

# NSI-CS Service Decoupling

## Status of This Document

Grid Working Document (GWD)

## Document Change History

Date	Change
June 28, 2013	Initial draft of service decoupling proposal.
July 5, 2013	Added serviceType element to identify Service Description template.
July 7, 2013	Removed serviceAttributes element and included external namespace service elements in body of criteria element.
July 23, 2013	Added decoupled service error proposal and reservation modification handling.
July 26, 2013	Revised decoupled service error proposal based on feedback from review.
July 30, 2013	Added EVTS service and state of Service Description section.

## Copyright Notice

Copyright © Open Grid Forum (2008-2013). Some Rights Reserved. Distribution is unlimited.

## Abstract

In NSI CS version 2.0 focused was placed on a number of improvements, including two-phase reserve/modify, better support for service activation, more complete state machine definition, and clearing up ambiguities around STP definition. All these improvements were an important step along the path to deploying production services, however, little focus was actually placed on the definition of the service itself. This proposal discusses the current service offering supported by NSI CS version 2.0, and a modification to allow for more flexibility in the protocol, allowing for multiple new services to be offered without the need to update the existing base NSI CS protocol. In addition, we resurrect the Service Description concept and bring it fully into the protocol specification.

## Contents

Abstract .....	1
Contents .....	1
1 Introduction .....	3
2 Current NSI-CS service mechanism .....	3
3 Decoupled services.....	5
3.1 Using XML schema ANY element.....	5
3.2 Restructuring <i>criteria</i> element.....	5
3.3 The <i>serviceType</i> element.....	6
3.4 Service specific schema .....	7
3.5 Reservation modification.....	10
3.6 Service error handling.....	10
4 Service Description .....	13
4.1 Element: serviceDescription .....	13
4.2 Element: comment .....	14
4.3 Element: serviceType .....	15
4.4 Element: schema .....	15

4.5	Element: parameter .....	16
4.6	Element: attribute .....	17
4.7	Element: error .....	17
5	Summary .....	18
6	Contributors .....	18
7	Acknowledgments .....	18
8	Intellectual Property Statement .....	18
9	Disclaimer .....	18
10	Full Copyright Notice .....	18
11	Appendix – NSI-CS Point-to-Point Service Schema .....	19
12	Appendix – NSI-CS Service Types Schema .....	22
13	Appendix – NSI-CS Framework Types Schema .....	24
14	Appendix - Service Description schema .....	28
15	Appendix – Service Description example – EVTS.A-GOLE .....	32
16	References .....	36

## 1 Introduction

Discuss the advantages of service decoupling from the core NSI-CS protocol.

## 2 Current NSI-CS service mechanism

In NSI CS version 1.x and 2.0 only unidirectional and bidirectional point-to-point services are offered as part of the protocol. In fact, the NSI working group violated most common protocol design principles by hard coding the service offering directly into the core protocol specification, when in fact, the NSI CS protocol was meant to support many service types. We will now discuss where we have made these easy to correct mistakes.

The base NSI CS *reserve* operation, as shown in Figure 1 below, is extremely generic from a service point of view, containing only base identification and descriptive information. The specifics of the service reservation request are held within the criteria element.

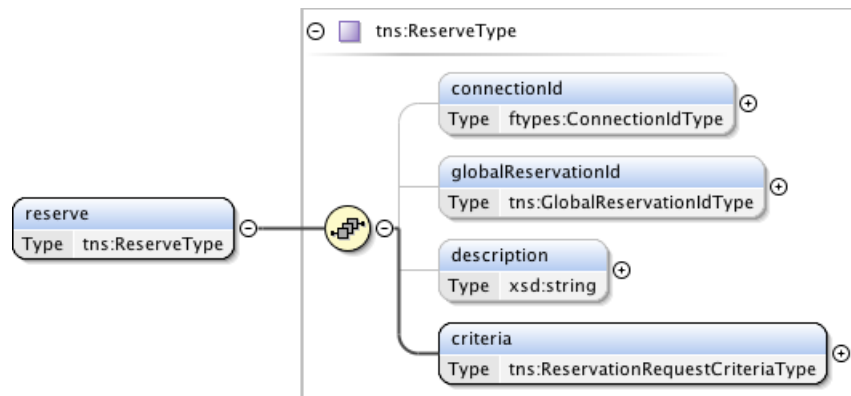


Figure 1 - Reserve request operation.

Exploding the criteria element in Figure 2 we can see the details of the service request. The **version** attribute represents the version of the reservation and is not related to the type of service being requested. The **schedule** element contains the reservation start and end time criteria, providing a generic scheduling capability independent of the service type being offered. The **serviceAttributes** element is also a generic structure allowing for the specification of parameters applying to the service reservation. At the moment, the **serviceAttributes** allows for generic type/value strings, or inclusion of elements from external namespaces. Where we run into a problem is with the **bandwidth** and **path** criteria elements.

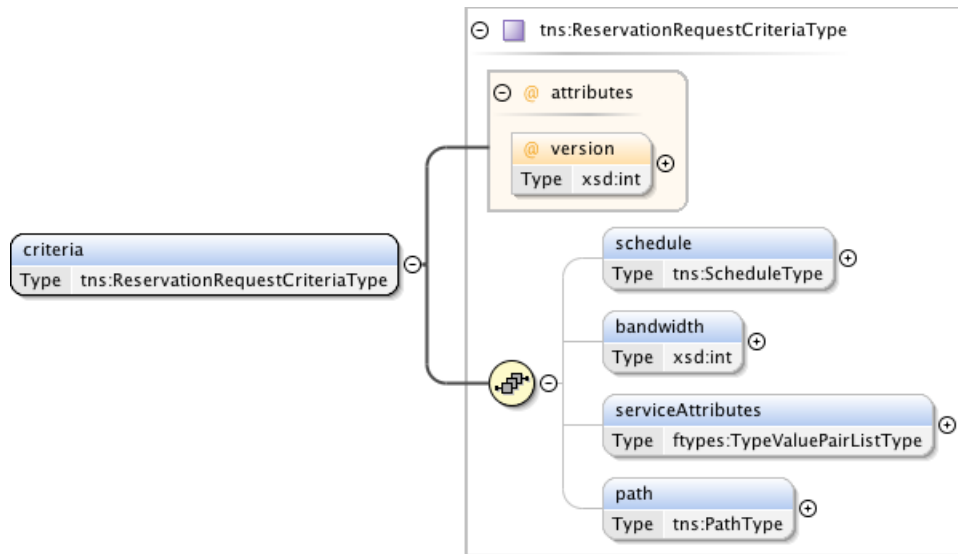


Figure 2 – Service criteria element.

As currently defined, **bandwidth** and **path** restrict a reservation request to either a unidirectional or symmetric bidirectional point-to-point service. Figure 3 shows details of the reservation **path** object.

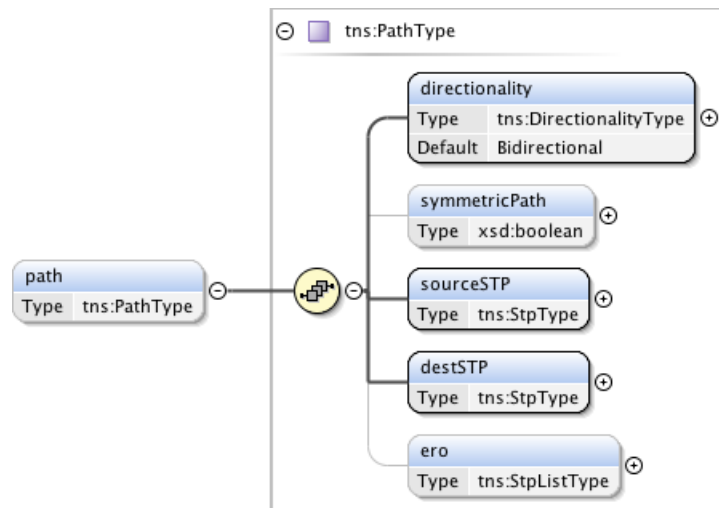


Figure 3 – Service path element.

It should be noted that the **symmetricPath** element applies to only the bidirectional service specification, and indicates whether unidirectional components of the bidirectional service must follow the same path (true) or are free to take diverse paths (false).<sup>1</sup>

As we can see from the current definition, any time we would like to add new services, or extend/correct an issue in an existing service, we need to modify the core NSI CS protocol

<sup>1</sup> Do not confuse this attribute with bandwidth symmetry as only symmetric bidirectional services are supported.

definition. This is an extremely expensive proposition for NSI as a standard body, as well as NSA implementations for the churn in the core protocol. Unfortunately, we have built what could be termed a brittle protocol with respect to offered services.

### 3 Decoupled services

Our primary goal should be to remove the dependencies of service specification from the core NSI CS protocol. This will allow the existing NSI CS protocol to remain stable while permitting changes to the services offered by NSA within the network. Fortunately, with XML and a small change to the structure of our existing NSI CS protocol we can achieve this exact behavior.

#### 3.1 Using XML schema ANY element

The key to this change is the use the XML **ANY** mechanism similar to the **serviceAttributes** element within the reservation criteria. This element has already been used by some NSA implementations to specify additional service parameters within their client requests. For example, SURFnet uses these parameters to allow customers to request Layer 1 sub network protection for their NSI-CS services. The following XML snippet would appear in the **criteria** element of an NSI CS v2.0 reservation request:<sup>2</sup>

```
<serviceAttributes>
  <surf:sNCP xmlns:surf="http://schemas.surfnet.nl/nsi/2013/04/services">Protected</surf:sNCP>
</serviceAttributes>
```

In this case, we are taking advantage of the flexible **ANY** element feature within XML that permits inclusion of elements from external namespaces, or in this specific case, we are seeing an element called “sNCP” from the SURFnet schema namespace. The SURFnet NSA knows how to interpret elements from their services namespace, and therefore, can provide additional capabilities outside the core NSI CS protocol. No extensions to the core NSI protocol were required.

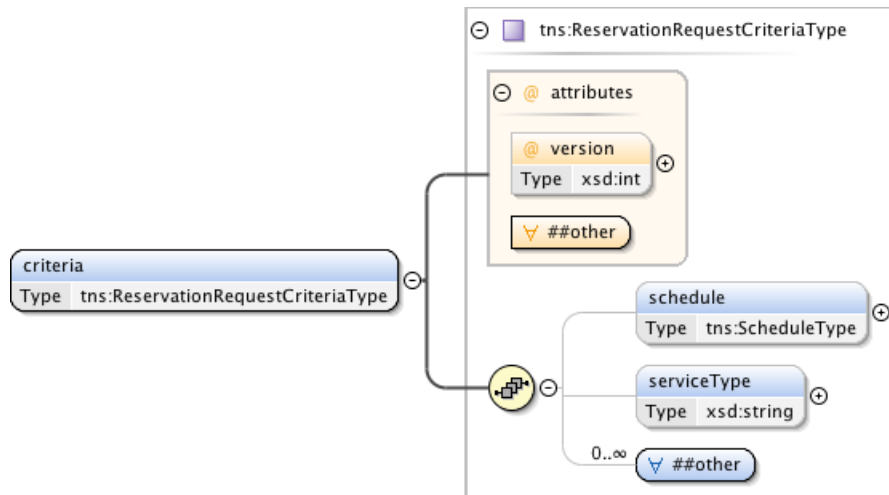
#### 3.2 Restructuring **criteria** element

The proposal is to remove the point-to-point specific **bandwidth**, **path** elements from the **criteria** element and place them into a separate service specific schema definition with its own dedicated namespace, and add an **ANY** child element to the **criteria** element to allow generic inclusion of external service specifications. With this new **ANY** child element we also remove the need for the **serviceAttributes** element as the capability is incorporated into the base **criteria** element. In addition to the service specification decoupling, we introduce a new string element called **serviceType**, which will be described in the next section.

Incorporating these described changes, the new **criteria** element would be simplified to what is now shown below in Figure 4.

---

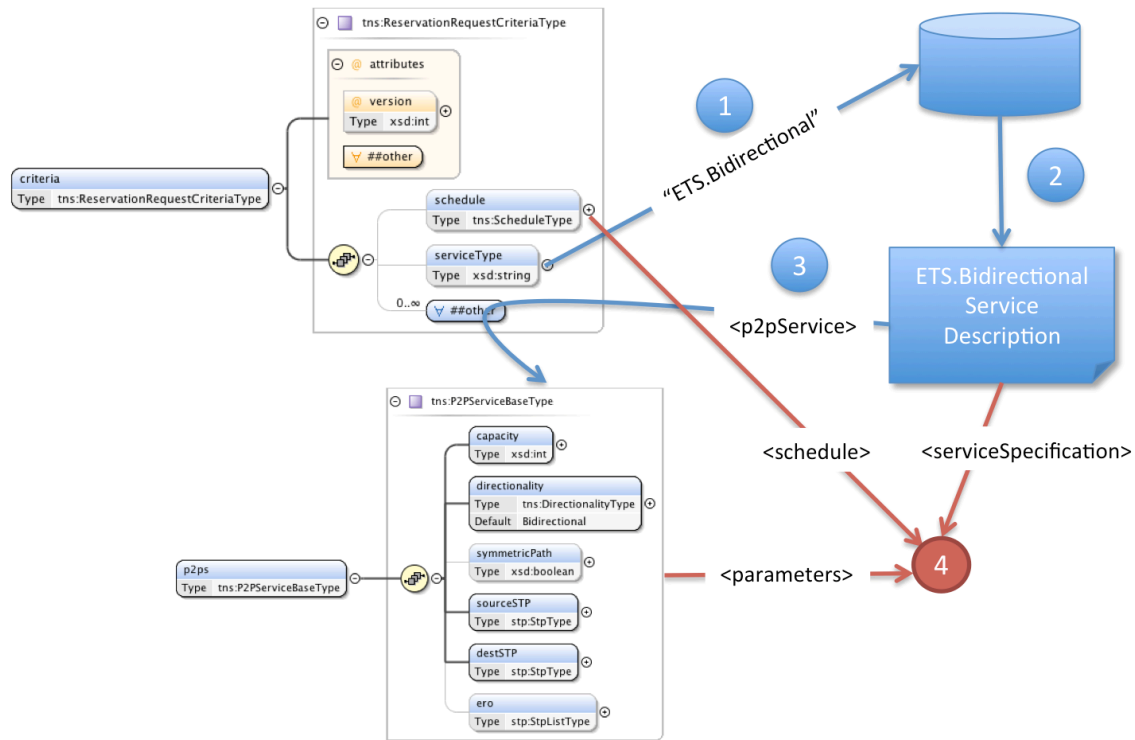
<sup>2</sup> This assumes the discussed simplification of the **ServiceAttributesType**.

Figure 4 – New reservation *criteria* element.

### 3.3 The *serviceType* element

The *serviceType* element will relay the specific service type being requested in the reservation. This service type string maps to a specific Service Description template defined by the network providers describing the type of service offered, parameters supported in a reservation request (mandatory and optional), defaults for parameters if not specified (as well as maximums and minimums), and other attributes relating to the service offering. The NSA in turn uses this information to determine the specific service parameters carried in the *criteria* element required to specify the requested service.

The Service Description template is an important component in the solution, linking the opaque information carried in the NSI CS protocol to the concrete parameters needed to satisfy a specific service request. Figure 5 is a pictorial view of how the *serviceType* maps through the Service Description template, identifying the mandatory and optional service elements needed to satisfy the request.



1. Extract **serviceType** value from incoming reservation request and lookup Service Description corresponding to **serviceType**.
2. Use Service Description to determine the service elements needed for the specific service requested and any other service related parameters.
3. Extract specific services elements from **criteria** as described in Service Description.
4. Process service request using supplied service parameters and service template information.

Figure 5 – Using **serviceType** to determine required and optional service elements.

### 3.4 Service specific schema

An interesting side effect of this service decoupling is the core NSI-CS protocol specification is simplified with the removal of the service specific types that now go into a new service schema. Figure 6 shows a repackaged **p2ps** element that will be included in **criteria** when a generic point-to-point service is requested (notice **bandwidth** has been renamed to **capacity**).

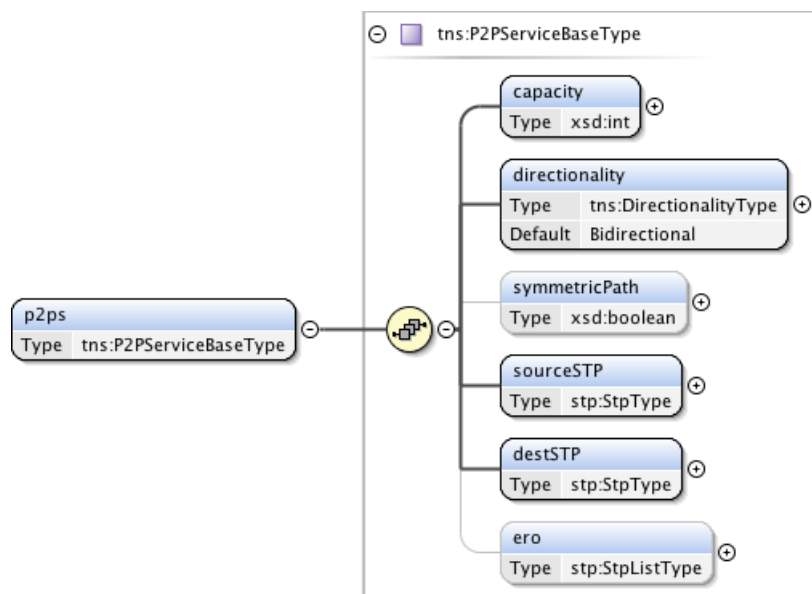


Figure 6 – A point-to-point service definition.

Below we see an example *reserve* request XML message for a bidirectional service as currently defined in NSI CS version 2.0. Notice the ***bandwidth*** and ***path*** elements are members of the ***criteria*** element. Also notice that ***serviceAttributes*** contains the SURFnet specific ***sNCP*** element as an example of including from an external namespace.

```
<nsi:reserve
  xmlns:nsi="http://schemas.ogf.org/nsi/2013/04/connection/types"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:surf="http://schemas.surfnet.nl/nsi/2013/04/services">
  <connectionId>urn:uuid:4b4a71d0-3c71-47cf-a646-beacb14a4c72</connectionId>
  <globalReservationId>urn:uuid:83fe4f36-5b38-41b6-bc46-a362a06a54ee</globalReservationId>
  <description> My example reservation with existing NSI CS 2.0 schema</description>
  <criteria version="1">
    <schedule>
      <startTime>2013-09-30T09:30:10Z</startTime>
      <endTime>2013-09-30T10:30:10Z</endTime>
    </schedule>
    <bandwidth>1000</bandwidth>
    <serviceAttributes>
      <attribute>
        <surf:sNCP>Protected</surf:sNCP>
      </attribute>
    </serviceAttributes>
    <path>
      <directionality>Bidirectional</directionality>
      <symmetricPath>true</symmetricPath>
      <sourceSTP>
        <networkId>urn:ogf:network:netherlight.net:2012</networkId>
        <localId> urn:ogf:network:netherlight.net:2012:uvalight-netherlight</localId>
        <labels>
          <attribute type="vlan">
            <value>1901</value>
          </attribute>
        </labels>
      </sourceSTP>
    </path>
  </criteria>
</nsi:reserve>
```



```

    </labels>
  </sourceSTP>
  <destSTP>
    <networkId>urn:ogf:network:netherlight.net:2012</networkId>
    <localId> urn:ogf:network:netherlight.net:2012:netherlight-czechlight</localId>
    <labels>
      <attribute type="vlan">
        <value>1901</value>
      </attribute>
    </labels>
  </destSTP>
</path>
</criteria>
</nsi:reserve>

```

Below we have the proposed restructured *reserve* request XML message:

```

<nsi:reserve xmlns:nsi="http://schemas.ogf.org/nsi/2013/04/connection/types"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:p2p="http://schemas.ogf.org/nsi/2013/04/services/point2point"
  xmlns:surf="http://schemas.surfnet.nl/nsi/2013/04/services">
  <connectionId>urn:uuid:4b4a71d0-3c71-47cf-a646-beacb14a4c72</connectionId>
  <globalReservationId>urn:uuid:83fe4f36-5b38-41b6-bc46-a362a06a54ee</globalReservationId>
  <description>My example reservation using decoupled service schema</description>
  <criteria version="1">
    <schedule>
      <startTime>2013-09-30T09:30:10Z</startTime>
      <endTime>2013-09-30T10:30:10Z</endTime>
    </schedule>
    <p2p:p2ps>
      <capacity>1000</capacity>
      <directionality>Bidirectional</directionality>
      <symmetricPath>true</symmetricPath>
      <sourceSTP>
        <networkId>urn:ogf:network:netherlight.net:2012</networkId>
        <localId> urn:ogf:network:netherlight.net:2012:uvalight-netherlight</localId>
        <labels>
          <attribute type="vlan">
            <value>1901</value>
          </attribute>
        </labels>
      </sourceSTP>
      <destSTP>
        <networkId>urn:ogf:network:netherlight.net:2012</networkId>
        <localId> urn:ogf:network:netherlight.net:2012:netherlight-czechlight</localId>
        <labels>
          <attribute type="vlan">
            <value>1901</value>
          </attribute>
        </labels>
      </destSTP>
    </p2p:p2ps>
    <surf:sNCP>Protected</surf:sNCP>
  </criteria>
</nsi:reserve>

```

These messages look quite similar, but there are some key differences:

- The **p2p** namespace is defined in the **reserve** element using a unique URL defining the service XSD document. We have encapsulated all types needed for this service in that XSD document.
 

```
xmlns:p2p="http://schemas.ogf.org/nsi/2013/04/services/point2point"
```
- The **capacity** and **path** elements members are now part of the **p2ps** element included within the **criteria** element. These attributes are part of the service specification, and therefore, separate from the core reservation criteria. We have effectively decoupled the attributes of a service from the core NSI protocol.
- The **serviceType** element is added to identify the desired service requested and will identify the specific service elements carried in **criteria**.
- The **serviceAttributes** element was removed and the ANY functionality placed directly into the criteria element to simplify specification of these external namespace elements.
- Multiple service attributes can be specified as show by inclusion of both the **p2ps** and SURFnet specific **sNCP** elements.

In addition to these changes, the restructuring of the **criteria** element would be done in the *reserveConfirmed* and *query* messages.

### 3.5 Reservation modification

For a base point-to-point service specification we support the modification of **schedule** (start or end time), as well as the **capacity** of the service. The **schedule** element is within the core **criteria** element, and remains as is, specifying a change in the combination of **startTime** and **endTime** as desired. For the external service schema, only the elements to be modified are included in the request. These will be defined as separate elements within their schema definition for inclusion as modifiable items.

Below is an example *reserve* modification request XML message where we are requesting a modification to the **capacity** parameter of the reservation:

```
<nsi:reserve xmlns:nsi="http://schemas.ogf.org/nsi/2013/04/connection/types"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:p2p="http://schemas.ogf.org/nsi/2013/04/services/point2point"
  xmlns:surf="http://schemas.surfnet.nl/nsi/2013/04/services">
  <connectionId>urn:uuid:4b4a71d0-3c71-47cf-a646-beacb14a4c72</connectionId>
  <criteria version="2">
    <p2p:capacity>500</p2p:capacity>
  </criteria>
</nsi:reserve>
```

### 3.6 Service error handling

The NSI-CS protocol utilizes the **ServiceExceptionType** structure to commonly convey error information in both *SOAP faults* and operation *Failed* messages. The structure is extremely

flexible and able to handle both simple high-level error information, as well as detailed errors down to the individual attribute value causing a problem. The current **ServiceExceptionType** is shown in Figure 7.

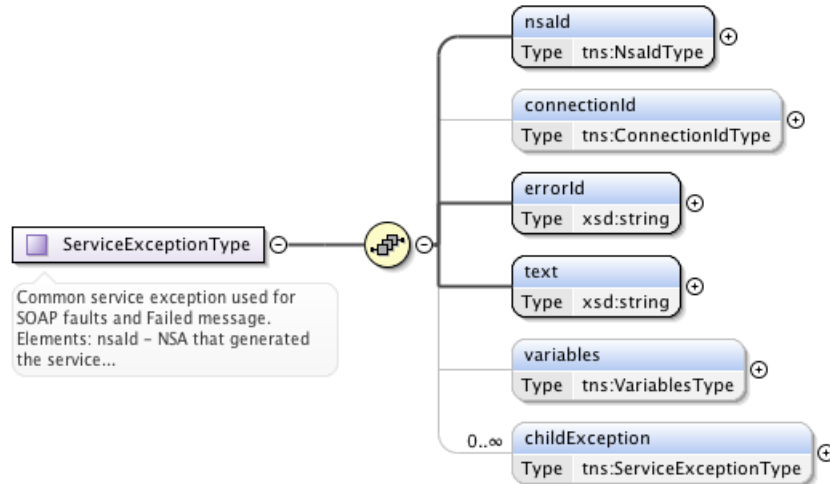


Figure 7 – Current **ServiceExceptionType** definition.

Parameter	M/O	Description
nsald	M	NSA that generated the service exception.
connectionId	O	The connectionId associated with the reservation impacted by this error.
errorId	M	Error identifier uniquely identifying each known fault within the protocol.
text	M	User-friendly message text describing the error.
variables	O	An optional collection of type/value pairs providing additional information relating to the error.
childException	O	Hierarchical list of service exceptions capturing failures within the request tree.

NSI-CS uses a hierarchal error code structure to group related error codes together under a common parent error code value. For example, the following table demonstrates the generic **TOPOLOGY\_ERROR** parent code and four more specific children topology related error codes.

text	errorId	Description
TOPOLOGY_ERROR	00400	Parent error classification.
UNKNOWN_STP	00401	Could not find STP in topology database.
STP_RESOLUTION_ERROR	00402	Could not resolve STP to a managing NSA.
NO_PATH_FOUND	00403	Path computation failed to resolve route for reservation.
VLANID_INTERCHANGE_NOT_SUPPORTED	00404	VLAN interchange not supported for requested path.

Table 1 – Topology related error codes.

Below we see an example XML segment for the **serviceException** element generated when an invalid STP was specified in the **sourceSTP** element of the reserve request operation.

```

<serviceException>
  <nsald>urn:ogf:network:netherlight.net:2012:nsa</nsald>
  <connectionId>urn:uuid:4b4a71d0-3c71-47cf-a646-beacb14a4c72</connectionId >
  <errorId>00401</errorId>
  <text>UNKNOWN_STP</text>
  <variables>

```

```

<variable type="sourceSTP.localId">
  <value>urn:ogf:network:netherlight.net:2012:stp:ams-uva-83</value>
</variable>
</variables>
</serviceException>

```

In this example we see the **nsald** of the NSA generating the error, the **connectionId** associated with the error, the **errorId** and **text** for an unknown STP, and the **variables** section that contains the parameter causing the issue.

Unfortunately, as can be seen with the **UNKNOWN\_STP**, **STP\_RESOLUTION\_ERROR**, and **VLANID\_INTERCHANGE\_NOT\_SUPPORTED** error codes, we have service specific error information in the current error-handling framework. The concept of STP and VLANID are currently specific to the point-to-point Ethernet service specification. As new services are offered, and existing ones modified, these errors will need to be modified or extended. This is not a desirable feature of the existing error handling strategy. For this reason we need to provide a decoupled service specific error solution.

In Figure 8 we provide a solution to the problem by extending the existing **ServiceExceptionType** to include an optional **serviceType** element.

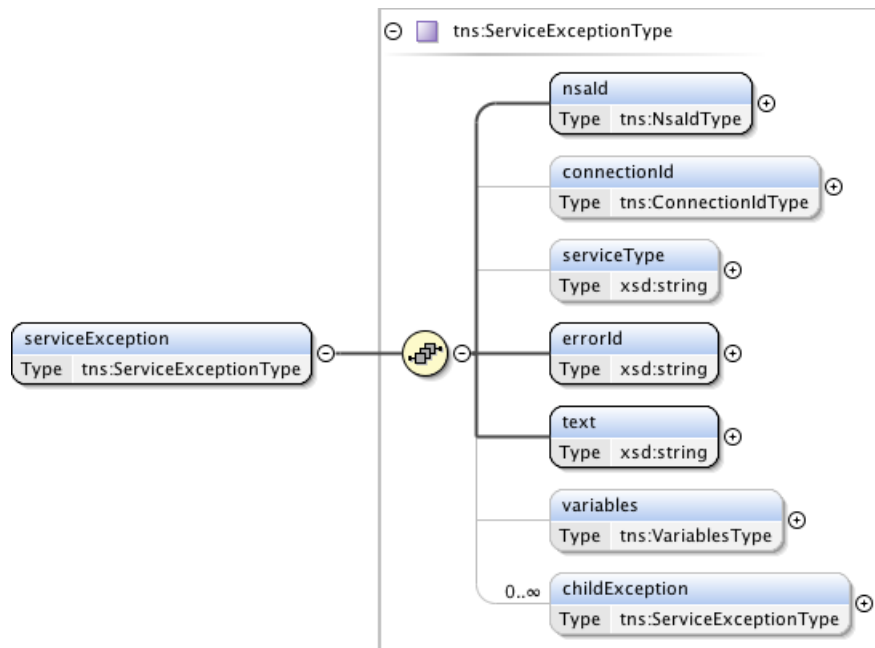


Figure 8 – Service decoupled error **ServiceExceptionType** definition.

The **serviceType** element is only included when there is a service specific error generated by an NSA, with the **serviceType** element mapping to the service description used for the service request<sup>3</sup> on the failed segment. We also modify the existing error codes to remove service specific values, and introduce a generic **SERVICE\_ERROR** parent code to indicate the existence of a service specific error. New service specific errors are then numbered as a child of this **SERVICE\_ERROR** parent code, and interpreted in the context of the **serviceType** returned in the service exception. Table 2 shows the service specific errors defined for the basic point-to-point service.

<sup>3</sup> The **serviceType** is included since the original **serviceType** specified in the *reserve* request may have been re-mapped into a different **serviceType** when sent to a child NSA.

text	errorId	Description
SERVICE_ERROR	00700	Parent error classification for a service specific error.
UNKNOWN_STP	00701	Could not find STP in topology database.
STP_RESOLUTION_ERROR	00702	Could not resolve STP to a managing NSA.
VLANID_INTERCHANGE_NOT_SUPPORTED	00703	VLAN interchange not supported for requested path.
STP_UNAVAILABLE	00704	Specified STP already in use.
BANDWIDTH_UNAVAILABLE	00705	Insufficient bandwidth available for reservation.

Table 2 – NSI-CS point-to-point service specific errors.

When new services are defined three pieces of information must now be generated:

1. Service specific XML schema if not reusing an existing schema.
2. Service description document.
3. Service error code definitions.

Using the previous service error example we would now repackage a service specific error as follows:

```

<serviceException>
  <nsald>urn:ogf:network:netherlight.net:2012:nsa</nsald>
  <connectionId>urn:uuid:4b4a71d0-3c71-47cf-a646-beacb14a4c72</connectionId >
  <serviceType>ETS.SURFnet</serviceType>
  <errorId>00701</errorId>
  <text>UNKNOWN_STP</text>
  <variables>
    <variable type="sourceSTP">
      <value>urn:ogf:network:netherlight.net:2012:stp:ams-uva-83</value>
    </variable>
  </variables>
</serviceException>

```

## 4 Service Description

Intro text...

### 4.1 Element: serviceDescription

This element is for representing a Service Description XML document. The associated **ServiceDescriptionType** defines the structure and content of a Service Description document.

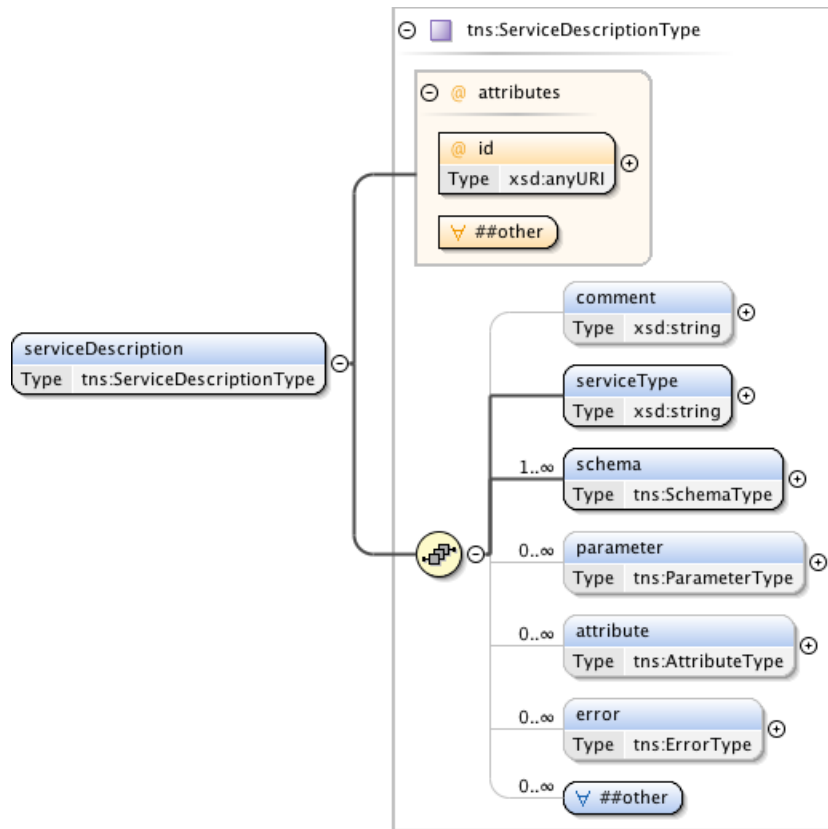
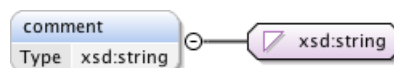


Figure 9 – serviceDescription element.

Name	M/O	Description
id	M	A service description must have a unique identifier within the network sourcing the description.
comment	O	A comment describes the basic service offered in this service description. Any differences from the global service description should be identified here.
serviceType	M	This is the standard service type as defined in the associated global service description. We allocate a standard namespace for each template so they are globally unique.
schema	M	This is the schema elements specified in a reservation. There can be multiple schema entries here for services if they requiring multiple schema in a reserve request.
parameter	O	Parameter definitions for the service and their values. These reflect the XML schema definitions and any local range restrictions.
attribute	O	Attribute definitions for the service and their values. Attributes are aspects of the service that are not specified in the XML schema for the service.
error	O	Errors defined for this service.

Table 3 – ServiceDescriptionType attributes and elements.

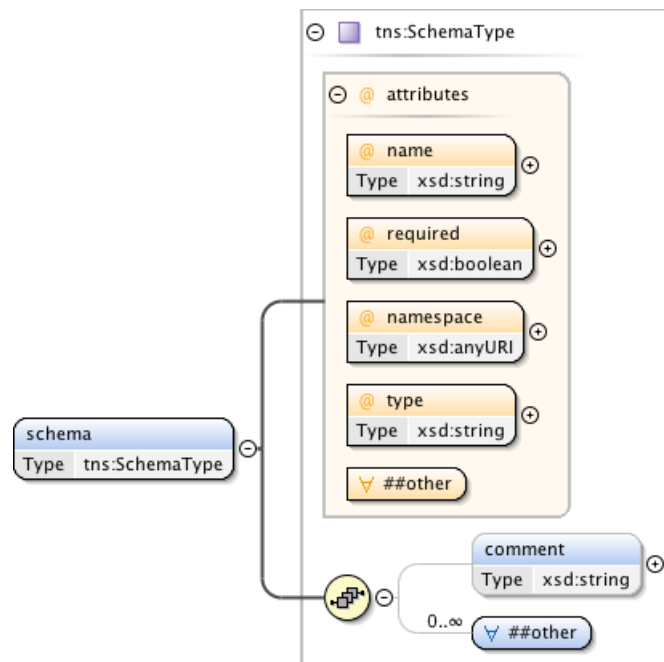
4.2 Element: comment



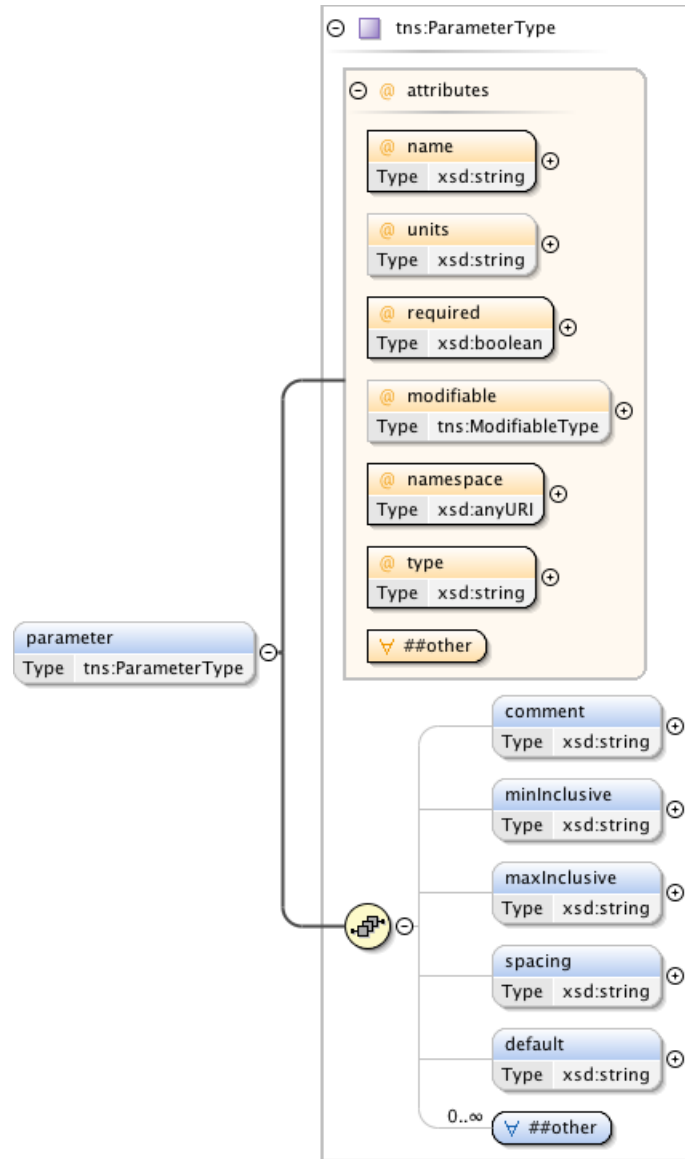
### 4.3 Element: serviceType



### 4.4 Element: schema

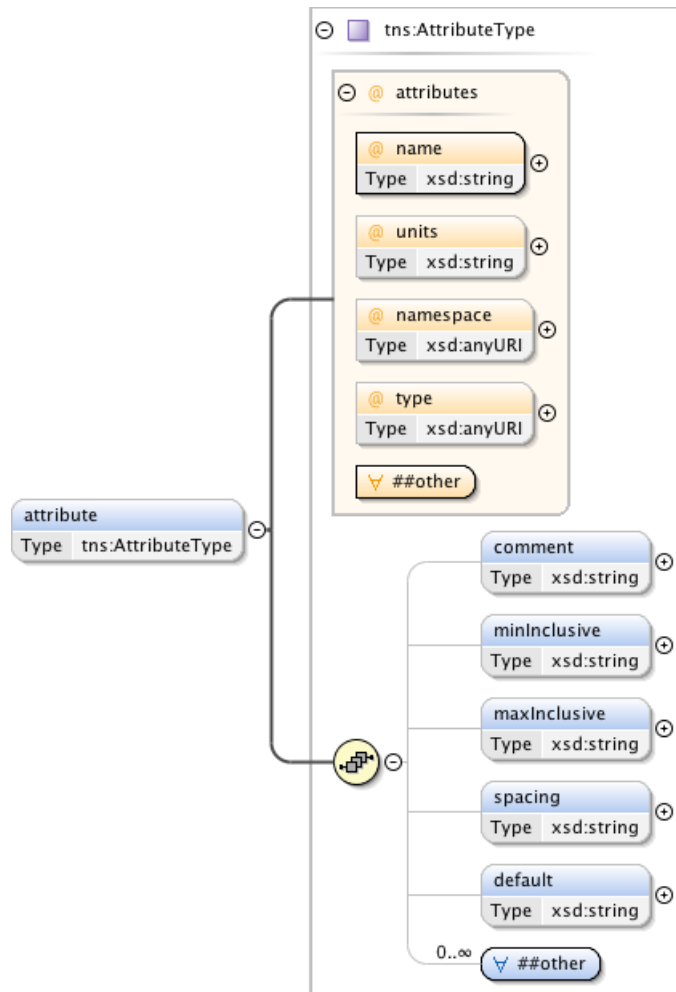


### 4.5 Element: parameter

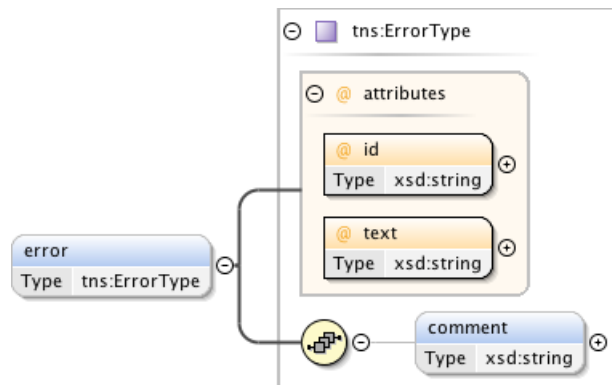




#### 4.6 Element: attribute



#### 4.7 Element: error



## 5 Summary

The proposed decoupling changes are an improvement over the current tight coupling of the point-to-point service definition within the NSI CS version 2.0 protocol specification. Refactoring the service elements into an external namespace and including through an **ANY** definition within the **criteria** element does not change the behavioral aspects of the protocol, nor the information carried in the messages, as this is just a syntactic change to where the data is carried. There will be minimal impact current NSA implementation of NSI CS version 2.0.

## 6 Contributors

The following people contributed to the content of this document:

John MacAuley  
Network Services, SURFnet  
The Netherlands  
Email: john@surfnet.nl

## 7 Acknowledgments

The author would like to thank the color blue for all your wonderful contributions to the world.

## 8 Intellectual Property Statement

The OGF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the OGF Secretariat.

The OGF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights, which may cover technology that may be required to practice this recommendation. Please address the information to the OGF Executive Director.

## 9 Disclaimer

This document and the information contained herein is provided on an "As Is" basis and the OGF disclaims all warranties, express or implied, including but not limited to any warranty that the use of the information herein will not infringe any rights or any implied warranties of merchantability or fitness for a particular purpose.

## 10 Full Copyright Notice

Copyright (C) Open Grid Forum (2008-2013). Some Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included as references to the derived portions on all such copies and derivative works. The published OGF document from which such works are derived, however, may not be modified in any way, such as by removing the copyright notice or

references to the OGF or other organizations, except as needed for the purpose of developing new or updated OGF documents in conformance with the procedures defined in the OGF Document Process, or as required to translate it into languages other than English. OGF, with the approval of its board, may remove this restriction for inclusion of OGF document content for the purpose of producing standards in cooperation with other international standards bodies.

The limited permissions granted above are perpetual and will not be revoked by the OGF or its successors or assignees.

## 11 Appendix – NSI-CS Point-to-Point Service Schema

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
```

```

    The OGF takes no position regarding the validity or scope of any intellectual property
    or other rights that might be claimed to pertain to the implementation or use of the
    technology described in this document or the extent to which any license under such rights
    might or might not be available; neither does it represent that it has made any effort to
    identify any such rights. Copies of claims of rights made available for publication and
    any assurances of licenses to be made available, or the result of an attempt made to
    obtain a general license or permission for the use of such proprietary rights by
    implementers or users of this specification can be obtained from the OGF Secretariat.
```

```

    The OGF invites any interested party to bring to its attention any copyrights, patents
    or patent applications, or other proprietary rights which may cover technology that may be
    required to practice this recommendation. Please address the information to the OGF
    Executive Director.
```

```

    This document and the information contained herein is provided on an "As Is" basis and
    the OGF disclaims all warranties, express or implied, including but not limited to any
    warranty that the use of the information herein will not infringe any rights or any
    implied warranties of merchantability or fitness for a particular purpose.
```

```

    Copyright (C) Open Grid Forum (2009-2012). All Rights Reserved.
```

```

    This document and translations of it may be copied and furnished to others, and
    derivative works that comment on or otherwise explain it or assist in its implementation
    may be prepared, copied, published and distributed, in whole or in part, without
    restriction of any kind, provided that the above copyright notice and this paragraph are
    included on all such copies and derivative works. However, this document itself may not be
    modified in any way, such as by removing the copyright notice or references to the OGF or
    other organizations, except as needed for the purpose of developing Grid Recommendations
    in which case the procedures for copyrights defined in the OGF Document process must be
    followed, or as required to translate it into languages other than English.
```

```

    The limited permissions granted above are perpetual and will not be revoked by the OGF
    or its successors or assignees.
```

```

    Open Grid Forum NSI Connection Services Protocol v2.0 - Point-to-point service
    definition.
```

```

-->
<xsd:schema targetNamespace="http://schemas.ogf.org/nsi/2013/07/services/point2point"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:ftypes="http://schemas.ogf.org/nsi/2013/07/framework/types"
  xmlns:types="http://schemas.ogf.org/nsi/2013/07/services/types"
  xmlns:tns="http://schemas.ogf.org/nsi/2013/07/services/point2point">

  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      This is an XML schema document describing the OGF NSI point-to-point
      service types.
    </xsd:documentation>
  </xsd:annotation>

  <!-- Import the common NSI framework types. -->
  <xsd:import namespace="http://schemas.ogf.org/nsi/2013/07/framework/types"
    schemaLocation="ogf_nsi_framework_types_v2_0.xsd"/>
```

```

<!-- Import the common NSI framework types. -->
<xsd:import namespace="http://schemas.ogf.org/nsi/2013/07/services/types"
  schemaLocation="ogf_nsi_services_types_v2_0.xsd"/>

<!-- Element definitions for services types. -->
<xsd:element name="p2ps" type="tns:P2PServiceBaseType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      This Point-to-Point Service element is used to specify a generic
      point-to-point service request in the NSI CS protocol.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>

<xsd:element name="ets" type="tns:EthernetBaseType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      This Ethernet Transport Service element is used to specify a
      point-to-point Ethernet service request in the NSI CS protocol.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>

<xsd:element name="evts" type="tns:EthernetVlanType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      This Ethernet VLAN Transport Service element is used to specify
      a point-to-point Ethernet VLAN service request in the NSI CS
      protocol.
    </xsd:documentation>
  </xsd:annotation>
</xsd:element>

<!-- These are the modifiable parameters within this service. -->
<xsd:element name="capacity" type="xsd:int" />

<!-- Type definitions for services types. -->
<xsd:complexType name="P2PServiceBaseType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Type defining a generic point-to-point service specification.
      At the moment this type supports a unidirectional or symmetric
      bidirectional service.

      Elements:

      capacity - Capacity of the service in Mb/s.

      directionality - The (uni or bi) directionality of the service.

      symmetricPath - An indication that both directions of a bidirectional
      circuit must follow the same path. Only applicable when
      directionality is "Bidirectional". If not specified then value
      is assumed to be false.

      sourceSTP - Source STP of the service.

      destSTP - Destination STP of the service.

      ero - Hop-by-hop ordered list of STP from sourceSTP to
      destSTP. List does not include sourceSTP and destSTP.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="capacity" type="xsd:int" />
    <xsd:element name="directionality" type="types:DirectionalityType"
      default="Bidirectional" />
    <xsd:element name="symmetricPath" type="xsd:boolean" minOccurs="0" />
    <xsd:element name="sourceSTP" type="types:StpType" />
    <xsd:element name="destSTP" type="types:StpType" />
    <xsd:element name="ero" type="types:StpListType" minOccurs="0" />
  </xsd:sequence>
</xsd:complexType>

```

```

</xsd:sequence>
</xsd:complexType>

<xsd:complexType name="EthernetBaseType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Point-to-Point Ethernet service definition.

      Elements:

      capacity - Capacity of the service in Mb/s including ethernet
      framing headers.

      directionality - The (uni or bi) directionality of the service.

      symmetricPath - An indication that both directions of a bidirectional
      circuit must follow the same path. Only applicable when
      directionality is "Bidirectional". If not specified then value
      is assumed to be false.

      sourceSTP - Source STP of the service.

      destSTP - Destination STP of the service.

      ero - Hop-by-hop ordered list of STP from sourceSTP to
      destSTP. List does not include sourceSTP and destSTP.

      mtu - Specifies the maximum transmission unit size in bits.

      burstsize - Specifies the maximum number of bits that can be
      send to the interface before the sender must wait before
      sending again.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="tns:P2PServiceBaseType">
      <xsd:sequence>
        <xsd:element name="mtu" type="xsd:int" minOccurs="0" />
        <xsd:element name="burstsize" type="xsd:int" minOccurs="0" />
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>

<xsd:complexType name="EthernetVlanType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Point-to-Point Ethernet VLAN service definition.

      Elements:

      capacity - Capacity of the service in Mb/s including ethernet
      framing headers.

      directionality - The (uni or bi) directionality of the service.

      symmetricPath - An indication that both directions of a bidirectional
      circuit must follow the same path. Only applicable when
      directionality is "Bidirectional". If not specified then value
      is assumed to be false.

      sourceSTP - Source STP of the service.

      destSTP - Destination STP of the service.

      ero - Hop-by-hop ordered list of STP from sourceSTP to
      destSTP. List does not include sourceSTP and destSTP.

      mtu - Specifies the maximum transmission unit size in bits.

      burstsize - Specifies the maximum number of bits that can be

```

```

        send to the interface before the sender must wait before
        sending again.

        sourceVLAN - Specifies the VLAN identifier for the source port
        (1 - 4095).

        destVLAN - Specifies the VLAN identifier for the destination port
        (1 - 4095).
    </xsd:documentation>
</xsd:annotation>
<xsd:complexContent>
    <xsd:extension base="tns:EthernetBaseType">
        <xsd:sequence>
            <xsd:element name="sourceVLAN" type="tns:vlanIdType" />
            <xsd:element name="destVLAN" type="tns:vlanIdType" />
        </xsd:sequence>
    </xsd:extension>
</xsd:complexContent>
</xsd:complexType>

<xsd:simpleType name="vlanIdType">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            A simple type modelling a VLAN identifier and restricting range
            to between 1 - 4095.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:int">
        <xsd:minInclusive value="1"/>
        <xsd:maxInclusive value="4095"/>
    </xsd:restriction>
</xsd:simpleType>
</xsd:schema>

```

## 12 Appendix – NSI-CS Service Types Schema

```

<?xml version="1.0" encoding="UTF-8"?>
<!--

```

The OGF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the OGF Secretariat.

The OGF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this recommendation. Please address the information to the OGF Executive Director.

This document and the information contained herein is provided on an "As Is" basis and the OGF disclaims all warranties, express or implied, including but not limited to any warranty that the use of the information herein will not infringe any rights or any implied warranties of merchantability or fitness for a particular purpose.

Copyright (C) Open Grid Forum (2009–2012). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the OGF or other organizations, except as needed for the purpose of developing Grid Recommendations in which case the procedures for copyrights defined in the OGF Document process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the OGF or its successors or assignees.

Open Grid Forum NSI Connection Services Protocol v2.0

```
-->
<xsd:schema targetNamespace="http://schemas.ogf.org/nsi/2013/04/services/types"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:ftypes="http://schemas.ogf.org/nsi/2013/04/framework/types"
  xmlns:tns="http://schemas.ogf.org/nsi/2013/04/services/types">

  <xsd:annotation>
    <xsd:documentation xml:lang="en">
    </xsd:documentation>
  </xsd:annotation>

  <!-- Import the common NSI framework types. -->
  <xsd:import namespace="http://schemas.ogf.org/nsi/2013/04/framework/types"
    schemaLocation="ogf_nsi_framework_types_v2_0.xsd"/>

  <xsd:element name="stp" type="tns:StpType">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">
      </xsd:documentation>
    </xsd:annotation>
  </xsd:element>

  <xsd:element name="stpList" type="tns:StpListType">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">
      </xsd:documentation>
    </xsd:annotation>
  </xsd:element>

  <xsd:complexType name="StpType">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">
        The Service Termination Point (STP) type used for path selection.

        Elements:

        networkId - A globally unique identifier (URN) that identifies the
        Network. Rather than forcing parsing of an STP to determine the
        Network, a separate Network object is defined to allow an
        intermediate NSA to forward the message to the target Network
        without needing to know about the STPs within that domain.

        localId - A locally unique identifier for the STP within the
        associated network.

        labels - Technology specific attributes associated with
        the Service Termination Point.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="networkId" type="xsd:string" />
      <xsd:element name="localId" type="xsd:string" />
      <xsd:element name="labels" type="ftypes:TypeValuePairListType"
minOccurs="0" />
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="StpListType">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">
        A simple ordered list type of Service Termination Point (STP). List
        order is determined by the integer order attribute in the orderedSTP
        element.

        Elements:

        orderedSTP - A list of STP ordered 0..n by their integer order attribute.
      </xsd:documentation>
    </xsd:annotation>
  </xsd:complexType>

```

```

        </xsd:documentation>
      </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="orderedSTP" type="tns:OrderedStpType" minOccurs="0"
maxOccurs="unbounded" />
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="OrderedStpType">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">
        A Service Termination Point (STP) that can be ordered in a list for
        use in PathObject definition.

        Attributes:

        order - Order attribute is provided only when the STP is part of an
        orderedStpList.

        Elements:

        stp - The Service Termination Point (STP).
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="stp" type="tns:StpType" />
    </xsd:sequence>
    <xsd:attribute name="order" type="xsd:int" use="required" />
  </xsd:complexType>

  <xsd:simpleType name="DirectionalityType">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">
        The directionality of the requested data service. Possible values
        are "Bidirectional" for a bidirectional data service, and
        "Unidirectional" for a unidirectional data service.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="Bidirectional" />
      <xsd:enumeration value="Unidirectional" />
    </xsd:restriction>
  </xsd:simpleType>
</xsd:schema>

```

### 13 Appendix – NSI-CS Framework Types Schema

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<!--
```

The OGF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the OGF Secretariat.

The OGF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this recommendation. Please address the information to the OGF Executive Director.

This document and the information contained herein is provided on an "As Is" basis and the OGF disclaims all warranties, express or implied, including but not limited to any warranty that the use of the information herein will not infringe any rights or any implied warranties of merchantability or fitness for a particular purpose.

Copyright (C) Open Grid Forum (2009–2012). All Rights Reserved.



This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the OGF or other organizations, except as needed for the purpose of developing Grid Recommendations in which case the procedures for copyrights defined in the OGF Document process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the OGF or its successors or assignees.

Open Grid Forum NSI Framework v2.0

Description: This is the NSI framework common types schema for the reference web services implementation of the OGF NSI Framework v2.0. Comments and questions can be directed to the mailing list group mailing list (nsi-wg@ogf.org).

-->

```
<xsd:schema targetNamespace="http://schemas.ogf.org/nsi/2013/04/framework/types"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:tns="http://schemas.ogf.org/nsi/2013/04/framework/types">

  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      This is an XML schema document describing the OGF NSI Framework
      common types v2.0.
    </xsd:documentation>
  </xsd:annotation>

  <!-- *****
  *           Complex XML types used in message elements           *
  ***** -->

  <xsd:element name="serviceException" type="tns:ServiceExceptionType" />

  <xsd:complexType name="ServiceExceptionType">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">
        Common service exception used for SOAP faults and Failed
        message.

        Elements:

        nsaId - NSA that generated the service exception.

        connectionId - The connectionId associated with the reservation
        impacted by this error.

        errorId - Error identifier uniquely identifying each known
        fault within the protocol. Acts as a parent functionality
        classification for service specific errors.

        text - User friendly message text describing the error.

        serviceError - An optional service specific error element
        defining the serviceType and service specific error Id. Must
        be included if this error is specific to the service.

        variables - An optional collection of type/value pairs providing
        additional information relating to the error.

        childException - Hierarchical list of service exceptions
        capturing failures within the request tree.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="nsaId" type="tns:NsaIdType" />
      <xsd:element name="connectionId" type="tns:ConnectionIdType" minOccurs="0"
    </xsd:sequence>
  </xsd:complexType>
</xsd:schema>
```

```

        <xsd:element name="errorId" type="xsd:string" />
        <xsd:element name="text" type="xsd:string" />
        <xsd:element name="serviceError" type="tns:ServiceSpecificErrorType"
minOccurs="0" />
        <xsd:element name="variables" type="tns:VariablesType" minOccurs="0" />
        <xsd:element name="childException" type="tns:ServiceExceptionType"
minOccurs="0" maxOccurs="unbounded" />
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="ServiceSpecificErrorType">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            A type definition containing service specific error information.

            Elements:

            serviceType - The service type identifying the applicable
            service description in the context of the NSA generating the
            error.

            serviceErrorId - The service specific error Id.

            serviceErrorText - User friendly message text describing the
            error.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="serviceType" type="xsd:string" />
        <xsd:element name="serviceErrorId" type="xsd:string" />
        <xsd:element name="serviceErrorText" type="xsd:string" />
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="VariablesType">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            A type definition providing a set of zero or more type/value
            variables used for modeling generic attributes.

            Elements:

            variable - The variable containing the type/values.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="variable" type="tns:TypeValuePairType" minOccurs="0"
maxOccurs="unbounded" />
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="TypeValuePairListType">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">
            A holder element providing an attribute list definition for the
            type/value pair.

            Elements:

            attribute - An instance of a type/value pair.
        </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
        <xsd:element name="attribute" type="tns:TypeValuePairType" minOccurs="0"
maxOccurs="unbounded" />
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="TypeValuePairType">
    <xsd:annotation>
        <xsd:documentation xml:lang="en">

```

Definition for a simple type and multi-value tuple. Includes simple string type and value, as well as more advanced extensions if needed. A targetNamespace attribute is included to provide context where needed.

Elements:

value - A string value corresponding to type.

any - Provides a flexible mechanism allowing additional elements to be provided as an alternative, or in combination with value. Use of this element field is beyond the current scope of this NSI specification, but may be used in the future to extend the existing protocol without requiring a schema change.

Attributes:

type - A string representing the name of the type.

targetNamespace - An optional URL to qualify the name space of the capability.

anyAttribute - Provides a flexible mechanism allowing additional attributes non-specified to be provided as needed for peer-to-peer NSA communications. Use of this attribute field is beyond the current scope of this NSI specification, but may be used in the future to extend the existing protocol without requiring a schema change.

```

</xsd:documentation>
</xsd:annotation>
<xsd:sequence>
  <xsd:element name="value" type="xsd:string" nillable="true" minOccurs="0"
maxOccurs="unbounded" />
  <xsd:any namespace="##other" processContents="lax" minOccurs="0"
maxOccurs="unbounded" />
</xsd:sequence>
<xsd:attribute name="type" type="xsd:string" use="optional" />
<xsd:attribute name="targetNamespace" type="xsd:anyURI" use="optional" />
<xsd:anyAttribute namespace="##other" processContents="lax" />
</xsd:complexType>

<!-- *****
*                               XML base types                               *
***** -->

<xsd:simpleType name="ConnectionIdType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A connectionId is a simple string value that uniquely identifies
      a reservation segment within the context of a Provider NSA. This
      value is not globally unique.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:string" />
</xsd:simpleType>

<xsd:simpleType name="DateTimeType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      The time zone support of W3C XML Schema is quite controversial
      and needs some additional constraints to avoid comparison
      problems. These patterns can be kept relatively simple since
      the syntax of the datetime is already checked by the schema
      validator and only simple additional checks need to be added.
      This type definition checks that the time part ends with a "Z"
      or contains a sign.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:dateTime">
    <xsd:pattern value=".+T.+(Z|[+-].+)" />
  </xsd:restriction>
</xsd:simpleType>

```

```

<xsd:simpleType name="NsaIdType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      A specific type for a Network Services Agent (NSA) identifier
      just in case we need to change it in the future. This type
      will be populated with a OGF URN (reference artifact 6478
      "Procedure for Registration of Subnamespace Identifiers in
      the URN:OGF Hierarchy") to be used for compatibility with
      other external systems.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:anyURI" />
</xsd:simpleType>

<xsd:simpleType name="UuidType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Universally Unique Identifier (UUID) URN as per ITU-T Rec.
      X.667 | ISO/IEC 9834-8:2005 and IETF RFC 4122.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:anyURI">
    <xsd:pattern value="urn:uuid:[a-f0-9]{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}"/>
  </xsd:restriction>
</xsd:simpleType>
</xsd:schema>

```

## 14 Appendix - Service Description schema

```

<?xml version="1.0" encoding="UTF-8"?>
<!--

```

The OGF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the OGF Secretariat.

The OGF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this recommendation. Please address the information to the OGF Executive Director.

This document and the information contained herein is provided on an "As Is" basis and the OGF disclaims all warranties, express or implied, including but not limited to any warranty that the use of the information herein will not infringe any rights or any implied warranties of merchantability or fitness for a particular purpose.

Copyright (C) Open Grid Forum (2009-2012). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the OGF or other organizations, except as needed for the purpose of developing Grid Recommendations in which case the procedures for copyrights defined in the OGF Document process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the OGF or its successors or assignees.

Open Grid Forum NSI Connection Services Protocol v2.0 - Service Description template definition.  
-->

```

<xsd:schema targetNamespace="http://schemas.ogf.org/nsi/2013/07/services/description"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:tns="http://schemas.ogf.org/nsi/2013/07/services/description">

  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      This is an XML schema document describing the OGF NSI Service Description
      template.
    </xsd:documentation>
  </xsd:annotation>

  <!-- Element definitions for services types. -->
  <xsd:element name="serviceDescription" type="tns:ServiceDescriptionType">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">
        Element for representing a Service Description XML document.
      </xsd:documentation>
    </xsd:annotation>
  </xsd:element>

  <!-- Type definitions for services types. -->
  <xsd:complexType name="ServiceDescriptionType">
    <xsd:annotation>
      <xsd:documentation xml:lang="en">
        Type defining the structure and content of a Service
        Description document.

        Attributes:

        id - A service description must have a unique identifier within
        the network sourcing the description.

        Elements:

        comment - A comment describes the basic service offered in this
        service description. Any differences from the global service
        description should be identified here.

        serviceType - This is the standard service type as defined in
        the associated global service description. We allocate a
        standard namespace for each template so they are globally
        unique.

        schema - This is the schema elements specified in a reservation.
        There can be multiple schema entries here for services if they
        requiring multiple schema in a reserve request.

        parameter - Parameter definitions for the service and their
        values. These reflect the XML schema definitions and any local
        range restrictions.

        attribute - Attribute definitions for the service and their
        values. Attributes are aspects of the service that are not
        specified in the XML schema for the service.

        error - Errors defined for this service.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="comment" type="xsd:string" minOccurs="0" />
      <xsd:element name="serviceType" type="xsd:string" />
      <xsd:element name="schema" type="tns:SchemaType" minOccurs="1"
maxOccurs="unbounded" />
      <xsd:element name="parameter" type="tns:ParameterType" minOccurs="0"
maxOccurs="unbounded" />
      <xsd:element name="attribute" type="tns:AttributeType" minOccurs="0"
maxOccurs="unbounded" />
      <xsd:element name="error" type="tns:ErrorType" minOccurs="0"
maxOccurs="unbounded" />
      <xsd:any namespace="##other" processContents="lax" minOccurs="0"
maxOccurs="unbounded" />
    </xsd:sequence>
  </xsd:complexType>

```

```

</xsd:sequence>
<xsd:attribute name="id" type="xsd:anyURI" use="required" />
<xsd:anyAttribute namespace="##other" processContents="lax" />
</xsd:complexType>

<xsd:complexType name="SchemaType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      This type identifies the specific service XML schema element
      specified in a reservation. There can be multiple schema
      entries for a service if they require multiple schema in a
      reserve request.

      Attributes:

      name - The name of the element carried in the reserve criteria.

      required - Is this element required in the reserve criteria or
      is it optional.

      namespace - The qualified namesapce of the XML schema defining
      the element.

      type - The XML schema type of the element.

      Elements:

      comment - Descriptive text.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="comment" type="xsd:string" minOccurs="0" />
    <xsd:any namespace="##other" processContents="lax" minOccurs="0"
maxOccurs="unbounded" />
  </xsd:sequence>
  <xsd:attribute name="name" type="xsd:string" use="required" />
  <xsd:attribute name="required" type="xsd:boolean" use="required" />
  <xsd:attribute name="namespace" type="xsd:anyURI" use="required" />
  <xsd:attribute name="type" type="xsd:string" use="required" />
  <xsd:anyAttribute namespace="##other" processContents="lax" />
</xsd:complexType>

<xsd:complexType name="ParameterType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Parameter definitions for the service and their values. These
      reflect the XML schema definitions and any local range
      restrictions. The associated service XML schema is the
      definitive source for and type and range definitions. If a
      parameter of the service is not contained in this service
      descrption then it is not supported for this profile.

      Attributes:

      name - The name of the parameter within the XML schema for the
      service.

      units - (Optinal) The unit of measurement for this parameter.

      required - Indicates if the parameter is mandatory or optional.

      modifiable - Indicates if this parameter be modified after the
      initial reservation has been created.

      namespace - The qualified namesapce of the XML schema defining
      the parameter.

      type - The fully qualified XML schema type of the parameter.

      Elements:
    </xsd:documentation>
  </xsd:annotation>

```

```

comment - Descriptive text describing the parameter.

minInclusive, maxInclusive - Allows the specification of
parameter ranges.

spacing - Allows for the specification of parameter value
increments.

default - The default for the parameter if a value is not
specified. This must not conflict with any default specified
in the XML schema definition for the parameter.
</xsd:documentation>
</xsd:annotation>
<xsd:sequence>
  <xsd:element name="comment" type="xsd:string" minOccurs="0" />
  <xsd:element name="minInclusive" type="xsd:string" minOccurs="0" />
  <xsd:element name="maxInclusive" type="xsd:string" minOccurs="0" />
  <xsd:element name="spacing" type="xsd:string" minOccurs="0" />
  <xsd:element name="default" type="xsd:string" minOccurs="0" />
  <xsd:any namespace="##other" processContents="lax" minOccurs="0"
maxOccurs="unbounded" />
</xsd:sequence>
<xsd:attribute name="name" type="xsd:string" use="required" />
<xsd:attribute name="units" type="xsd:string" use="optional" />
<xsd:attribute name="required" type="xsd:boolean" use="required" />
<xsd:attribute name="modifiable" type="tns:ModifiableType" use="optional" />
<xsd:attribute name="namespace" type="xsd:anyURI" use="required" />
<xsd:attribute name="type" type="xsd:string" use="required" />
<xsd:anyAttribute namespace="##other" processContents="lax" />
</xsd:complexType>

<xsd:complexType name="AttributeType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Attributes are aspects of the service that are not specified in
      the XML schema for the service. They can be as detailed as
      parameters, but are not specified in the reservation request.

      Attributes:

      name - The name of the attribute within the service definition.

      units - (Optional) The unit of measurement for this attribute.

      namespace - The qualified namespace of the XML schema defining
      the attribute. Will typically be the namespace of the service
      specific definition.

      type - The fully qualified XML schema type of the attribute if
      needed.

      Elements:

      comment - Descriptive text describing the attribute.

      minInclusive, maxInclusive - Allows the specification of
      attribute ranges.

      spacing - Allows for the specification of attributes value
      increments.

      default - The default for the attribute if a value is not
      specified.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="comment" type="xsd:string" minOccurs="0" />
    <xsd:element name="minInclusive" type="xsd:string" minOccurs="0" />
    <xsd:element name="maxInclusive" type="xsd:string" minOccurs="0" />
    <xsd:element name="spacing" type="xsd:string" minOccurs="0" />
    <xsd:element name="default" type="xsd:string" minOccurs="0" />
  </xsd:sequence>

```

```

        <xsd:any namespace="##other" processContents="lax" minOccurs="0"
maxOccurs="unbounded" />
    </xsd:sequence>
    <xsd:attribute name="name" type="xsd:string" use="required" />
    <xsd:attribute name="units" type="xsd:string" use="optional" />
    <xsd:attribute name="namespace" type="xsd:anyURI" use="optional" />
    <xsd:attribute name="type" type="xsd:anyURI" use="optional" />
    <xsd:anyAttribute namespace="##other" processContents="lax" />
</xsd:complexType>

<xsd:complexType name="ErrorType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Models an error defined for this service.

      Attributes:

      id - Error identifier uniquely identifying each known fault
      within this service. Must be allocated in the service specific
      error range of 00700.

      text - User friendly message text describing the error.

      Elements:

      comment - Descriptive text describing the error, conditions
      triggering it, and any variables returned.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="comment" type="xsd:string" minOccurs="0" />
  </xsd:sequence>
  <xsd:attribute name="id" type="xsd:string" use="required" />
  <xsd:attribute name="text" type="xsd:string" use="required" />
</xsd:complexType>

<xsd:simpleType name="ModifiableType">
  <xsd:annotation>
    <xsd:documentation xml:lang="en">
      Indicates when in the shedule lifecycle of a reservation a
      parameter can be modified. There are four enumerated values
      for modifiable:

      false - parameter cannot be modified.
      true - parameter can be modified and there are no restrictions.
      pre - parameter can only be modified before schedule start time.
      post - parameter can only be modified after schedule start time.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="false" />
    <xsd:enumeration value="pre" />
    <xsd:enumeration value="post" />
    <xsd:enumeration value="true" />
  </xsd:restriction>
</xsd:simpleType>
</xsd:schema>

```

## 15 Appendix – Service Description example – EVTS.A-GOLE

```

<?xml version="1.0" encoding="UTF-8"?>

<!-- A service description must have a unique identifier within the network
sourcing the description. -->
<sd:serviceDescription id="urn:ogf:network:netherlight.net:2012:service:EVTS.A-GOLE"
  xmlns:sd="http://schemas.ogf.org/nsi/2013/07/services/description"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

  <!-- A comment describes the basic service offered in this service

```



```

description. Any differences from the global service description
should be identified here. -->
<comment>
  This is the Common Service Definition file that defines all parameters
  associated with the Ethernet Framed Transport Service as defined for the
  Automated GOLE network.
</comment>

<!-- This is the standard service type as defined in the associated global
service description. We allocate a standard namespace for each template
so they are globally unique. -->
<serviceType>http://services.ogf.org/nsi/2013/07/descriptions/EVTS.A-
GOLE</serviceType>

<!-- This is the schema elements specified in a reservation. There can be
multiple schema entries here for services if they require multiple
schema in a reserve request.

name - The name of the element carried in the reserve criteria.

required - Is this element required in the reserve criteria or is it
optional.

namespace - The qualified namespace of the XML schema defining the
element.

type - The XML schema type of the element.
-->
<schema name="evts" required="true"
namespace="http://schemas.ogf.org/nsi/2013/07/services/point2point"
type="{http://schemas.ogf.org/nsi/2013/07/services/point2point}EthernetVlanType"/>

<!-- Parameter definitions for the service and their values. These reflect
the XML schema definitions and any local range restrictions. The
associated service XML schema is the definitive source for and type
and range definitions. If a parameter of the service is not contained
in this service description then it is not supported for this profile.

Attributes:

name - The name of the parameter within the XML schema for the
service.

units - (Optional) The unit of measurement for this parameter.

required - Indicates if the parameter is mandatory or optional.

modifiable - Can this parameter be modified after the initial
reservation has been created. There are four values for modifiable:
  false - parameter cannot be modified.
  true - parameter can be modified and there are no restrictions.
  pre - parameter can only be modified before schedule start time.
  post - parameter can only be modified after schedule start time.

namespace - The qualified namespace of the XML schema defining the
parameter.

type - The fully qualified XML schema type of the parameter.

Elements:

comment - Descriptive text describing the parameter.

minInclusive, maxInclusive - Allows the specification of parameter
ranges.

spacing - Allows for the specification of parameter value increments.

default - The default for the parameter if a value is not specified.
This must not conflict with any default specified in the XML schema
definition for the parameter.

```

```

-->
<parameter name="capacity" units="bps" required="true" modifiable="true"
  namespace="http://schemas.opengroup.org/2013/07/services/point2point"
  type="{http://www.w3.org/2001/XMLSchema}int">
  <comment>
    Capacity is defined to be the average quantity of data that will be
    guaranteed to be transported per unit time from ingress to egress,
    normalized to 1 second. The "capacity" for the EVTS instance
    includes the ethernet frame header(s) but does not include sync
    bits or any inter-frame gap considerations. Further, the EVTS
    "capacity" attribute does not include any additional headers that
    may be applied in transit.
  </comment>
  <minInclusive>0</minInclusive>
  <maxInclusive>10000000000</maxInclusive>
  <spacing>1</spacing>
  <default>0</default>
</parameter>

<!-- The service XML schema should fully define enumerated types, their
  defaults, and optionality. This is here for informational purposes. -->
<parameter name="directionality" required="false" modifiable="false"
  namespace="http://schemas.opengroup.org/2013/07/services/point2point"
  type="{http://schemas.opengroup.org/2013/07/services/types}DirectionalityType">
  <comment>
    Directionality specifies whether the Connect requested is to be a
    "unidirectional" or "bidirectional" circuit. The implications for
    unidirectional are fairly obvious. However for bidirectional
    connections the STPs must be defined as bi-directional. Note:
    there may be ambiguity in certain topological scenarios where the
    routing of the connection may not be clear.
  </comment>
  <default>Bidirectional</default>
</parameter>

<!-- The service XML schema should fully define boolean types, their
  defaults, and optionality. The only value for this being here is
  informational and to indicate if the parameter can be modified. -->
<parameter name="symmetricPath" required="false" modifiable="false"
  namespace="http://schemas.opengroup.org/2013/07/services/point2point"
  type="{http://www.w3.org/2001/XMLSchema}boolean">
  <comment>
    An indication that both directions of a bidirectional circuit must
    follow the same path. Only applicable when directionality is
    "Bidirectional". If not specified then value is assumed to be false.
  </comment>
  <default>>false</default>
</parameter>

<!-- This is here for informational purposes. The only value for this being
  here is informational and to indicate if the parameter can be modified.
-->
<parameter name="sourceSTP" required="true" modifiable="false"
  namespace="http://schemas.opengroup.org/2013/07/services/point2point"
  type="{http://schemas.opengroup.org/2013/07/services/types}StpType">
  <comment>
    Source STP of the service.
  </comment>
</parameter>

<parameter name="destSTP" required="true" modifiable="false"
  namespace="http://schemas.opengroup.org/2013/07/services/point2point"
  type="{http://schemas.opengroup.org/2013/07/services/types}StpType">
  <comment>
    Destination STP of the service.
  </comment>
</parameter>

<parameter name="ero" required="false" modifiable="false"
  namespace="http://schemas.opengroup.org/2013/07/services/point2point"
  type="{http://schemas.opengroup.org/2013/07/services/types}StpListType">

```

```

    <comment>
include Hop-by-hop ordered list of STP from sourceSTP to destSTP. List does not
    sourceSTP and destSTP.
    </comment>
</parameter>

<parameter name="mtu" units="bytes" required="false" modifiable="false"
namespace="http://schemas.ogf.org/nsi/2013/07/services/point2point"
type="{http://www.w3.org/2001/XMLSchema}int">
    <comment> Specifies the maximum transmission unit size in bytes. </comment>
    <minInclusive>1500</minInclusive>
    <maxInclusive>9000</maxInclusive>
    <spacing>1</spacing>
    <default>9000</default>
</parameter>

<parameter name="burstsize" units="bytes" required="false" modifiable="false"
namespace="http://schemas.ogf.org/nsi/2013/07/services/point2point"
type="{http://www.w3.org/2001/XMLSchema}int">
    <comment>
        Burstsize describes the maximum amount of data that the ingress STP
        will accept in a single sequence of packets at line rate of that
        STP. Bursts must be separated by a long enough quiescent period for
        the average long term rate of data transfer is within the capacity
        limits. Packets exceeding this burst profile are policed and will
        be immediately dropped when detected.
    </comment>
    <minInclusive>0</minInclusive>
    <maxInclusive>100000000000</maxInclusive>
    <spacing>1</spacing>
</parameter>

<parameter name="sourceVLAN" required="true" modifiable="false"
namespace="http://schemas.ogf.org/nsi/2013/07/services/point2point"
type="{http://www.w3.org/2001/XMLSchema}int">
    <comment>
        The VLAN identifier must be in the range 1780 - 1783.
    </comment>
    <minInclusive>1780</minInclusive>
    <maxInclusive>1783</maxInclusive>
    <spacing>1</spacing>
</parameter>

<parameter name="destVLAN" required="true" modifiable="false"
namespace="http://schemas.ogf.org/nsi/2013/07/services/point2point"
type="{http://www.w3.org/2001/XMLSchema}int">
    <comment>
        The VLAN identifier must be in the range 1780 - 1783.
    </comment>
    <minInclusive>1780</minInclusive>
    <maxInclusive>1783</maxInclusive>
    <spacing>1</spacing>
</parameter>

<!-- Attributes are aspects of the service that are not specified in the
XML schema for the service. -->

<!-- This is an example of something that is not requested, but may be
interesting to someone looking for additional support. Perhaps
annotating support hours would be useful as well.
-->
<attribute name="monitored"
namespace="http://schemas.ogf.org/nsi/2013/07/services/point2point/EVTS.A-GOLE">
    <comment> Indicates this service is monitored. </comment>
</attribute>

<!-- This is an example of a restriction on schedule duration that is
calculated by (endTime - startTime). Would this be considered policy?
-->
<attribute name="duration" units="seconds"

```

```

    namespace="http://services.ogf.org/nsi/2013/07/descriptions/ETS.A-GOLE">
    <comment>
        Indicates this service must not be shorter than 300 seconds nor
        longer than 18000 seconds.
    </comment>
    <minInclusive>300</minInclusive>
    <maxInclusive>18000</maxInclusive>
</attribute>

<!-- Could we also bound times and days of week for schedule? -->

<!-- Errors defined for this service. -->
<error id="00700" text="SERVICE_ERROR">
    <comment>
        Parent error classification for a service specific error. This
        generic service error is generated for any errors that have not
        been specifically enumerated.
    </comment>
</error>
<error id="00701" text="UNKNOWN_STP">
    <comment>
        Could not find the specified STP within the topology database.
        Variables will contain the unknown STP.
    </comment>
</error>
<error id="00702" text="STP_RESOLUTION_ERROR">
    <comment>
        Could not resolve STP to a managing NSA. This would indicate that
        the networkId component of the STP does not have a managing NSA
        within the topology database. Variables will contain the unresolved
        STP.
    </comment>
</error>
<error id="00703" text="VLANID_INTERCHANGE_NOT_SUPPORTED">
    <comment>
        VLAN interchange not supported for requested path. Variables will
        contain the networkId of the network unable to support VLAN
        interchange.
    </comment>
</error>
<error id="00704" text="STP_UNAVAILABLE">
    <comment>
        Specified STP already in use during the requested period. Variables
        will contain the STP already in use.
    </comment>
</error>
<error id="00705" text="BANDWIDTH_UNAVAILABLE">
    <comment>
        Insufficient bandwidth available for reservation. Variable will
        contain requested capacity.
    </comment>
</error>
<error id="00706" text="DIRECTIONALITY_NOT_SUPPORTED">
    <comment>
        Requested directionality is not supported. Variable will contain
        requested directionality.
    </comment>
</error>
</sd:serviceDescription>

```

## 16 References

- [BRADNER] Scott Bradner. Key Words for Use in RFCs to Indicate Requirement Levels, RFC 2119. The Internet Society. March 1997. <http://tools.ietf.org/html/rfc2026>
- [BUSH] Randy Bush, Thomas Narten. Clarifying when Standards Track Documents may Refer Normatively to Documents at a Lower Level. RFC 3967. The Internet Society. December 2004. <http://tools.ietf.org/html/rfc3967>

- [CATLETT] Charlie Catlett, Cees de Laat, David Martin, Gregory B. Newby, Dane Skow. GFD-C.152: Open Grid Forum Document Process and Requirements. Open Grid Forum. June 2009. <http://www.ogf.org/documents/GFD.152.pdf>
- [RESCORLA] Eric Rescorla, Brian Korver, Internet Architectures Board, Guidelines for Writing RFC Text on Security Considerations. RFC 3552. The Internet Society. July 2003. <http://tools.ietf.org/html/rfc3552>