NSI Topology v2.0

Version 1.1

John MacAuley, SURFnet

Derived from discussions with Jerry @ SC 2013

Terminology

Network

- A group of network resources managed by a single network provider and a single NSA.
- A network exposes a set of defined service types representing the services offered to a user by the network.

Intra-Network Topology

- Refers to the topology of resources within a network, and the services offered by that network.

Inter-Network Topology

- Refers to the topology of interconnected Networks and the common services offered across these interconnected Networks.
- Inter-Network Topology is concerned with describing the way in which Networks are statically interconnected by treating each Network as an aggregated set of Network capabilities and Edge Points.

Service

- A service is a “connection” between two points in a Network with certain predefined and dynamically specified characteristics that will deliver a payload from Network ingress to Network egress unmodified.

Service Type

- A predefined type of service offered by a network and specified by a Service Definition.

Terminology

Service Definition

- A document that describes the predefined characteristics and requestable elements associated with a service being offered by a Network.

Service Termination Point (STP)

- An STP names a topological location that is the ingress/egress point of a Network and is defined by a single Service Type.
- An STP can be fully specified representing a single termination point, or under specified representing a set or bundle of STP.

Service Domain

- A group of STP within a Network described by a single Service Type and can be fully interconnected without restriction.
- Service Domain is equivalent to the NSI Transfer Function as defined in the NSI Reference Architecture.

Service Demarcation Point (SDP)

- SDP are formed when a pair STPs of matching capabilities are considered adjacent (and connectable) between two Service Domains.

Service Region

- The set of interconnected Service Domains of the same Service Type. (i.e. Inter-Network Topology for that service type).

Terminology

Adaptation (Interworking)

- By definition, Service Domains of different Service Types cannot be directly connected due to the differing Service Definitions, however, an Adaptation can be defined that permits interconnection of STP from two different Service Domains using the concepts of encapsulation and adaptation.
- An Adaptation defines the (de)encapsulation or (de)adaptation of one service type into another service type if the Network is capable of offering the service.
- An Adaptation has directionality (adaptation and de-adaptation).
- Unidirectional and bidirectional Adaptations are supported, with bidirectional Adaptations containing a symmetric pair of adaptation and de-adaptation functions.
- An Adaptation can also be defined between STP of the same service type in the case where encapsulation/adaptation of the input service type results in the same output service type.
- An Adaptation has an associated Service Definition describing the Service Adaptation, parameters of the service, attributes of the service, and specifically any restrictions/ limitations.

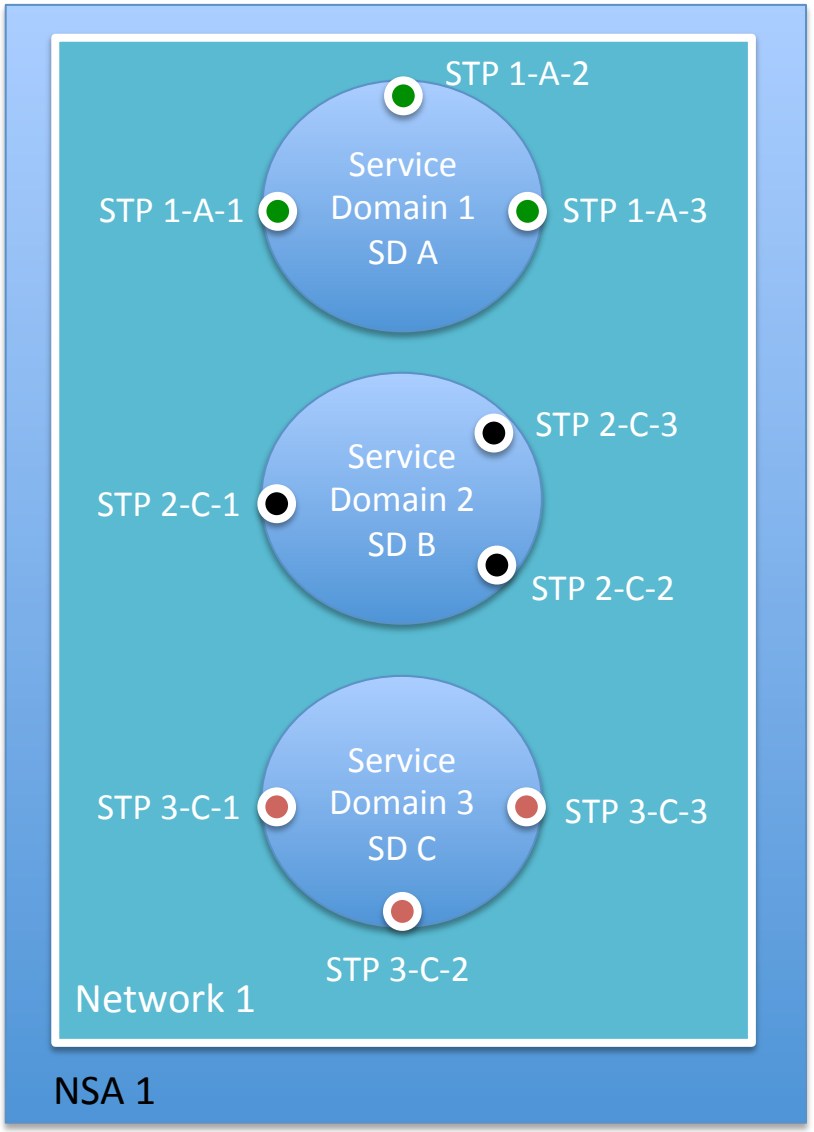
Service Domain

In a Service Domain any STP can be connected to any other STP.

A Service Domain has an associated Service Definition (SD) describing the service being offered.

Service Domains are grouped into Network topologies that can be advertised by at most one NSA.

An NSA can advertise multiple Network topologies.



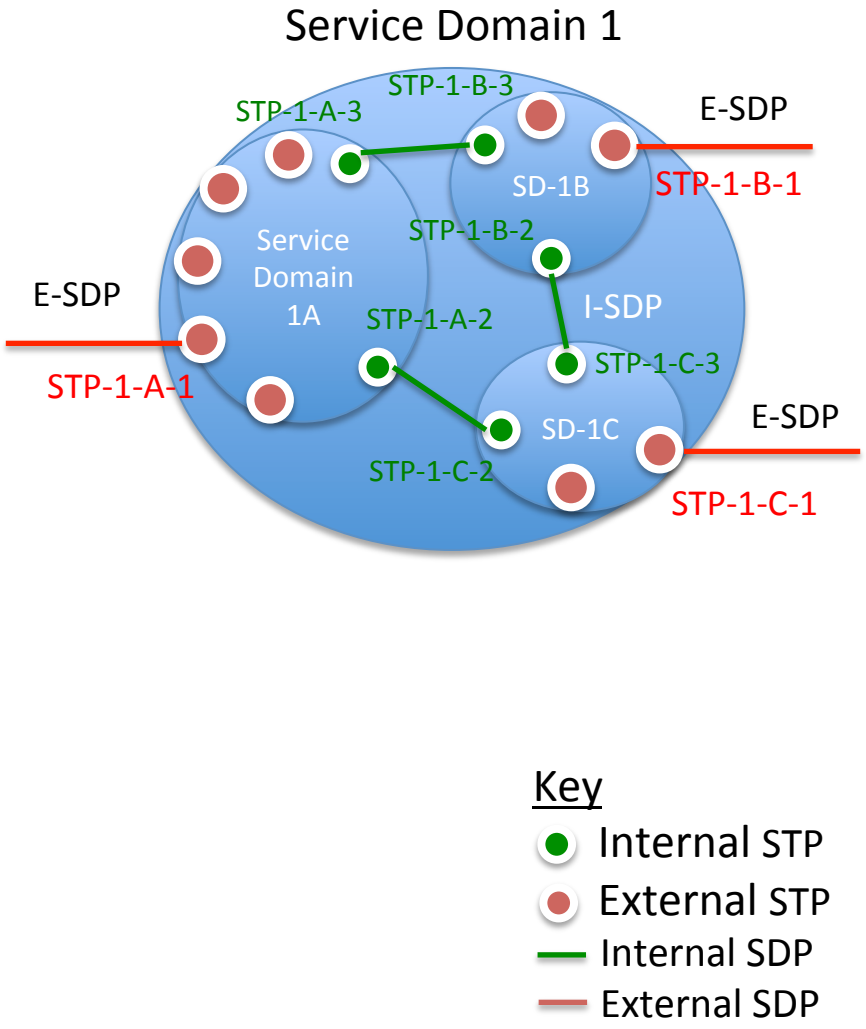
Service Domain

Service Domains can be nested to model internal structure.

Externally visible STP are used for inter-domain interconnection to peer networks.

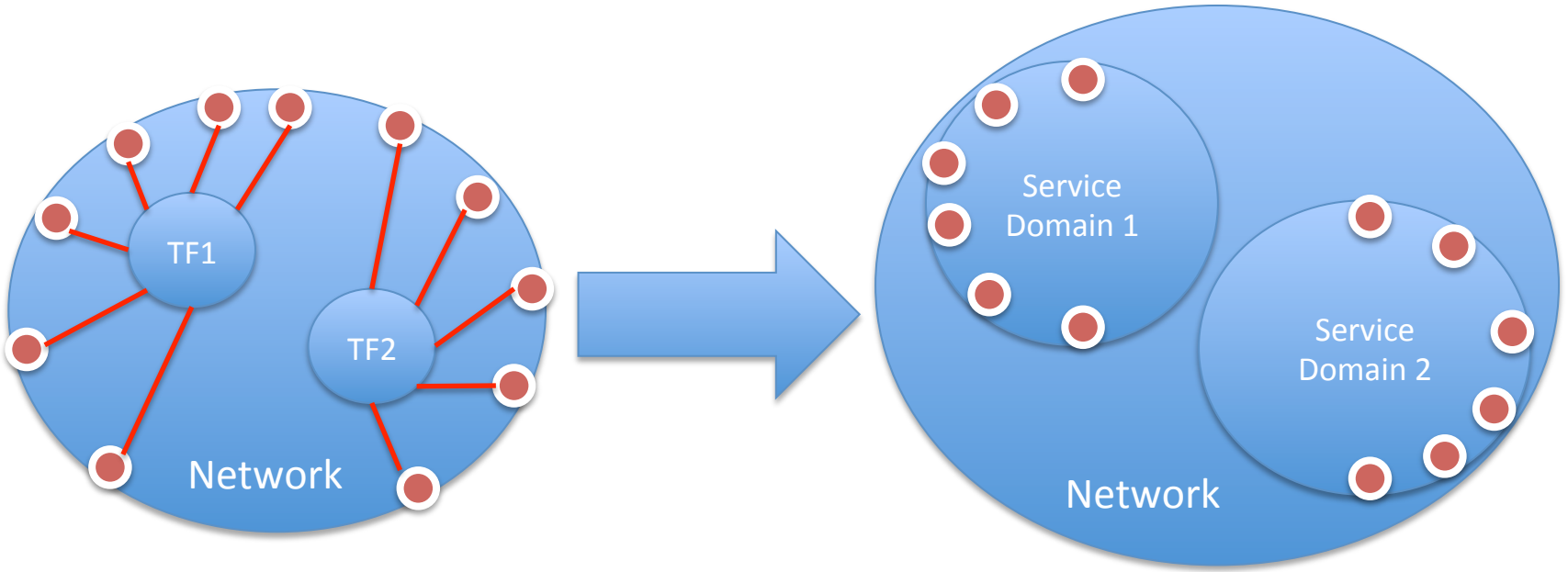
Internal STP are used to connect the internal Service subdomain as well as to the Domain's external STP points.

An external path finder could issue a request to connect STP-1-A-1 to STP-1-C-1 and delegate internal path finding to the uPA, or if the path finder would like to provide additional guidance, it could specify a more detailed path such as (STP-1-A-1, STP-1-A-3), (STP-1-B-3, STP-1-B-2), and (STP-1-C-3, STP-1-C-1).



NSI Transfer Function

- In this new model the Transfer Function is not required since similar outcomes can be achieved through the definition of separate Service Domains.
- Remember a Service Domain has the requirement for single Service Type and full interconnectivity between member STP.



Adaptation

Service Domains contain a set of STP of the same Service Type that are capable of being interconnected.

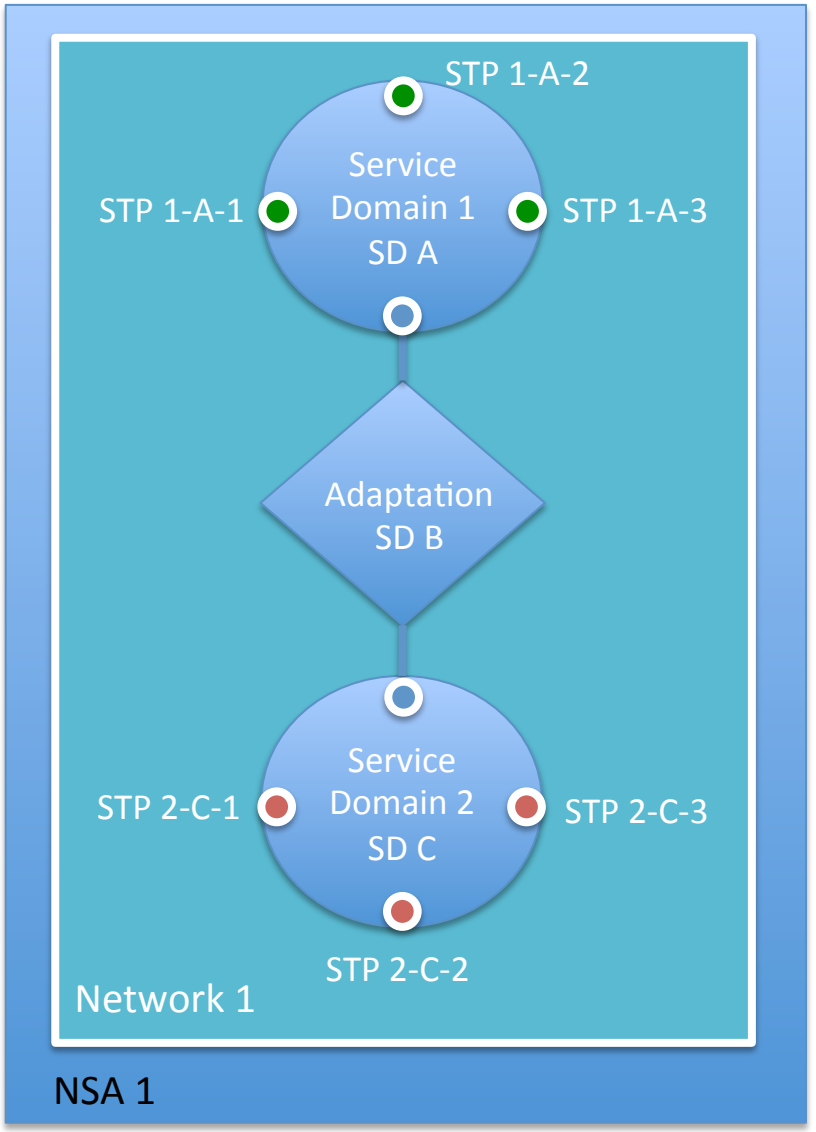
Adaptations are used to connect STP from two Service Domains within the same Network, essentially allowing a path finder to determine if it is possible to “enter” an STP in one Service Domain and “exit” an STP on a different Service Domain.

Adaptation STP are added to each Service Domain to anchor the transitional SDP associated with the Adaptation. This transitional SDP is not used in connections, and is only present to enable path finding between the domains.

Adaptations are defined with their own Service Definition, describing the capabilities of the adaptation, and the service specific parameters needed to make the reservation requests across the Adaptation.

A request to NSA 1 going from STP 1-A-2 to STP 2-C-2 can be made in a SINGLE request, however, the request parameters must specify the Service Type of the Adaptation. These parameters will contain values required by each component STP, as well as adaptation specific parameters. For example:

```
resv.req (
        serviceType=“B”,
        sourceStp=“STP 1-A-2”,
        sourceParams={Ethernet parameters},
        destStp=“STP 2-C-2”,
        destParams={SDH parameters},
        adaptParms={Adaptation specific params}
);
```



Adaptation within Service Domains

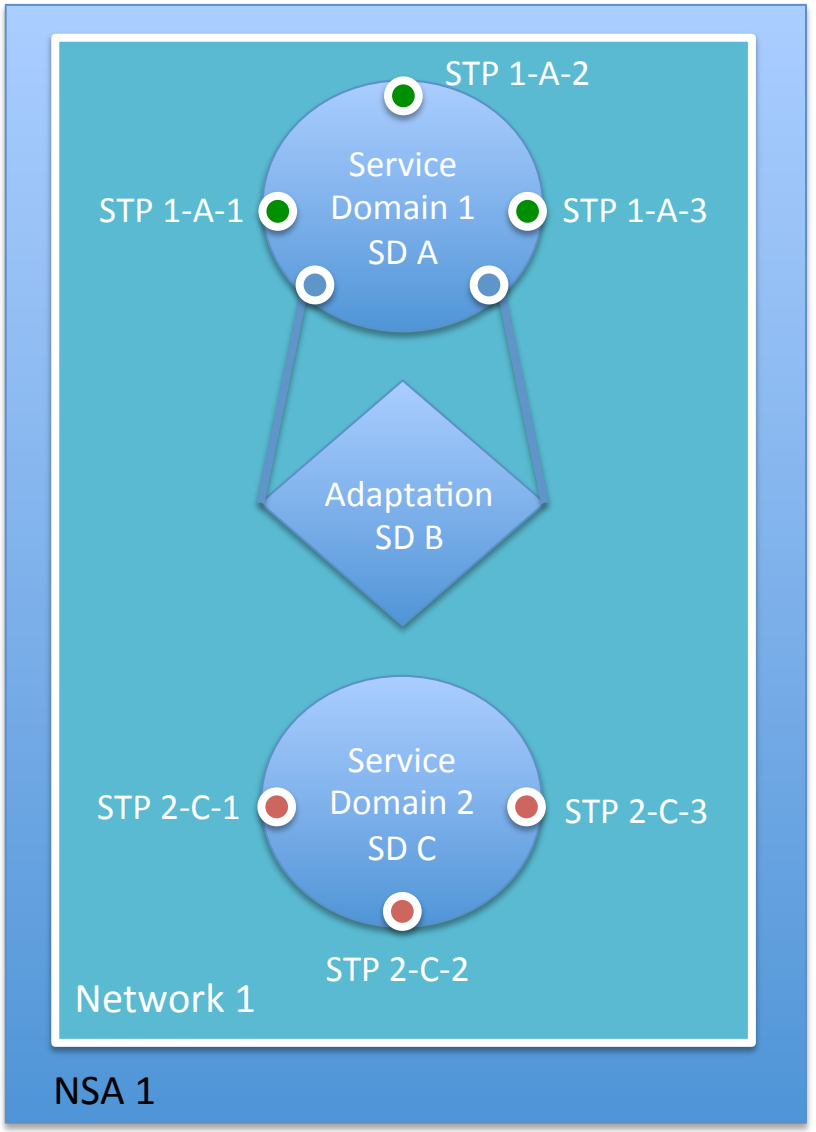
Adaptations can be defined between STP of the same type, resulting in an adaptation occurring between two STP within the same Service Domain.

A service request of this type would require specification of the Service Definition for the adaptation, and not the Service Definition associated with the Service Domain.

A request to NSA 1 going from STP 1-A-1 to STP 1-A-3 can be performed using the Service Type of the Adaptation allowing for manipulation of the ingress service. These parameters will contain values required by each component STP, as well as adaptation specific parameters. For example, we define SD B to allow the encapsulation of the ingress service PDU with the addition of a VLAN identifier:

```
resv.req (
        serviceType="B",
        sourceStp="STP 1-A-1",
        destStp="STP 1-A-3",
        EthernetParams={Ethernet parameters},
        adaptParms={Push vlan=1790}
);
```

The path finder can then "pop" the added VLAN identifier later in the path using the reverse Adaptation.



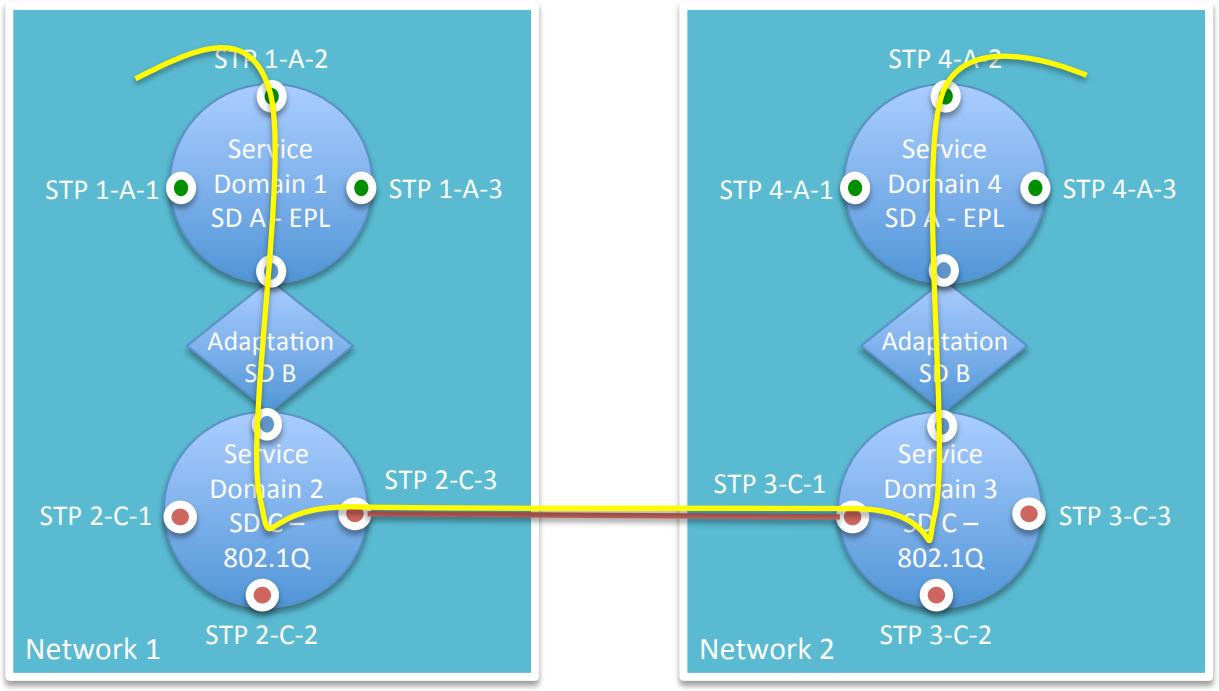
Adaptation in action

In this example we have an EPL service offered in Service Domain 1 defined by Service Description A, and an 802.1Q Trunk service in Service Domain 2 defined by Service Description C.

The EPL service offered in Service Domain 1 could be implemented using a number of technologies, however, from an end user perspective this is not important as they are interested in the service being offered, and not the technology behind the service.

For adaptation between Service Domain 1 (Service Definition A) and Service Domain 2 (Service Definition C) in Network 1 we require a transform from the EPL service to the 802.1Q Trunk compliant service. For this example, we define an Adaptation (Service Definition B) that interconnects STPs in Service Domain 1 to STPs in Service Domain 2 through encapsulation of the original EPL service Ethernet frames using 802.1AH. The 802.1AH frame is compatible with Service Domain 2, using the S-TAG from the 802.1AH frame as the switching VID. Where this encapsulation occurs in the network is irrelevant from a path finder perspective, and is left up to the supporting NSA to worry about the details.

Similarly, in Network 2 we must reverse the encapsulation performed in Network 1 to get back the original service type to deliver to the end user. In this case the paired Adaptation (Service Definition B) in Network 2 is used to remove the 802.1AH header allowing an STP in Service Domain 3 (Service Definition C) to be connected to an STP in Service Domain 4 (Service Definition A) getting us back to the original Service Type.



Terminology Alignment

Framework Terminology	New Terminology	Notes
Inter-Network Topology		The full interconnectivity of all defined networks.
	Service Region	A Service Region is a single Service Type subset of the Inter-Network Topology.
Network Service Agent (NSA)	Network Service Agent (NSA)	
Network	Network	A network object collects a set of Service Domains (Transfer Functions), STP, SDP, and Adaptations.
Transfer Function	Service Domain	A Service Region has equivalent functionality and behaviors to the Transfer Function minus any implied adaptation.
Service Termination Point (STP)	Service Termination Point (STP)	
	Adaptation	Provides ability to connect services using STP of different Service Types, or adapt between two STP of the same service type.
Service Demarcation Point (SDP)	Service Demarcation Point (SDP)	
Service Type	Service Type	
Service Definition	Service Definition	

NSI Network Service Agent

The NSA element is the root container holding all NSI topology components relating to a single NSA. It identifies the NSA managing the topology, and the version of the topology document.

```
<!-- The NSI Network Service Agent Object. -->
<ext:NSA id="urn:ogf:network:netherlight.net:2013:nsa:bod" version="2013-11-12T21:27:48+01:00"
    xmlns="http://schemas.ogf.org/nml/2013/05/base#"
    xmlns:ext="http://schemas.ogf.org/nsi/2013/09/topology#"
    xmlns:nml="http://schemas.ogf.org/nml/2013/05/base#"
    xmlns:sd="http://schemas.ogf.org/nsi/2013/07/services/definition"
    xmlns:vcard="urn:ietf:params:xml:ns:vcard-4.0"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://schemas.ogf.org/nsi/2013/09/topology# nsi-ext.xsd http://schemas.ogf.or

    <!-- User friendly and displayable name for the NSA. -->
    <name>NetherLight A-GOLE test bed</name>

    <!-- Physical location of the NSA. -->
    <Location id="urn:ogf:network:netherlight.net:2013:nsa:bod:location">
        <long>4.954585</long>
        <lat>52.3567</lat>
    </Location>

    <!-- The NSI CS Protocol endpoints supported by the NSA. -->
    <ext:Interface id="urn:ogf:network:netherlight.net:2013:nsa:bod:connection-service">
        <ext:link>https://bod.netherlight.net//nsi/v2/provider</ext:link>
        <ext:type>application/vnd.org.ogf.nsi.cs.v2+soap</ext:type>
        <ext:Relation type="http://schemas.ogf.org/nsi/2013/09/topology#providedBy">
            <ext:NSA id="urn:ogf:network:netherlight.net:2013:nsa:bod"/>
        </ext:Relation>
    </ext:Interface>

    <!-- Admin contact information for this NSA. -->
    <ext:Relation type="http://schemas.ogf.org/nsi/2013/09/topology#adminContact">
        <vcard:fn><vcard:text>Hans Trompert</vcard:text></vcard:fn>
        <vcard:n><vcard:surname>Trompert</vcard:surname><vcard:given>Hans</vcard:given></vcard:n>
        <vcard:org><vcard:text>SURFnet, http://www.surf.nl</vcard:text></vcard:org>
    </ext:Relation>

    <!-- Each Topology element represents an NSI Network object. -->
    <Topology id="urn:ogf:network:netherlight.net:2013:topology:a-gole:testbed">
```

NSA identifier associated with this topology, and the version of the document.

User friendly name of the NSA used for display.

Geographical coordinates of the NSA.

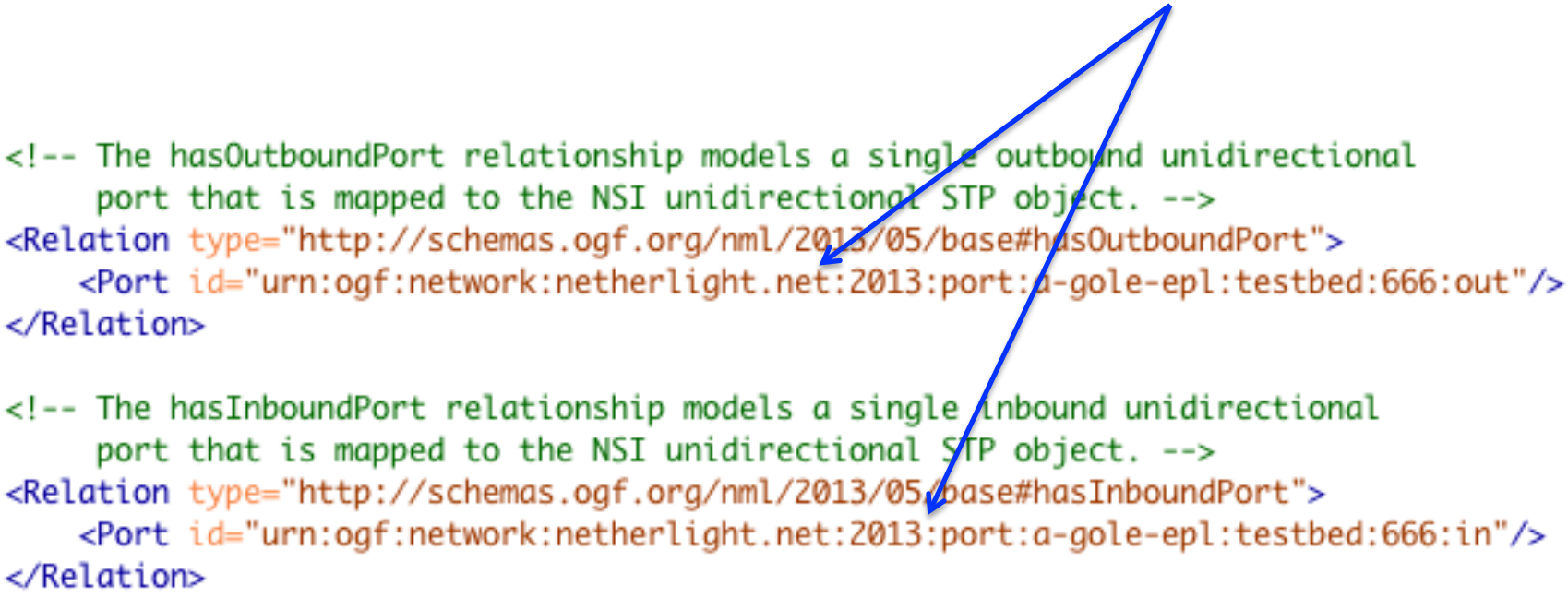
Versions of the NSI CS protocol supported by this NSA.

Administrative contact information for this NSA.

Beginning of the detailed network topology element. One topology element is specified per NSI Network with the Topology “id” mapping to the NSI Network Id.

NSI Service Termination Points

The Port element maps to a unidirectional STP.



```
<!-- The hasOutboundPort relationship models a single outbound unidirectional
     port that is mapped to the NSI unidirectional STP object. -->
<Relation type="http://schemas.ogf.org/nml/2013/05/base#hasOutboundPort">
    <Port id="urn:ogf:network:netherlight.net:2013:port:a-gole-epl:testbed:666:out"/>
</Relation>

<!-- The hasInboundPort relationship models a single inbound unidirectional
     port that is mapped to the NSI unidirectional STP object. -->
<Relation type="http://schemas.ogf.org/nml/2013/05/base#hasInboundPort">
    <Port id="urn:ogf:network:netherlight.net:2013:port:a-gole-epl:testbed:666:in"/>
</Relation>
```

From these two port definitions we would generate the following unidirectional STP identifiers:

urn:ogf:network:netherlight.net:2013:port:a-gole-epl:testbed:666:out
urn:ogf:network:netherlight.net:2013:port:a-gole-epl:testbed:666:in

Unidirectional STP from Port and Label

The Port element maps to a unidirectional STP.

```
<!-- This hasOutboundPort relationship models a single outbound
     unidirectional port that maps to the NSI unidirectional STP object. -->
<Relation type="http://schemas.ogf.org/nml/2013/05/base#hasOutboundPort">
    <Port id="urn:ogf:network:netherlight.net:2013:port:a-gole:testbed:232:out">
        <Label labeltype="http://schemas.ogf.org/nml/2013/05/ethernet#vlan">1799</Label>
    </Port>
</Relation>

<!-- This hasInboundPort relationship models a single inbound
     unidirectional port that maps to the NSI unidirectional STP object. -->
<Relation type="http://schemas.ogf.org/nml/2013/05/base#hasInboundPort">
    <Port id="urn:ogf:network:netherlight.net:2013:port:a-gole:testbed:232:in">
        <Label labeltype="http://schemas.ogf.org/nml/2013/05/ethernet#vlan">1799</Label>
    </Port>
</Relation>
```

The Label elements contains a single value that is used to fully qualify an STP. From these two port definitions we would generate the following STP identifiers:

urn:ogf:network:netherlight.net:2013:port:a-gole:testbed:232:out?vlan=1799
urn:ogf:network:netherlight.net:2013:port:a-gole:testbed:232:in?vlan=1799

Unidirectional STP from PortGroup and LabelGroup

The PortGroup element is a summary mechanism used to specify a mapping to one or more unidirectional STP.

```
<!-- This hasOutboundPort relationship models a series of outbound
     unidirectional ports by specifying a list of vlan labels.  This
     will map to multiple NSI unidirectional STP objects. -->
<Relation type="http://schemas.ogf.org/nml/2013/05/base#hasOutboundPort">
    <PortGroup id="urn:ogf:network:netherlight.net:2013:port:a-gole:testbed:manlan:1:out">
        <LabelGroup labeltype="http://schemas.ogf.org/nml/2013/05/ethernet#vlan">1779-1799</LabelGroup>
    </PortGroup>
</Relation>

<!-- This hasInboundPort relationship models a series of inbound
     unidirectional ports by specifying a list of vlan labels.  This
     will map to multiple NSI unidirectional STP objects. -->
<Relation type="http://schemas.ogf.org/nml/2013/05/base#hasInboundPort">
    <PortGroup id="urn:ogf:network:netherlight.net:2013:port:a-gole:testbed:manlan:1:in">
        <LabelGroup labeltype="http://schemas.ogf.org/nml/2013/05/ethernet#vlan">1779-1799</LabelGroup>
    </PortGroup>
</Relation>
```

The LabelGroup elements can contain a range of values that are used to generate fully qualify STP or underspecified STP. From these two port definitions we would generate the following underspecified STP identifiers:

urn:ogf:network:netherlight.net:2013:port:a-gole:testbed:232:out?vlan=1779-1799
urn:ogf:network:netherlight.net:2013:port:a-gole:testbed:232:in?vlan=1779-1799

For fully qualified STP we would get 20 of each inbound and outbound built as on the previous slide.

NSI Service Demarcation Points

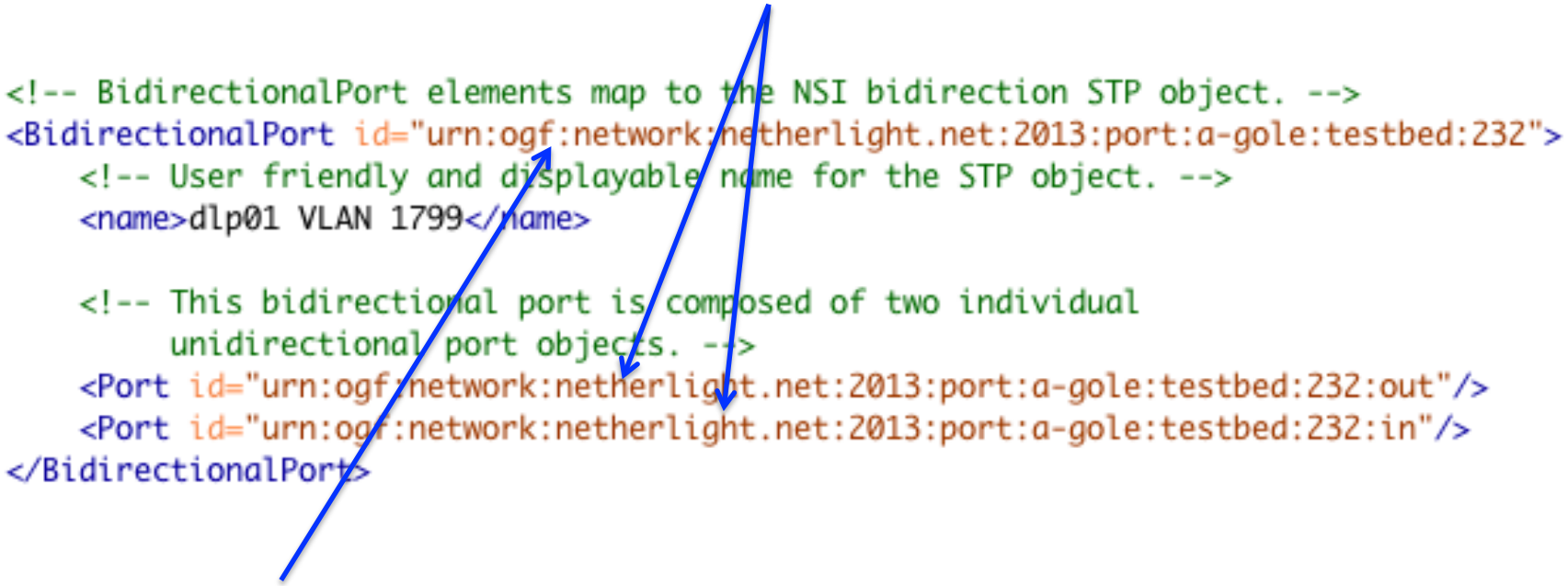
The Port and PortGroup elements may contain an isAlias relationship indicating connectivity to another Port or PortGroup. We use this isAlias relationship to create an SDP between the adjacent STP generated from the Port or PortGroup elements.

```
<!-- This hasOutboundPort relationship models a series of outbound
     unidirectional ports by specifying a list of vlan labels.  This
     will map to multiple NSI unidirectional STP objects. -->
<Relation type="http://schemas.ogf.org/nml/2013/05/base#hasOutboundPort">
    <PortGroup id="urn:ogf:network:netherlight.net:2013:port:a-gole:testbed:manlan:1:out">
        <LabelGroup labeltype="http://schemas.ogf.org/nml/2013/05/ethernet#vlan">1779-1799</LabelGroup>

        <!-- This isAlias relationsip identifies the inbound unidirectional
             port connected on the far end of this port. In this specific
             case the port is hosted in a different network. -->
        <Relation type="http://schemas.ogf.org/nml/2013/05/base#isAlias">
            <PortGroup id="urn:ogf:network:manlan.internet2.edu:2013:netherlight:in"/>
        </Relation>
    </PortGroup>
</Relation>
```

Bidirectional STP

The BidirectionalPort element groups a pair of inbound and outbound Port or PortGroup elements to create a bidirectional STP.



```
<!-- BidirectionalPort elements map to the NSI bidirection STP object. -->
<BidirectionalPort id="urn:ogf:network:netherlight.net:2013:port:a-gole:testbed:232">
    <!-- User friendly and displayable name for the STP object. -->
    <name>dlp01 VLAN 1799</name>

    <!-- This bidirectional port is composed of two individual
         unidirectional port objects. -->
    <Port id="urn:ogf:network:netherlight.net:2013:port:a-gole:testbed:232:out"/>
    <Port id="urn:ogf:network:netherlight.net:2013:port:a-gole:testbed:232:in"/>
</BidirectionalPort>
```

The BidirectionalPort id is used as the root for the bidirectional STP identifier, but we must also navigate to the unidirectional Port/PortGroup definitions to determine any labels being used. The following bidirectional STP identifier would be created based on the unidirectional Port specification:

urn:ogf:network:netherlight.net:2013:port:a-gole:testbed:232?vlan=1799

NSI CS Services

```
<!-- NSI CS Services offered by this Network. -->
<sd:ServiceDefinition id="urn:ogf:network:netherlight.net:2013:ServiceDefinition:EVTS.A-GOLE">
    <name>GLIF Automated GOLE Ethernet VLAN Transfer Service</name>
    <serviceType>http://services.ogf.org/nsi/2013/07/definitions/EVTS.A-GOLE</serviceType>
    <schema xmlns="" name="p2ps" required="true"
            namespace="http://schemas.ogf.org/nsi/2013/07/services/point2point"
            type="{http://schemas.ogf.org/nsi/2013/07/services/point2point}P2PServiceBaseType">
        ...
    </schema>
    ...
</sd:ServiceDefinition>

<sd:ServiceDefinition id="urn:ogf:network:netherlight.net:2013:ServiceDefinition:EPL.A-GOLE">
    <name>GLIF Automated GOLE Ethernet Private Line Service</name>
    <serviceType>http://services.ogf.org/nsi/2013/07/definitions/EPL.A-GOLE</serviceType>
</sd:ServiceDefinition>

<sd:ServiceDefinition id="urn:ogf:network:netherlight.net:2013:ServiceDefinition:EPL-to-802.1Q">
    <name>Service adaptation from EPL-to-802.1Q</name>
    <serviceType>http://services.ogf.org/nsi/2013/07/definitions/EPL-to-802.1Q</serviceType>
</sd:ServiceDefinition>
```

Full service definition can be included.

Reference to Service Definition type

Adaptation Service Definition type

NSI Service Domain

The SwitchingService element is used to model NSI Service Domains. A single SwitchingService declaration can expand into many NSI Service Domains depending if label swapping is supported. Inbound and outbound Port/PortGroup elements are specified as members.

```
<!-- We define a hasService relationship to hold our definitions of
     NSI Service Domains and Adaptations. -->
<Relation type="http://schemas.ogf.org/nml/2013/05/base#hasService">
    <!-- We use the SwitchingService element to define NSI Service Domains
         for the A-GOLE-EPL service. -->
    <SwitchingService id="urn:ogf:network:netherlight.net:2013:ServiceDomain:A-GOLE-EPL"
        encoding="http://schemas.ogf.org/nml/2013/05/ethernet">

        <!-- The standard EPL ports. -->
        <Relation type="http://schemas.ogf.org/nml/2013/05/base#hasInboundPort">
            <Port id="urn:ogf:network:netherlight.net:2013:port:a-gole-epl:testbed:666:in"/>
            <Port id="urn:ogf:network:netherlight.net:2013:port:a-gole-epl:testbed:668:in"/>
        </Relation>

        <nml:Relation type="http://schemas.ogf.org/nml/2013/05/base#hasOutboundPort">
            <Port id="urn:ogf:network:netherlight.net:2013:port:a-gole-epl:testbed:666:out"/>
            <Port id="urn:ogf:network:netherlight.net:2013:port:a-gole-epl:testbed:668:out"/>
        </nml:Relation>

        <!-- NSI CS Services supported by this Service Domain. -->
        <sd:ServiceDefinition id="urn:ogf:network:netherlight.net:2013:ServiceDefinition:EPL.A-GOLE"/>
    </SwitchingService>
</Relation>
```

The ServiceDefinition element identifies the services supported in this Service Domain.

Label Swapping

The SwitchingService supports the concept of label swapping. If the labelSwapping attribute is set to true then any port within the SwitchingService can be connected to any other port independent of label. If set to false, then only ports with equivalent labels can be interconnected.

```
<SwitchingService id="urn:ogf:network:netherlight.net:2013:ServiceDomain:A-GOLE-EVTS"
    encoding="http://schemas.ogf.org/nml/2013/05/ethernet"
    labelSwapping = "true"
    labelType="http://schemas.ogf.org/nml/2013/05/ethernet#vlan">

    <!-- Port relations have to be specified separately from PortGroups as defined
         in the NML schema. -->
    <Relation type="http://schemas.ogf.org/nml/2013/05/base#hasInboundPort">
        <Port id="urn:ogf:network:netherlight.net:2013:port:a-gole:testbed:232:in"/>
    </Relation>

    <nml:Relation type="http://schemas.ogf.org/nml/2013/05/base#hasOutboundPort">
        <Port id="urn:ogf:network:netherlight.net:2013:port:a-gole:testbed:232:out"/>
    </nml:Relation>

    <Relation type="http://schemas.ogf.org/nml/2013/05/base#hasInboundPort">
        <PortGroup id="urn:ogf:network:netherlight.net:2013:port:a-gole:testbed:241:in"/>
        <PortGroup id="urn:ogf:network:netherlight.net:2013:port:a-gole:testbed:manlan:1:in"/>
    </Relation>

    <nml:Relation type="http://schemas.ogf.org/nml/2013/05/base#hasOutboundPort">
        <PortGroup id="urn:ogf:network:netherlight.net:2013:port:a-gole:testbed:241:out"/>
        <PortGroup id="urn:ogf:network:netherlight.net:2013:port:a-gole:testbed:manlan:1:out"/>
    </nml:Relation>

    <!-- NSI CS Services supported by this Service Domain. -->
    <sd:ServiceDefinition id="urn:ogf:network:netherlight.net:2013:ServiceDefinition:EVTS.A-GOLE"/>
</SwitchingService>
```

labelSwapping set to “true” indicates that ports can be connected with different label values.

labelType indicates the label that could be switched.

Port elements can be members of SwitchingService.

PortGroup elements can also be members of SwitchingService.

Adaptation Port Definition

Adaptation is port based in NML, so to model an adaptation between NSI Service Domains, we must create special STP for referencing within the Adaptation. In our specific case, we create four unidirectional ports per adaptation. The outbound port from the source Service Domain #1 references the AdaptationService, and the AdaptationService references the inbound port of the destination Service Domain #2. Similarly, for deadaptation the outbound port in Service Domain #2 references the DeadaptationService, and the DeadaptationService references the inbound port on Service Domain #1.

```
<Relation type="http://schemas.ogf.org/nml/2013/05/base#hasOutboundPort">
    <!-- This port is on the sending end of the EPL-to-802.1Q adaptation
         within the EPL Service Domain. -->
    <Port id="urn:ogf:network:netherlight.net:2013:adaptation:a-gole:testbed:EPL-to-802.1Q:out">
        <Relation type="http://schemas.ogf.org/nml/2013/05/base#hasService">
            <AdaptationService id="urn:ogf:network:netherlight.net:2013:AdaptationService:EPL-to-802.1Q"/>
        </Relation>
    </Port>
</Relation>

<Relation type="http://schemas.ogf.org/nml/2013/05/base#hasInboundPort">
    <!-- This port is on the receiving end of the EPL-to-802.1Q adaptation within the EPL Service Domain. -->
    <Port id="urn:ogf:network:netherlight.net:2013:adaptation:a-gole:testbed:EPL-to-802.1Q:in"/>
</Relation>

<Relation type="http://schemas.ogf.org/nml/2013/05/base#hasOutboundPort">
    <!-- This port is on the sending end of the 802.1Q-to-EPL adaptation
         within the 802.1Q Service Domain. -->
    <Port id="urn:ogf:network:netherlight.net:2013:adaptation:a-gole:testbed:802.1Q-to-EPL:out">
        <Relation type="http://schemas.ogf.org/nml/2013/05/base#hasService">
            <DeadaptationService id="urn:ogf:network:netherlight.net:2013:DeadaptationService:EPL-to-802.1Q"/>
        </Relation>
    </Port>
</Relation>

<Relation type="http://schemas.ogf.org/nml/2013/05/base#hasInboundPort">
    <!-- This port is on the receiving end of the EPL-to-802.1Q
         adaptation within the 802.1Q Service Domain. -->
    <Port id="urn:ogf:network:netherlight.net:2013:adaptation:a-gole:testbed:EPL-to-802.1Q:in"/>
</Relation>
```

Outbound Port elements reference the (De)AdaptationService.

Adaptation Service Definition

Adaptation Id is referenced by source port.

Service Definition for adaptation is referenced by Id.

```
<Relation type="http://schemas.ogf.org/nml/2013/05/base#hasService">
    <!-- The AdaptationService element describes a unidirectional adaptation. In this case we are describing an
         adaptation from an EPL to an 802.1Q Trunk service where we take input Ethernet frames, encapsulate
         them in an 802.1AH header, and allocate an S-TAG for transfer over a switch 802.1Q Trunk service. We
         place the Service Definition id in the adaptationFunction attribute to allow for adaptation details
         to be discovered. -->
    <nml:AdaptationService id="urn:ogf:network:netherlight.net:2013:AdaptationService:a-gole:testbed:EPL-to-802.1Q"
            adaptationFunction="urn:ogf:network:netherlight.net:2013:ServiceDefinition:EPL-to-802.1Q">
        <name>EPL-to-802.1Q Adaptation</name>
        <nml:Relation type="http://schemas.ogf.org/nml/2013/05/base#providesPort">
            <nml:Port id="urn:ogf:network:netherlight.net:2013:adaptation:a-gole:testbed:EPL-to-802.1Q:in"/>
        </nml:Relation>
    </nml:AdaptationService>

    <!-- The DeadaptationService element describes a unidirectional deadaptation. In this case we are describing
         a deadaptation from an 802.1Q Trunk to an EPL service where we are removing an 802.1AH header from
         original Ethernet fame. We place the Service Definition id in the adaptationFunction attribute to
         allow for deadaptation details to be discovered. -->
    <nml:DeadaptationService id="urn:ogf:network:netherlight.net:2013:AdaptationService:a-gole:testbed:802.1Q-to-EPL"
            adaptationFunction="urn:ogf:network:netherlight.net:2013:ServiceDefinition:EPL-to-802.1Q">
        <name>802.1Q-to-EPL Adaptation</name>
        <nml:Relation type="http://schemas.ogf.org/nml/2013/05/base#providesPort">
            <nml:Port id="urn:ogf:network:netherlight.net:2013:adaptation:a-gole:testbed:802.1Q-to-EPL:in"/>
        </nml:Relation>
    </nml:DeadaptationService>
</Relation>
```

Target port of adaptation.

Target port of deadaptation.

Service Definition for deadaptation must be same reference Id as adaptation.