /              |     |              |    |              |
     / \\\\      ////     / \\\\      ////      \\\\      ////
   +---+   ------        /      ------    \         ------ \
   |DC2|                /                  \                \\+---+
   +---+               /                    \                 |DC6|
    {VNF a}         +---+                    +---+            +---+
                    |DC3|                    |DC4|      {VNF a,b,c}
                    +---+                    +---+
            {VNF a, b}                      {VNF a, c}
```

Figure A.2: Application Service Policy-aware Network Operation

This scenario is similar to the previous case in that the VN service policy for the application can be met by a set of multiple destinations that provide the required virtual network functions (VNF). Virtual network functions can be, for example, security functions required by the VN application. The VN service policy by the CNC would indicate the locations of a certain VNF that can be fulfilled. This policy information is critical in finding the optimal network path subject to this constraint. As VNFs can be dynamically moved across different DCs, this policy should be dynamically enforced from the CNC to the MDSC and the PNCs.

A.3. Network Function Virtualization Service Enabled Connectivity

```
                           +----------------+
                           |       CNC      |
                           |   (Global DC   |
                           |   Operation    |
                           |    Control)    |
                           +--------+-------+
                                    | | Service Policy related to VNF
                                    | | (e.g., firewall, traffic
                                    | | optimizer)
                                    | |
                                    | v
                          +---------+---------+
                          |  Multi-domain     | Select network
                          |Service Coordinator| connectivity subject to
                          +-----+---+---+-----+ meeting service policy
                                |   |   |
                                |   |   |
                +---------------+   |   +----------------+
                |                   |                    |
         +------+-----+       +-----+------+      +------+-----+
         |   PNC for  |       |  PNC for   |      |  PNC for   |
         | Transport  |       | Transport  |      | Transport  |
         | Network A  |       | Network B  |      | network C  |
         |            |       |            |      |            |
         +------------+       +------------+      +------------+
                |                   |                    |
                |                   |                    |
+---+      ------               ------              ------     +---+
|DC1|--////      \\\\       ////      \\\\      ////      \\\\-|DC5|
+---+ |              |     |              |    |              |+---+
     |      TN A      +---+     TN B       +--+      TN C      |
      /              |     |              |    |              |
     / \\\\      ////     / \\\\      ////      \\\\      ////
   +---+   ------        /      ------    \         ------ \
   |DC2|                /                  \                \\+---+
   +---+               /                    \                 |DC6|
                    +---+                    +---+            +---+
                    |DC3|                    |DC4|
                    +---+                    +---+
```

Figure A.3: Network Function Virtualization Service Enabled Connectivity

Network Function Virtualization Services are usually setup between customers' premises and service provider premises and are provided mostly by cloud providers or content delivery providers. The context may include, but not limited to a security function like firewall, a traffic optimizer, the provisioning of storage or computation capacity where the customer does not care whether the service is implemented in a given data center or another. The customer has to provide (and CNC is providing this)the type of VNF he needs and the policy associated with it (e.g. metric like estimated delay to reach where VNF is located in the DC). The policy linked to VNF is requested as part of the VN instantiation. These services may be hosted virtually by the provider or physically part of the network. This allows the service provider to hide his own resources (both network and data centers) and divert customer requests where most suitable. This is also known as "end points mobility" case and introduces new concepts of traffic and service provisioning and resiliency (e.g., Virtual Machine mobility).

A.4. Dynamic Service Control Policy Enforcement for Performance and Fault Management

```
        +-------------------------------------------------+
        |           Customer Network Controller           |
        +-------------------------------------------------+
        1.Traffic|  /|\4.Traffic           | /|\
        Monitor& |   |  Monitor            |  | 8.Traffic
        Optimize |   |  Result   5.Service |  | modify &
        Policy   |   |             modify& |  | optimize
                \|/  |       optimize Req.\|/ | result
        +-------------------------------------------------+
        |         Multi-domain Service Coordinator        |
        +-------------------------------------------------+
        2. Path  |  /|\3.Traffic           | /|\
        Monitor  |   | Monitor             |  |7.Path
        Request  |   | Result     6.Path   |  | modify &
                 |   |            modify&  |  | optimize
                \|/  |       optimize Req.\|/ | result
        +-------------------------------------------------+
        |          Physical Network Controller            |
        +-------------------------------------------------+
```

Figure A.4: Dynamic Service Control for Performance and Fault Management

Figure A.4 shows the flow of dynamic service control policy enforcement for performance and fault management initiated by customer per VN. The feedback loop and filtering mechanism tailored for VNs performed by the MDSC differentiates this ACTN scope from traditional network management paradigm. VN level dynamic OAM data model is a building block to support this capability.

A.5. E2E VN Survivability and Multi-Layer (Packet-Optical) Coordination for Protection/Restoration

```
                        +----------------+
                        |   Customer     |
                        |   Network      |
                        |   Controller   |
                        +--------*-------+
                                 *  |     E2E VN Survivability Req.
                                 *  |     - VN Protection/Restoration
                                 *  v        - 1+1, Restoration, etc.
                          +------*-----+   - End Point (EP) info.
                          |            |
                          |    MDSC    | MDSC enforces VN survivability
                          |            | requirement, determining the
                          |            | optimal combination of Packet/
                          +------*-----+ Optical protection/restoration
                                 *       Optical bypass, etc.
                                 *
                                 *
              **********************************************
              *               *             *              *
         +----*-----+    +----*----+   +----*-----+   +----*----+
         |PNC for   |    |PNC for  |   |PNC for   |   |PNC for  |
         |Access N. |    |Packet C.|   |Optical C.|   |Access N.|
         +----*-----+    +----*----+   +----*-----+   +---*-----+
              *             --*---          *             *
              *          ///      \\\       *             *
            --*---      |   Packet   |      *         ----*-
         ///      \\\   |    Core    +------+------///      \\\
        |  Access    +----\\      ///       *     |   Access   |
        |  Network   |      ---+--          *     |   Network  |  +---+
        |\\\      ///          |            *      \\\      ///---+EP6|
        |   +---+-  |          |       -----*         -+---+      +---+
      +-+-+     |   |          +----///      \\\       |   |
      |EP1|     |   +--------------+  Optical   |      |   |  +---+
      +---+     |                  |    Core    +------+   +--+EP5|
              +-+-+                 \\\      ///              +---+
              |EP2|                    ------ |
              +---+                     |     |
                                     +--++   ++--+
                                     |EP3|   |EP4|
                                     +---+   +---+
```

Figure A.5: E2E VN Survivability and Multi-layer Coordination for Protection and Restoration

Figure A.5 shows the need for E2E protection/restoration control coordination that involves CNC, MDSC and PNCs to meet the VN survivability requirement. VN survivability requirement and its policy need to be translated into multi-domain and multi-layer network protection and restoration scenarios across different controller types. After an E2E path is setup successfully, the MDSC has a unique role to enforce policy-based flexible VN survivability requirement by coordinating all PNC domains.

As seen in Figure A.5, multi-layer (i.e., packet/optical) coordination is a subset of this E2E protection/restoration control operation. The MDSC has a role to play in determining an optimal protection/restoration level based on the customer's VN survivability requirement. For instance, the MDSC needs to interface the PNC for packet core as well as the PNC for optical core and enforce protection/restoration policy as part of the E2E protection/restoration. Neither the PNC for packet core nor the PNC for optical core is in a position to be aware of the E2E path and its protection/restoration situation. This role of the MDSC is unique for this reason. In some cases, the MDSC will have to determine and enforce optical bypass to find a feasible reroute path upon packet core network failure which cannot be resolved the packet core network itself.

To coordinate this operation, the PNCs will need to update its domain level abstract topology upon resource changes due to a network failure or other factors. The MDSC will incorporate all these update to determine if an alternative E2E reroute path is necessary or not based on the changes reported from the PNCs. It will need to update the E2E abstract topology and the affected CN's VN topology in real-time. This refers to dynamic synchronization of topology from Physical topology to abstract topology to VN topology.

MDSC will also need to perform the path restoration signaling to the affected PNCs whenever necessary.