TAPI v2.1.3 Reference Implementation Agreement

TR-547

Version 1.0

July 2020

ONF Document Type: Technical Recommendation

Disclaimer

Open Networking Foundation
1000 El Camino Real, Suite 100, Menlo Park, CA 94025
www.opennetworking.org

Table of Contents

List of Figures

List of Tables

Document History

Version	Date	Description of Change
1.0	July 28, 2020	TR Official version.

1 Introduction

1.1 General introduction to the model

This ONF Technical Recommendation (TR) focuses on the definition of a Reference Implementation guide for a TRANSPORT-API (TAPI) based RESTCONF implementation focus on the v2.1.3 version of the TAPI information models (pruned/refactored from the ONF Core Information Model 1.4) and available in the public ONF GitHub repository at:

https://github.com/OpenNetworkingFoundation/TAPI/releases/tag/v2.1.3

1.2 Introduction to this document

The purpose of this document is to describe a set of guidelines and recommendations for a standard use of the TAPI models in combination with RESTCONF protocol for the implementation of the interface between network systems in charge of the control/management of networks based on WDM/OTN technologies.

The target architectures, for which this reference interface is proposed, it is conceptually described in Figure 2-1. As depicted in the architecture, this reference NBI will be the single interface between Operations Support System (OSS), Orchestrator, (super) Controller, etc [1]. The scope of the architecture covers multiple domains within the same network, and it might consist of one or more layers of controllers, where each layer controller will export a certain level of abstraction through its TAPI context (e.g., a hierarchical controller may consume several domain SDN-C TAPI contexts to conform a multi-domain network and exposed it as an aggregated TAPI context). In this document we will refer the lower layer of controllers as SDN domain controller or SDN-C, and, to any hierarchical controller performing the same management/control capabilities or use cases over multiple network domains as Software-Defined Transport Network (SDTN) controller.

Thus, this specification is intended for the interface between an SDN-C and Orchestrator, (super) Controller or client layer systems (such OSS) where the SDN-C provides its network management through a TAPI context and maintains a synchronized view in a database.

Moreover, the client layer which will consume the TAPI context systems may have distinct roles (e.g., physical inventory) and that they may be composed of different components or applications. E.g., an OSS system composed by different pieces dedicated to different applications (such inventory, assurance or planning).

This document aims to define the base requirements for any TAPI Server entity (e.g., a SDN-C) which is intended to expose the management/control[2] capabilities of any use case such activation/configuration, service provisioning, path-computation and monitoring over a WDM/OTN network, through the interface defined in this document.

The term management/control shall express that the scope is much wider than just configuring. The proposed common interface shall account for:

- Configuration, e.g. for automating and optimizing the network services creation and processes.
- Status, e.g. for automated configuration depending on current network status.
- Events (Alarms), e.g. for automated initiation of countermeasures.
- Current and Historical Performance Values, e.g. for perpetual network analysis.

[1] Any system with a repository that maintains alignment with a view of the underlying system as presented by the controller.
[2] At the time management is automated it simply becomes control as explained by [ONF Core Model].

This specification is supported by standards, protocol specifications, IETF RFCs, ITU-T recommendations and the ONF Transport API (TAPI) documentation. The appropriate references to this supplementary material are included where appropriate along the document to support the statements which conforms this specification. However, this document does not intend to re-define the protocols or information models composing the specification but to complement, clarify or extends in those cases where a corner case or different interpretations have been found along the mentioned standards.

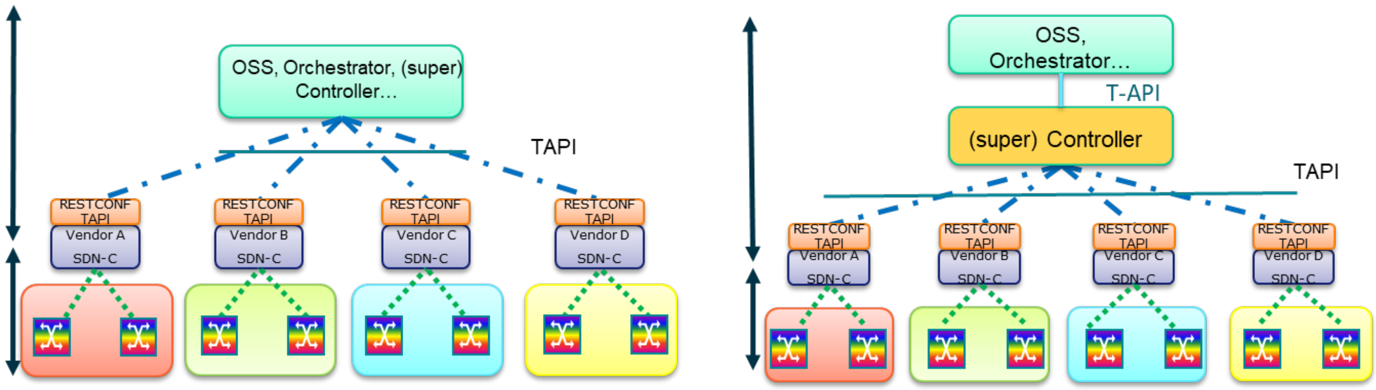


Figure 2-1 Example SDN architecture for WDM/OTN network

2 RESTCONF/YANG Protocol considerations

RESTCONF [RFC 8040] is proposed as the transport protocol for all the defined management operations in the SDN architecture NBIs.

RESTCONF is a HTTP-based protocol that provides a programmatic interface for accessing data defined in YANG 1.0 [RFC 6020] using the data store concepts defined in the Network Configuration Protocol (NETCONF) [RFC 6241].

The RESTCONF specification consists of the following resources:

- **{+restconf}/data (Data API):** Create/Retrieve/Update/Delete (CRUD) based API for the entire data tree defined in the TAPI information model YANG files (see section 3.3).
- **{+restconf}/operations (Operations API):** RPC based API consisting on a small set of operations defined as RPCs in the TAPI information model YANG files.
- **{+restconf}/data/ietf-restconf-monitoring:restconf-state/streams (Notifications API):** Notifications implementation of RESTCONF protocol is defined in https://tools.ietf.org/html/rfc8040#section-6.3.
- **{+restconf}/yang-library-version:** This mandatory leaf identifies the revision date of the "ietf-yang-library" YANG module that is implemented by this server.
- **{+restconf}/data/ietf-restconf-monitoring:restconf-state/capabilities:** leaf to report the server capability of supporting query parameters defined in https://tools.ietf.org/html/rfc8040#section-9.1.

2.1 Root tree discovery

The RESTCONF API **{+restconf}** root resource can be discovered by getting the *"/.well-known/host-meta"* resource ([RFC 6415]) and using the <Link> element containing the *"restconf"* attribute.

The client will send the following query:

```
GET /.well-known/host-meta HTTP/1.1
Host: example.com
Accept: application/xrd+xml
```

The server might respond as follows:

```
HTTP/1.1 200 OK
Content-Type: application/xrd+xml
Content-Length: nnn

<XRD xmlns='http://docs.oasis-open.org/ns/xri/xrd-1.0'>
      <Link rel='restconf' href='/restconf'/>
</XRD>
```

2.2 YANG model's discovery

RESTCONF utilizes the YANG library [RFC 7895] and [RFC 8525] to allow a client to discover the YANG module conformance information for the server, in case the client wants to use it.

The mandatory **{+restconf}/yang-library-version** resource is used to clearly identify the version of the YANG library used by the server.

The server MUST implement the "*ietf-yang-library*" module, which MUST identify all the YANG modules used by the server, within the "modules-state/module" and "yang-library/module-set/module" list resource. The modules set resource is located at (both implementations are accepted so far):

- **According to [RFC 7895]: {+restconf}/data/ietf-yang-library**:**modules-state**
- **According to [RFC 8525]: {+restconf}/data/ietf-yang-library**:**yang-library**

2.3 Operation API (RPC) vs Data API (REST)

Currently there are two allowed APIs resources defined in RESTCONF. Although both are compatible and valid, given the low penetration in the industry of the RPC-based API implementation, it is not currently included in this specification.

Thus, **the current specification takes the support of the RESTCONF ‘data’ API as mandatory; while the ‘operations’ API, based on the TAPI defined RPCs, is considered optional.**

2.4 Query filtering

According to RESTCONF specification, each operation allows zero or more query parameters to be present in the request URI. Specifically, query operations’ parameters are described in Section 4.8 of [RFC 8040]. Thus, the following query parameters MUST be supported by any interface compliant with this specification:

Table 1: RESTCONF Query filters

Name	Methods	Description
content	GET, HEAD	Select config and/or non-config data resources
depth	GET, HEAD	Request limited subtree depth in the reply content
fields	GET, HEAD	Request a subset of the target resource contents
filter	GET, HEAD	Boolean notification filter for event stream resources
with-defaults	GET, HEAD	Control the retrieval of default values
start-time	GET, HEAD	Replay buffer start time for event stream resources
stop-time	GET, HEAD	Replay buffer stop time for event stream resources

The specific use of these query parameters will be detailed in the different Use Cases Low Level Design (LLDs) included in section 6.

The *"depth", “fields”, “filter”, “replay” (which applies to "start-time" and "stop-time" query parameters) and “with-defaults”* query parameter URIs SHALL be listed in the *"capability"* leaf-list as part of the container definition in the *"ietf-restconf-monitoring"* module, defined in Section 9.3 of [RFC 8040], to advertise the server capability of supporting these query parameters. This resource shall be located at:

- **{+restconf}/data/ietf-restconf-monitoring:restconf-state/capabilties**

2.5 Data encoding

JSON encoding formats MUST be supported according to with Section 3.2 of [RFC 8040].

The solution adhering to this specification MUST support media type "application/yang-data+json" as defined in [RFC 7951]. This MUST be advertised in the HTTP Header fields "Accept" or "Content-Type" of the corresponding HTTP Request/Response messages.

2.5.1 Namespaces

According to Section 1.1.5 of [RFC 8040], "*The JSON representation is defined in "JSON Encoding of Data Modeled with YANG" [RFC7951] and supported with the "application/yang-data+json" media type"*.

Thus, any implementation according to this specification MUST be compliant with the rules and definitions included in [RFC 7951], specifically those related to namespaces qualification included in Section 4 of [RFC 7951].

Example:

```
GET /restconf/data/tapi-common:context HTTP/1.1
Host: example.com
Accept: application/yang-data+json

{
  "tapi-common:context": { # Root tree object is qualified by the module name.
    "tapi-connectivity:connectivity-context": { # Any augmentation introduces a new
qualification of the module name where the augmentation was defined.
      "connectivity-service": [
        {
          "uuid": "0b530f9f-0fc3-4d27-b6c3-5c821214db1f"
        }
      ]
    }
  }
```

2.6 Notifications

The solution adhering to this specification must support all YANG-defined event notifications included in the information models included in section 3.2 of this document.

The solution implementing the RESTCONF server must expose its supported notification streams by populating the *"restconf-state/streams"* container definition in the *"ietf-restconf-monitoring"* module defined in Section 9.3 of [RFC 8040]. The streams resource can be found at:

- **{+restconf}/data/ietf-restconf-monitoring:restconf-state/streams**

The RESTCONF server MUST support, at least, the NETCONF event stream with JSON encoding format, as defined in Section 3.2.3 of [RFC5277] and Section 6.2 of [RFC 8040].

The RESTCONF server MUST support the RESTCONF Notifications subscription mechanism is defined in Section 6.3 of [RFC 8040].

The solution must support the *"filter"* Query Parameter, as defined in Section 4.8.4 of [RFC 8040], to indicate the target subset of the possible events being advertised by the RESTCONF server stream.

2.6.1 SSE vs WebSocket

The RESTCONF standard defines the Server Sent Events (SSE) [W3C.REC-xml-20081126] as the standard protocol for RESTCONF stream notification service; however, traditionally (such in in previous OIF TAPI interoperability activities) the WebSocket [RFC 6455] protocol implementations have been availed by ONF and thus it is widely implemented.

3 ONF Transport – API (TAPI) considerations

3.1 TAPI SDK version and documentation

The ONF Transport API (T-API/TAPI) project is constantly evolving and new releases of the information models are periodically updated. All TAPI release notes can be found at:

https://github.com/OpenNetworkingFoundation/TAPI/releases

Current document is focus on TAPI v2.1.3 release.

DISCLAIMER: The PHOTONIC_LAYER_QUALIFIER_MC and PHOTONIC_LAYER_QUALIFIER_OTSiMC layer-protocol-qualifier values have been newly introduced in TAPI v2.1.3. They are equivalent to previous PHOTONIC_LAYER_QUALIFER_SMC and PHOTONIC_LAYER_QUALIFER_NMC of v2.1.2 respectively. Please find more detailed information in [TAPI-TOP-MODEL-REQ-20].

3.2 TAPI Information model

The Transport API abstracts a common set of control plane functions such as Network Topology, Connectivity Requests, Path Computation, OAM and Network Virtualization to a set of Service interfaces. It also includes support for the following technology-specific interface profiles for Carrier Ethernet (L2), Optical Transport Network (OTN) framework (L1-ODU) and Photonic Media (L0-WDM).

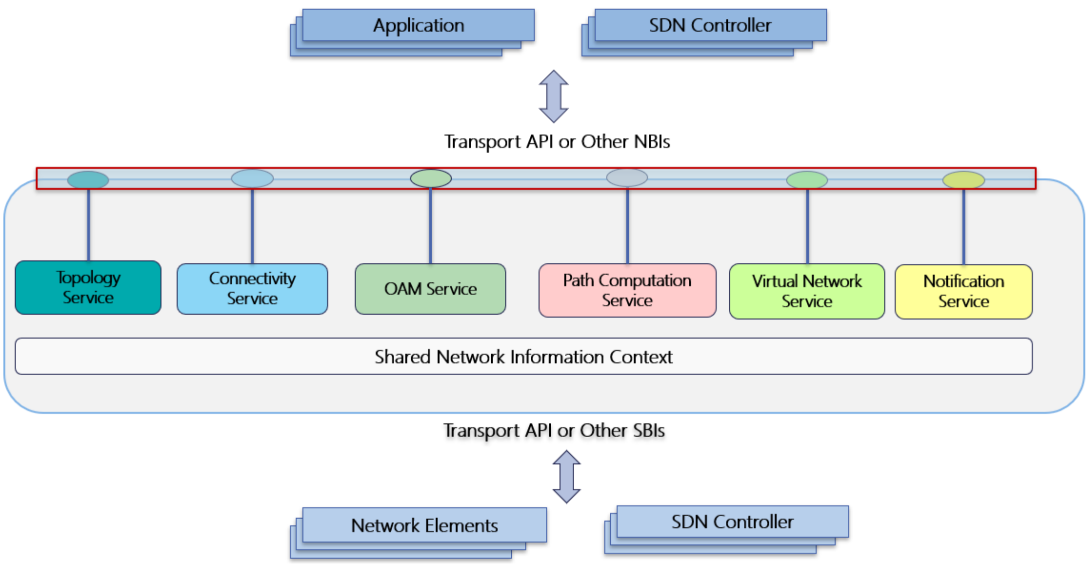


Figure 3-1 Transport API Functional Architecture

The entire list of YANG models composing the TAPI information model can be found in Table 2.

Table 2: TAPI YANG models summary.

Model	Version	Revision (dd/mm/yyyy)
tapi-common.yang	2.1.3	23/04/2020
tapi-connectivity.yang	2.1.3	16/06/2020
tapi-equipment.yang	2.1.3	23/04/2020
tapi-eth.yang	2.1.3	23/04/2020

Model	Version	Revision (dd/mm/yyyy)
tapi-dsr.yang	2.1.3	23/04/2020
tapi-streaming.yang	2.1.3	16/06/2020
tapi-notification.yang	2.1.3	16/06/2020
tapi-oam.yang	2.1.3	23/04/2020
tapi-odu.yang	2.1.3	23/04/2020
tapi-photonic-media.yang	2.1.3	16/06/2020
tapi-path-computation.yang	2.1.3	23/04/2020
tapi-topology.yang	2.1.3	23/04/2020
tapi-virtual-network.yang[3]	2.1.3	16/06/2020

Finally, TAPI models are pruned/refactored from the ONF Core Information Model (Core IM) 1.4 [ONF TR-512], thus some of the Core IM model concepts are key to understand the TAPI semantics and meanings. In this section, we introduce some associations to ONF Core IM concepts, for more a full explanation of these concepts please refer to [ONF TR-512] document.

In the rest of this section it is included a brief overview of the main TAPI concepts which will be used along the rest of the document.

3.2.1 Context

T-API is based on a context relationship between a server and client. A *Context* is an abstraction that allows for logical isolation and grouping of network resource abstractions for specific purposes/applications and/or information exchange with its users/clients over an interface.

It is understood that the APIs are executed within a shared Context between the API provider and its client application. A shared Context models everything that exists in an API provider to support a given API client.

The TAPI server ***tapi-common:context*** includes the following information:

- The set of **Service-Interface-Points** exposed to the TAPI client applications representing the available customer-facing access points for requesting network services. This set must allow connectivity-service creation at the following layers:
  - **DSR Layer:** Models a Digital Signal of an unspecified rate, it could be any type of DSR signal such xGigE, FC-x, STM-x or OTU-k which are included as DSR **tapi-common:LAYER_PROTOCOL_QUALIFIER** valid identities in ***tapi-dsr***. This value can be used when the intent is to represent a generic digital layer signal without making any statement on its format or overhead (processing) capabilities.
  - **ODU Layer:** Models the ODU layer as per [ITU-T G.709]
  - **PHOTONIC_MEDIA Layer:** Models the OCH, OTSi, OTSiA, OTSiG, OMS, OTS and Media channels as per [ITU-T G.872].
- A **topology-context** which includes one or more top-level **Topology** objects which are:

[3] This model is not being use in the present version of this reference implementation agreement.

    - Dynamic representations of the L2-L0 network based on stateful synchronization of the SDN Controller with the network elements.
    - For more details please see Section 4
- A **connectivity-context** which includes the list of **Connectivity-Service** and **Connection** objects created within the TAPI Context.
    - For more details please see Section 5
- A **physical-context** which includes the list of **Devices, Equipment** and **Physical-spans** objects representing the physical inventory provided by the TAPI server.
- A **path-computation-context** which includes the list of **Path Computation Services** (*tapi-path-computation:path-comp-service)* requested to the TAPI server and the set of **Path** objects computed by the server.
- A **notification-context** which includes the list of **notification subscriptions** and the list **notifications** emitted through each notification subscription stream.
    - For more details please see Section 3.4

3.2.2 Node and Topology Aspects of Forwarding Domain

The Forwarding-Domain described in the ONF Core IM, represents the opportunity to enable forwarding between its edge-points. The Forwarding-Domain can hold zero or more instances of Forwarding Constructs (or Connections) and provides the context for requesting and instructing the formation, adjustment and removal of Connections.

The Forwarding-Domain supports a recursive aggregation relationship such that the internal construction of a Forwarding-Domain can be exposed as multiple lower level Forwarding-Domains and associated Links (partitioning).

For the purposes of API requirements, the Forwarding-Domain has been refactored as two separate entities: Topology and Node.

3.2.2.1 Topology

The T-API Topology is an abstract representation of the topological-aspects of a particular set of Network Resources. It is described in terms of the underlying topological network of Nodes and Links that enable the forwarding capabilities of that particular set of Network Resources.

3.2.2.2 Node

The T-API Node is an abstract representation of the forwarding-capabilities of a particular set of Network Resources. It is described in terms of the aggregation of set of ports (Node-Edge-Point) belonging to those Network Resources and the potential to enable forwarding of information between those edge ports.

3.2.2.3 Link

The T-API Link is an abstract representation of the effective adjacency between two or more associated Nodes in a Topology. It is terminated by Node-Edge-Points of the associated Nodes.

3.2.3 Node Edge Point v/s Service End Point v/s Connection End Point

The TAPI Logical-Termination-Point (LTP) – is realized by three different constructs: Node-Edge-Point (NEP), Connection-End-Point (CEP) and Connectivity Service-End-Point (CSEP); they are by design, intended to be a generic, flexible modeling constructs, that can model:

- Different technology layers
- Different network configurations
- Different vendor equipment capabilities

So as such, the inherent flexibility, while preserving the underlying pattern, could lead to different model arrangements for same functional configuration.

The Logical-Termination-Point (LTP) described in the ONF Core IM, represents encapsulation of the addressing, mapping, termination, adaptation and OAM functions of one or more transport layers (including circuit and packet forms). Where the client – server relationship is fixed 1:1 and immutable, the different layers can be encapsulated in a single LTP instance. Where there is a n:1 relationship between client and server, the layers are split over separate instances of LTP.

Functions that can be associated/disassociated to/from a Connection, such as OAM, protection switching, and performance monitoring are referenced as secondary entities through the associated LTP instance.

Following an illustrative mapping between ITU-T G.800/805 and TAPI constructs is described.

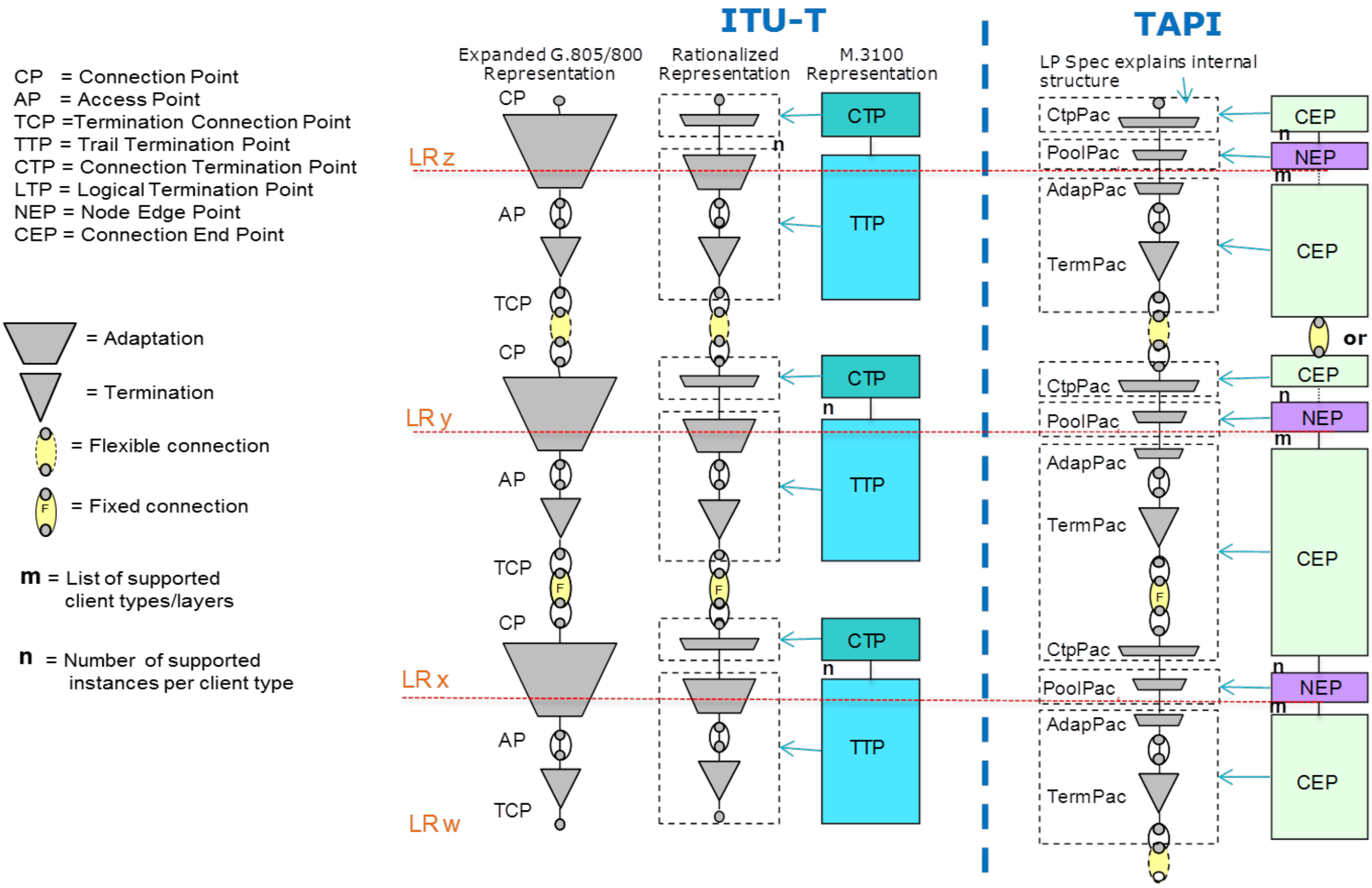


Figure 3-2 TAPI Mapping from ITU-T.

3.2.3.1 Node-Edge-Point (NEP)

The Node-Edge-Point represents the inward network-facing aspects of the edge-port functions that access the forwarding capabilities provided by the Node. Hence it provides an encapsulation of addressing, mapping, termination, adaptation and OAM functions of one or more transport layers (including circuit and packet forms) performed at the entry and exit points of the Node. The Node-Edge-Points have a specific role and directionality with respect to a specific Link.

3.2.3.2 Service-Interface-Point (SIP)

The TAPI Service-Interface-Point represents the outward customer-facing aspects of the edge-port functions that access the forwarding capabilities provided by the Node. Hence it provides a limited, simplified view of interest to external clients (e.g. shared addressing, capacity, resource availability, etc.), that enable the clients to request connectivity without the need to understand the provider network internals. Service-Interface-Point have a mapping relationship (one-to-one, one-to-many, many-to-many) to Node-Edge-Points.

3.2.3.3 Connection-End-Point (CEP)

The Connection-End-Point represents the ingress/egress port aspects that access the forwarding function provided by the Connection. The Connection-End-Points have a client-server relationship with the Node-Edge-Points. The Connection-End-Points have a specific role and directionality with respect to a specific Connection.

3.2.4 Service, Connection and Route

3.2.4.1 Connectivity-Service (CS)

The T-API Connectivity-Service represents an “intent-like” request for connectivity, between two or more Service-Interface-Points exposed by the Context. As such, Connectivity-Service is a container for connectivity request details and is distinct from the Connection that realizes the request. The requestor of the Connectivity-Service is expected to be able to express their intent using just an “external” Node view of Forwarding-Domain and the advertised Service-Interface-Points and not require knowledge of the “internal” Topology details of the Forwarding-Domain.

3.2.4.2 Connection

The T-API Connection represents an enabled (provisioned) potential for forwarding (including all circuit and packet forms) between two or more Node-Edge-Points (another realization of LTP described in the ONF Core IM) from the Node aspect of the Forwarding-Domain. A Connection is typically described utilizing the “internal” Topology view of Forwarding-Domain. The T-API Connection is terminated by Connection-End-Points which are clients of the associated Node-Edge-Points. As such, the Connection is a container for provisioned connectivity that tracks the state of the allocated resources and is distinct from the Connectivity-Service request.

In this specification we distinguish two different types of connections:

- **Cross-Connections (XC)** – defined as a connection between Connection-End-Points of the same layer within a Forwarding-Domain (represented as a *tapi-topology:node* object).
- **Top Connections**– are defined as end-to-end connections between Connection-End-Points within the same layer which may span multiple Forwarding-Domains. Top connections are composed by zero or more XCs which belong to the same layer of the Top Connection. The general rules that applies to the creation of Top Connections are introduced in Section 5.1.

3.2.4.3 Route

The TAPI Route represents the route of a Top Connection through the Topology representation. It is described as a list of Connection End-Points (CEPs) cross-connected by the underlying Lower-Connections referenced in the lower-connection list of the Top Connection[4].

The following route states are foreseen:

- Current route, i.e. the route where the signal is flowing according to Controller's best knowledge.
- Not Current route, applicable in case of resiliency schemes.

Note that *lower-connections* are used to reflect partitioning and *route* to reflect signal flow.

3.2.4.4 Path

The TAPI Path is used to represent the output of path computation APIs and is described by an ordered list of TE Links, either as strict hops (Node-Edge-Points) or as loose hops (Nodes).

3.3 TAPI Data API

As it was described in Section 3.2 he TAPI information models can be divided into: Data API and Operations API. In this first specification, only the Data API be supported, and the Operations API is considered as optional and its support will be considered as a plus, because it can give for flexibility to the TAPI Client applications in order to implement the use cases proposed in this document.

Therefore, the list of Data API entries inferred from the TAPI YANG information models, following a RESTCONF API implementation according to the guidelines included in section0, contains a huge list of entries which MAY not be needed to implement the use cases proposed, and thus not all entries are considered mandatory in this version of the NBI requirements. This API entries support different REST operations namely, CREATE, RETRIEVE, UPDATE and DELETE (i.e., CRUD). For further clarification of its implementation according to the RESTCONF standard please see Section 4, RFC 8040.

Thus, for the first version of this specification, it is proposed a minimal set of objects which shall support full CRUD support according to the TAPI YANG model's specification (e.g., configurable objects should support all operations while non configurable objects shall support only the RETRIEVE operation). Please note that although the list of API entries is reduced here, the whole model MUST be supported, e.g., all child resources of the proposed list of objects need to be configurable.

The complete mandatory operation set of TAPI objects required here can be found in Table 3.

Table 3: Minimum subset required of TAPI RESTCONF Data API

API Entry	RESTCONF Operations allowed
/tapi-common:context/	GET, PUT, PATCH[5]
/tapi-common:context/service-interface-point/	GET
/tapi-common:context/service-interface-point={uuid}/	GET, PUT, DELETE, PATCH
/tapi-common:context/tapi-topology:topology-context/nw-topology-service/	GET

[4] The TAPI Connection Route is described in terms of Cross-Connections rather than Link-Connections (Top Connections). Conceptually a Connection Route is concatenation of Link Connections (resources associated with a Link) and Cross-Connections (resources within the Nodes in the underlying Topology).

[5] POST, DELETE operations are not intended for the context root object.

/tapi-common:context/tapi-topology:topology-context/topology={uuid}/	GET
/tapi-common:context/tapi-topology:topology-context/topology={uuid}/node={uuid}/	GET
/tapi-common:context/tapi-topology:topology-context/topology={uuid}/link={uuid}/	GET
/tapi-common:context/tapi-topology:topology-context/topology={uuid}/node={uuid}/owned-node-edge-point={uuid}/	GET
/tapi-common:context/tapi-topology:topology-context/topology={uuid}/node={uuid}/owned-node-edge-point={uuid}/tapi-connectivity:cep-list/	GET
/tapi-common:context/tapi-topology:topology-context/topology={uuid}/node={uuid}/owned-node-edge-point={uuid}/tapi-connectivity:cep-list/connection-end-point={uuid}/	GET
/tapi-common:context/tapi-connectivity:connectivity-context/	GET, POST, PUT, DELETE, PATCH
/tapi-common:context/tapi-connectivity:connectivity-context/connectivity-service/	GET
/tapi-common:context/tapi-connectivity:connectivity-context/connectivity-service={uuid}/	GET, PUT, DELETE, PATCH
/tapi-common:context/tapi-connectivity:connectivity-context/connection={uuid}/	GET
/tapi-common:context/tapi-path-computation:path-computation-context/	GET, POST, PUT, DELETE, PATCH
/tapi-common:context/tapi-path-computation:path-computation-context/path-comp-service/	GET
/tapi-common:context/tapi-path-computation:path-computation-context/path-comp-service={uuid}/	GET, PUT, DELETE, PATCH
/tapi-common:context/tapi-path-computation:path-computation-context/path={uuid}/	GET
/tapi-common:context/tapi-common:context/tapi-notification:notification-context/	GET
/tapi-common:context/tapi-common:context/tapi-notification:notification-context/notif-subscription/	GET
/tapi-common:context/tapi-common:context/tapi-notification:notification-context/notif-subscription={uuid}/	GET, PUT, DELETE, PATCH

Please note that the TAPI implementation based on the RESTCONF standard defines a much wider set of API entries, thus consider the previous list as a reduction of the implementation scope.

3.4 TAPI Notifications

The current TAPI information model includes a specific model, the **tapi-notification@2018-06-16.yang**, which defines the TAPI notifications format but also a custom TAPI notification subscription procedure to enable a TAPI clients to subscribe to receive these notifications in the form of asynchronous events.

This TAPI Notification mechanism MUST be compatible with the standard RESTCONF notification subscription mechanism already described in Section 2.6.

4 Topology abstraction model

In this chapter a reference topology abstraction model is described. Due to the need of composing a unified view of the network resources along different TAPI implementations, some guidelines are required in order to constrain the possibilities or interpretations of the current proposed models.

The topology model should provide the explicit multi-layer topology representation of the L2-L0 network including OTS, OMS, MC, OTSIMC, OTSi/OTSiA, ODU, DSR layers.

Please note that OTU layer is intentionally simplified in TAPI model. ODU and OTSiA/OTSi representation is considered enough to cover all our defined use cases.

Based on ONF TAPI 2.1.3 models, a topology abstraction view is described for vendor agnostic integration across management/control systems in the frame of the proposed architecture in Section 3. The **TAPI Topology Flat Abstraction model** collapses all layers in a single multi-layer topology. The nomenclature **T0 – Multi-layer topology** and **T#0** is used interchangeably to reference this topology in the remaining document.

4.1 Model guidelines

To properly describe the topology abstraction model proposed, the following global guidelines are presented:

[TAPI-TOP-MODEL-REQ-1] The network logical abstraction collapses all network layers (DSR, ODU, OTSi/OTSiA and Photonic Media (OTSiMC, MC, OMS, OTS)) which are represented explicitly into a single topology **(T0 – Multi-layer topology),** modelled as a ***tapi-topology:topology*** object within the ***tapi-topology:topology-context/tapi-topology:nw-topology-service*** and ***tapi-topology:topology-context/topology.***

[TAPI-TOP-MODEL-REQ-2] The T#0 abstraction MUST be presented as a ***tapi-topology:topology*** object within the ***tapi-topology:topology-context/topology*** and referenced at ***tapi-topology:topology-context/tapi-topology:nw-topology-service,*** together with other topologies that the server may implement.

```
module: tapi-topology
  augment /tapi-common:context:
    +--rw topology-context
       +--ro nw-topology-service
       |  +--ro topology* [topology-uuid]
       |  |  +--ro topology-uuid    -> /tapi-common:context/tapi-topology:topology-
context/topology/uuid
       |  +--ro uuid?       uuid
       |  +--ro name* [value-name]
       |  |  +--ro value-name    string
       |  |  +--ro value?        string
       +--ro topology* [uuid]
```

[TAPI-TOP-MODEL-REQ-3] The Service Interface Points (SIPs) are employed to represent the available service entry points. Each SIPs MUST have at least one NEP related to it, and all DSR, ODU and PHOTONIC_MEDIA node-edge-points (NEPs), which support service configuration, SHOULD have SIPs associated to them (e.g., restrictions imposed by hardware constrains could reduce this list, i.e., transponders with fixed DSR-ODU mapping).

[TAPI-TOP-MODEL-REQ-4] A SIP is logically mapped to topology NEPs through the ***tapi-topology:owned-node-edge-point/mapped-service-interface-point*** attribute:

```
  augment /tapi-common:context:
       +--ro topology* [uuid]
          +--ro node* [uuid]
          |  +--ro owned-node-edge-point* [uuid]
          |  |  +--ro mapped-service-interface-point* [service-interface-point-uuid]
          |  |  |  +--ro service-interface-point-uuid -> /tapi-common:context/service-
interface-point/uuid
```

The T0 – Multi-layer topology MUST include:

The following topology abstraction model proposes some degrees of freedom for the TAPI Server topology abstraction model implementation (e.g., L0 Photonic Media layer bidirectional/unidirectional or Transitional Link vs Multi-Layer node approaches). Thus, the TAPI Client would eventually need to deal with the TAPI Server implementation as soon as it is according to the rules described in this document.

DSR/ODU Layers:

[TAPI-TOP-MODEL-REQ-5] **DSR/ODU forwarding domains** represented as multi-layer and multi-rate ***tapi-topology:node***, allowing the representation of the internal mapping between DSR and ODU NEPs (multi-layer) and the multiplexing/de-multiplexing across different ODU rates (multi-rate)**.**

[TAPI-TOP-MODEL-REQ-6] The DSR/ODU layer network MUST be represented explicitly at the lowest partitioning level, i.e., each **DSR/ODU forwarding domain** MUST be represented as a single ***tapi-topology:node.*** The following network components included within the category of ODU forwarding domain are:

- Transponders.
- Muxponders.
- OTN switching nodes connecting client and line boards.

[TAPI-TOP-MODEL-REQ-7] The NEPs layer qualifications allowed for these layer nodes are:

- **layer-protocol-name=DSR, ODU**
- **supported-cep-layer-protocol-qualifier = [**
  - all identities with base **tapi-dsr:DIGITAL_SIGNAL_TYPE,**
  - all identities with base **tapi-odu:ODU_TYPE**]

[TAPI-TOP-MODEL-REQ-8] NEPs forwarding capabilities within a node can be **optionally** constrained by using ***node-rule-group/rules/forwarding-rules.*** This feature might be required for some use cases where an external path computation entity is placed on top of the TAPI Server, the details of whether this requirement MUST be fulfil will be introduced, where appropriate, in the use cases section. The NEPs can be segmented according to the following conditions:

- **Different layer-protocol-qualifier.** In case a multi-layer DSR/ODU node includes NEPs with different **layer-protocol-qualifier** types (i.e., between different DSR_SIGNAL_TYPEs or ODU_TYPE), each group SHALL be segmented with a node-rule-group, including:
  - **forwarding-rule=MAY_FORWARD_ACROSS_GROUP**
- **Not forwarding between same device ports**. In some case it might be relevant to restrict the forwarding between client ports at the same network device (e.g., transponder). In this case ALL NEPs related to client ports at the same device SHALL be segmented with a node-rule-group, including:

- **forwarding-rule=CANNOT_FORWARD_ACROSS_GROUP**

In case these constrains exists in the network and a service is requested between NEPs which are not potentially connected, **the TAPI Server MUST reject any Connectivity Service request no matter this restriction was exposed or not.**

```
module: tapi-topology
  augment /tapi-common:context:
      +--rw topology-context
        +--ro topology* [uuid]
          +--ro node* [uuid]
          |  +--ro node-rule-group* [uuid]
          |  |  +--ro rule* [local-id]
          |  |  +--ro rule-type?           rule-type
          |  |  |  +--ro forwarding-rule?     forwarding-rule
          |  |  |  +--ro override-priority?   uint64
          |  |  |  +--ro local-id             string
          |  |  |  +--ro name* [value-name]
          |  |  |     +--ro value-name    string
          |  |  |     +--ro value?        string
          |  |  +--ro node-edge-point* [topology-uuid node-uuid node-edge-point-uuid]
          |  |  |  +--ro topology-uuid           -> /tapi-common:context/tapi-
topology:topology-context/topology/uuid
          |  |  |  +--ro node-uuid               -> /tapi-common:context/tapi-
topology:topology-context/topology/node/uuid
          |  |  |  +--ro node-edge-point-uuid -> /tapi-common:context/tapi-
topology:topology-context/topology/node/owned-node-edge-point/uuid
```

[TAPI-TOP-MODEL-REQ-9] The ODU <-> OTSI transitions MAY be represented by a **transitional link** between ODU NEPs representing the line side transmission and OTSi NEPs. A transitional link MUST be represented as a ***tapi-topology:link*** object including a ***tapi-topology:transitioned-layer-protocol-name*** leaf-list attribute, which includes the pair of layer transition, e.g., [ODU, PHOTONIC_MEDIA].

```
module: tapi-topology
  augment /tapi-common:context:
    +--ro topology* [uuid]
       +--ro link* [uuid]
       |  +--ro transitioned-layer-protocol-name* string
```

The transitional-link representation MUST be present at 'Day 0' (for more detailed description please check the examples included in section 6.1.2.2) TAPI Server representation of the topology, between two NEP pools at the ODU and PHOTONIC_MEDIA (OTSi layer qualified) layers. These NEP pools are intended to represent solely the adjacency between the two nodes representing the electrical and optical side of the target optical terminal (i.e., these NEPs are not intended to represent the inverse multiplexing capabilities of the ODU CEPs over OTSi generated NEP resources).

The model assumes a single NEP pool at the OTSi layer. It MUST include a reference to every NEP representing the TTP facing side of each optical line port exposing OTSi connectivity capacity. These references MUST be implemented by referencing individual OTSi NEPs associated to each physical Optical Line interface using the ***tapi-topology:node/owned-node-edge-point/aggregated-node-edge-point*** attribute within the NEP pool representation.

At the ODU side, the NEP representation is the same, but it is assumed that the individual NEPs will be created (and "attached" to the ODU NEP pool) dynamically by TAPI server, as a response of the creation of OTSi connections in the lower layer, i.e., the individual ODU NEPs exposing ODU connectivity resources will only be available after the lower layer (OTSi) will be provisioned.

[TAPI-TOP-MODEL-REQ-10] Alternatively, the ODU <-> OTSI transitions MAY be represented within the same DSR/ODU multi-layer node, by including the OTSI/OTSiA layers representation of the optical side of the optical terminals. In this case the ODU<-> OTSI transition MUST be represented as stack of ***tapi-topology:node-edge-point*** and ***tapi-connectivity:connection-end-points*** related to each other by ***tapi-connectivity:connection-end-point/parent-node-edge-point*** and ***tapi-connectivity:connection-end-point/client-node-edge-point*** attributes:

```
augment /tapi-common:context/tapi-topology:topology-context/tapi-
topology:topology/tapi-topology:node/tapi-topology:owned-node-edge-point:
    +--ro cep-list
       +--ro connection-end-point* [uuid]
          +--ro parent-node-edge-point
          |  +--ro topology-uuid?          -> /tapi-common:context/tapi-
topology:topology-context/topology/uuid
          |  +--ro node-uuid?              -> /tapi-common:context/tapi-
topology:topology-context/topology/node/uuid
          |  +--ro node-edge-point-uuid?   -> /tapi-common:context/tapi-
topology:topology-context/topology/node/owned-node-edge-point/uuid
          +--ro client-node-edge-point* [topology-uuid node-uuid node-edge-point-uuid]
          |  +--ro topology-uuid           -> /tapi-common:context/tapi-
topology:topology-context/topology/uuid
          |  +--ro node-uuid               -> /tapi-common:context/tapi-
topology:topology-context/topology/node/uuid
          |  +--ro node-edge-point-uuid    -> /tapi-common:context/tapi-
topology:topology-context/topology/node/owned-node-edge-point/uuid
```

In case of the multi-layer nodes collapses the OTSi/OTSiA layer together with DSR and ODU layers into the same ***tapi-topology:node*** representation the NEPs CAN be qualified as:

- **tapi-photonic-media:PHOTONIC_LAYER_QUALIFER_OTSI**

Moreover, in case of the multi-layer node implementation, the following requirements' set, related to the OTSI PHOTONIC_MEDIA layer, MUST apply to the multi-layer node construct too.

Please note, the TAPI Client MUST deal with both ODU-OTSI transition modelling approaches.

OTSi/Photonic Media layers:

[TAPI-TOP-MODEL-REQ-11] The OTSi layer represents the optical side of the optical terminals (transponders/muxponders). This layer consists of nodes representing the mapping and multiplexing of OTSi signals. It consists of nodes including OTSi client endpoints representing the Trail Termination Points (TTPs) of the OTSi connections and OTSi/OMS endpoints representing the physical connectivity with ROADM/FOADM add/drop ports. This optical line interfaces representation shall be available at 'Day 0' i.e., after the Optical Terminals commissioning stage and prior to any service deployment over the optical line interfaces. For more detail please check the examples included in section 6.1.2.2.

[TAPI-TOP-MODEL-REQ-12] The physical connectivity between transponder/muxponder line ports and ROADM/FOADM's add/drop ports MUST be represented as UNIDIRECTIONAL or BIDIRECTIONAL **tapi-topology:links** between **PHOTONIC_MEDIA NEPs. These NEPs MUST be qualified as layer-protocol-qualifier: LAYER_PROTOCOL_QUALIFIER_UNSPECIFIED.**

These NEPs represents the OTS and OMS switching layers and they should be the mounting point for any monitoring function in any of those layers.

[TAPI-TOP-MODEL-REQ-13] Independently of OMS NEPs directionality, the OTSI/OTSiA NEP representations on top of OMS NEP/CEPs MUST be BIDIRECTIONAL.

[TAPI-TOP-MODEL-REQ-14] OTSi NEPs MUST be present on top of OMS NEPs/CEPs to represent the effective forwarding of OTSi connections over the non-qualified PHOTONIC_MEDIA link between Transponder Line Ports and ROADM Add/Drop ports.

[TAPI-TOP-MODEL-REQ-15] The NEPs layer qualifications allowed **within OTSi** nodes MUST be:

- **layer-protocol-name= PHOTONIC_MEDIA**
- **supported-cep-layer-protocol-qualifier= [PHOTONIC_LAYER_QUALIFIER_OTSi/OTSiA], [LAYER_PROTOCOL_QUALIFIER_UNSPECIFIED]**

[TAPI-TOP-MODEL-REQ-16] Each OTSi/OTSiA NEP MAY include the ***tapi-photonic-media:media-channel-node-edge-point-spec*** to represent the media channel pool resources supportable, available and occupied.

[TAPI-TOP-MODEL-REQ-17] Generally, transponder/muxponder line ports and ROADM/FOADM's add/drop ports are 1:1 relation, in case Optical Line Protection (OLPs) are present, OLP complexity shall be always represented in the Photonic Media layer (for further description please see Use Case 5b).

Photonic-Media forwarding domains represents the Photonic Media (Open Line System (OLS)) network

[TAPI-TOP-MODEL-REQ-18] The Photonic-Media forwarding domains can be mapped to OLP, ROADM/FOADM and ILA network elements which connectivity is always represented as PHOTONIC_MEDIA links (which may be augmented with OTS/OMS layers monitoring OAM functions). These forwarding domains SHALL expose the capability of create Media Channel connection and connectivity services between its endpoints.

[TAPI-TOP-MODEL-REQ-19] The NEPs layer qualifications allowed **at the Photonic-Media forwarding domains (i.e., tapi-topology:node)** are:

- **layer-protocol-name= PHOTONIC_MEDIA**
- **supported-cep-layer-protocol-qualifier = [**

  LAYER_PROTOCOL_QUALIFIER_UNSPECIFIED,

  PHOTONIC_LAYER_QUALIFIER_MC,

  PHOTONIC_LAYER_QUALIFIER_OTSiMC].

[TAPI-TOP-MODEL-REQ-20] Media Channel layer is consisting on Media-Channel (MC) constructs, representing a reserved portion of the spectrum to route the OTSi signals, and by the OTSiMC layer which represents the actual portion of the spectrum occupied by the signal (MC spectrum must be wider than the OTSiMC). Please see Figure 4-1 graphical representation for more clarity.

The MC layer MUST be represented by client NEPs/CEPs on top of the PHOTONIC_MEDIA/ LAYER_PROTOCOL_QUALIFIER_UNSPECIFIED (representing OTS/OMS layers) NEPs/CEPs at the OLS side of the links between Transponder Line Ports and ROADM Add/Drop ports.

The OTSiMC layer MAY be optionally represented by client NEPs/CEPs on top of the PHOTONIC_LAYER_QUALIFER_MC CEPs. The representation of this layer does only provide monitoring information but not switching (which only happens at MC layer), thus its inclusion depends on the HW monitoring capabilities of the optical HW components of the network.

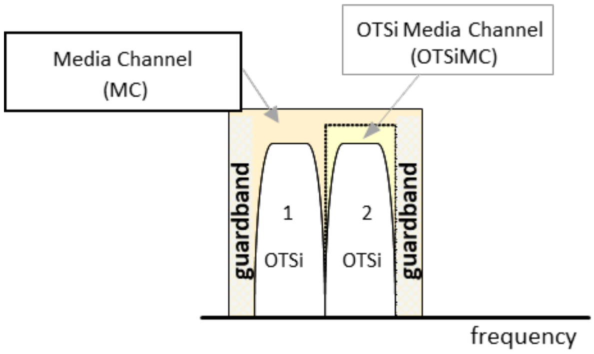


Figure 4-1 Media-channel entities relationship.

[TAPI-TOP-MODEL-REQ-21] Each MC, OTSiMC NEP MUST include the ***tapi-photonic-media:media-channel-node-edge-point-spec*** to represent the media channel pool resources supportable, available and occupied.

[TAPI-TOP-MODEL-REQ-22] It is assumed at minimum, the lower layer forwarding constructs (**tapi-topology:link**) between forwarding domains (**tapi-topology:node**) which need to be represented in this topology should be PHOTONIC_MEDIA links collapsing OTS and OMS layers. **These links MUST be configured in the network in advance without being needed to be created during upper layer services provisioning process**.

[TAPI-TOP-MODEL-REQ-23] In case OLP constructs are present for OMS or OTS protection, this should be represented in TAPI by using ***tapi-topology:resilience-type/tapi-topology:protection-type*** link's attribute. Underlying switch control for OMS or OTS protection is out of the scope of this modelling.

4.2 Inventory considerations

Hardware identifiers currently stored in legacy OSS inventory systems MUST be correlated with T-API UUID identifiers. This information will be provided by the SDN optical domain controller suppliers as a pre-requisite for the use cases described in section 6 of the present document.

For every inventory element represented as a logical element in TAPI by the SDN Domain controller, an **INVENTORY_ID** ***tapi-common:name*** property shall be included into the logical element construct.

The **INVENTORY_ID** tag SHALL be included for the following TAPI objects:

- ***tapi-topology:node-edge-point***
- ***tapi-common:service-interface-point***

The proposal for a common definition of the **INVENTORY_ID** tag, follows 2 main principles and it is based on [TMF-814] naming standards:

- It is explicit and clear: there's no ambiguity to which field each index correspond

- It can be augmented: if a new type of field needs to be inserted it does not break compatibility with the former format.

The generic format is the concatenation of *n* tuple elements "/<field>=<index>"

The supported fields for tuple elements are:

Table 4: Inventory-id fields format.

<field>	**meaning**
ne	Network Element
r	Rack
sh	Shelf
s_sh	Sub-shelf
sl	Slot
s_sl	Sub-slot
p	Port

The supported sequence for the tuple is the following and covers a variety of supported scenarios that may not all be applicable.

- [ ] means that may not be present
- […] means that multiple values can be specified (marked as green **x** in the matrix)

```
/ne=<nw-ne-name>[/r=<r_index>][/sh=<sh_index>[/s_sh=<s_sh_index>
…]][[/sl=<sl_index>[/s_sl=<s_sl_index> …]][/p=<p_index> …]]
```

- **<nw-ne-name>** is the real **Network Element (NE)** name configured in the network (i.e. not managed by the SDN-C) and MUST be unique along all exposed interfaces of the network control and management planes (i.e., Network Management Systems (NMSs) or Element Management Systems (EMSs) exposing network information).
- **<r_index>** is the real **Rack index** configured in the network (i.e. not managed by the SDN-C) and MUST be unique along all exposed interfaces of the network control and management planes (i.e., Network Management Systems (NMSs) or Element Management Systems (EMSs) exposing network information).
- **<sh_index>** is the real **Shelf index** configured in the network (i.e. not managed by the SDN-C) and MUST be unique along all exposed interfaces of the network control and management planes (i.e., Network Management Systems (NMSs) or Element Management Systems (EMSs) exposing network information).
- **<s_sh_index>** is the real **Sub-Shelf index** configured in the network (i.e. not managed by the SDN-C) and MUST be unique along all exposed interfaces of the network control and management planes (i.e., Network Management Systems (NMSs) or Element Management Systems (EMSs) exposing network information).
- **<sl_index>** is the real **Slot index** configured in the network (i.e. not managed by the SDN-C) and MUST be unique along all exposed interfaces of the network control and management planes (i.e.,

Network Management Systems (NMSs) or Element Management Systems (EMSs) exposing network information).

- **<s_sl_index>** is the real **Sub-Slot index** configured in the network (i.e. not managed by the SDN-C) and MUST be unique along all exposed interfaces of the network control and management planes (i.e., Network Management Systems (NMSs) or Element Management Systems (EMSs) exposing network information).
- **<p_index>** is the real **Port index** configured in the network (i.e. not managed by the SDN-C) and MUST be unique along all exposed interfaces of the network control and management planes (i.e., Network Management Systems (NMSs) or Element Management Systems (EMSs) exposing network information).

Meaning for the port the following possible combinations depicted in the following matrix.

Each column represents which tuples can be after the element listed in the first column.

Table 5: Inventory-id fields combination allowance.

	/r= <r_index>	/sh= <sh_index>	/s_sh= <s_sh_index>	/sl= <sl_index>	/s_sl= <s_sl_index>	/p= <p_index>
/ne=<nw-ne-name>	**X**	**x**	**-**	**x**	**-**	**x**
/r=<r_index>	**-**	**x**	**-**	**x**	**-**	**-**
/sh=<sh_index>	**-**	**-**	**x**	**x**	**-**	**-**
/s_sh=<s_sh_index>	**-**	**-**	**-**	**x**	**-**	**-**
/sl=<sl_index>	**-**	**-**	**-**	**-**	**x**	**x**
/s_sl=<s_sl_index>	**-**	**-**	**-**	**-**	**x**	**x**
/p=<p_index>	**-**	**-**	**-**	**-**	**-**	**-**

Some examples of INVENTORY_ID for the node-edge-points potentially mapped to the ports described in the examples shown in Figure 6-25 in Section 6.4.2.2:

```
Example 1:
"name": [{"value_name": "INVENTORY_ID", "value":
"/ne=MadridNorte/r=1/sh=1/sl=1/s_sl=0"}]

Example 2:
"name": [{"value_name": "INVENTORY_ID", "value":
"/ne=MadridNorte/r=1/sh=2/sl=2/s_sl=1/p=3"}]

Example 3:
"name": [{"value_name": "INVENTORY_ID", "value":
"/ne=MadridNorte/r=1/sh=3/sl=7/s_sl=2/p=2"}]
```

4.3 Network scenarios

Now we introduce two network scenarios, as the base examples to clearly explain the model assumptions explained before.

4.3.1 Scenario 1: ROADM network equipped with OTN matrices

This first scenario represents a three-ROADM network equipped with OTN matrices to perform grooming of Ethernet 1G and 10G client signals into 100G line-side interfaces transmitting OTSi colored optical wavelengths.

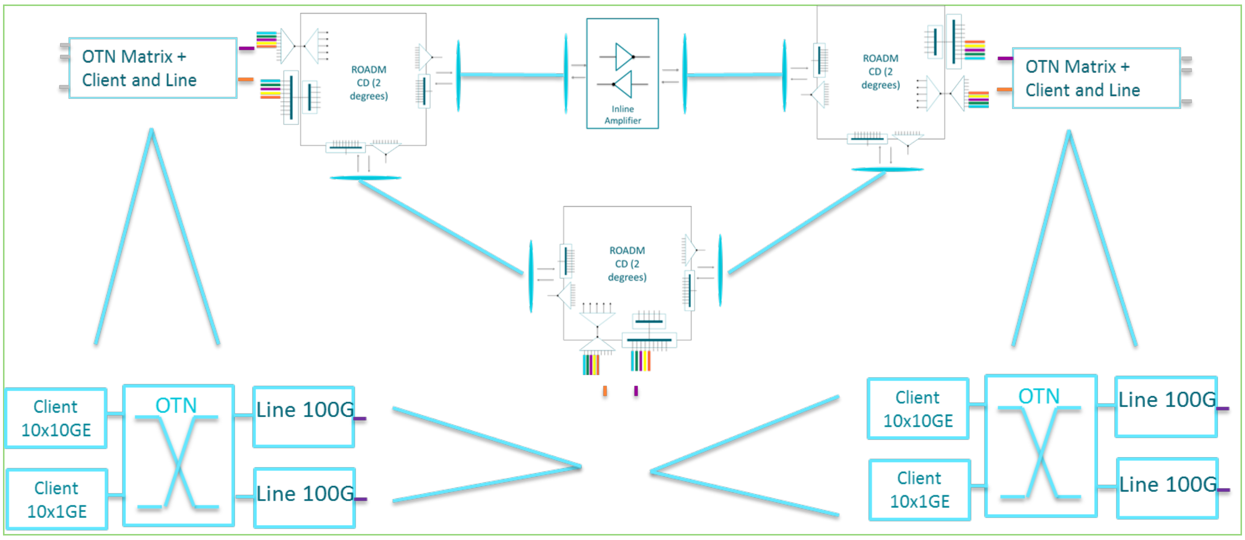


Figure 4-2 NS-1: OTN/WDM Network scenario

4.3.1.1 Model representation (Transitional Link approach)

Please note, in the following representations, the OMS and OTS layers are present purely for information purposes. The specification has stated that these two layers are collapsed from the switching representation perspective. They are represented using LAYER_PROTOCOL_QUALIFIER_UNSPECIFIED layer-protocol-qualifier construct. Please find more details in [TAPI-TOP-MODEL-REQ-22].

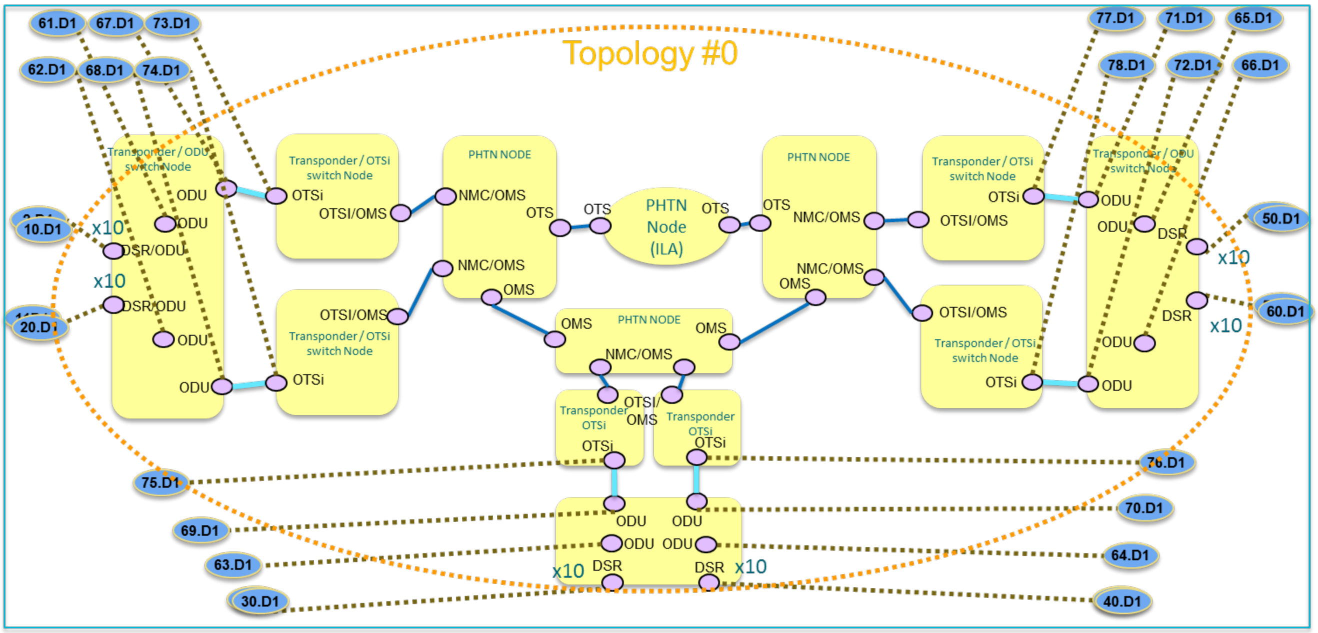


Figure 4-3 NS-1. T0: TAPI Topology Flat Abstraction model, transitional link approach.

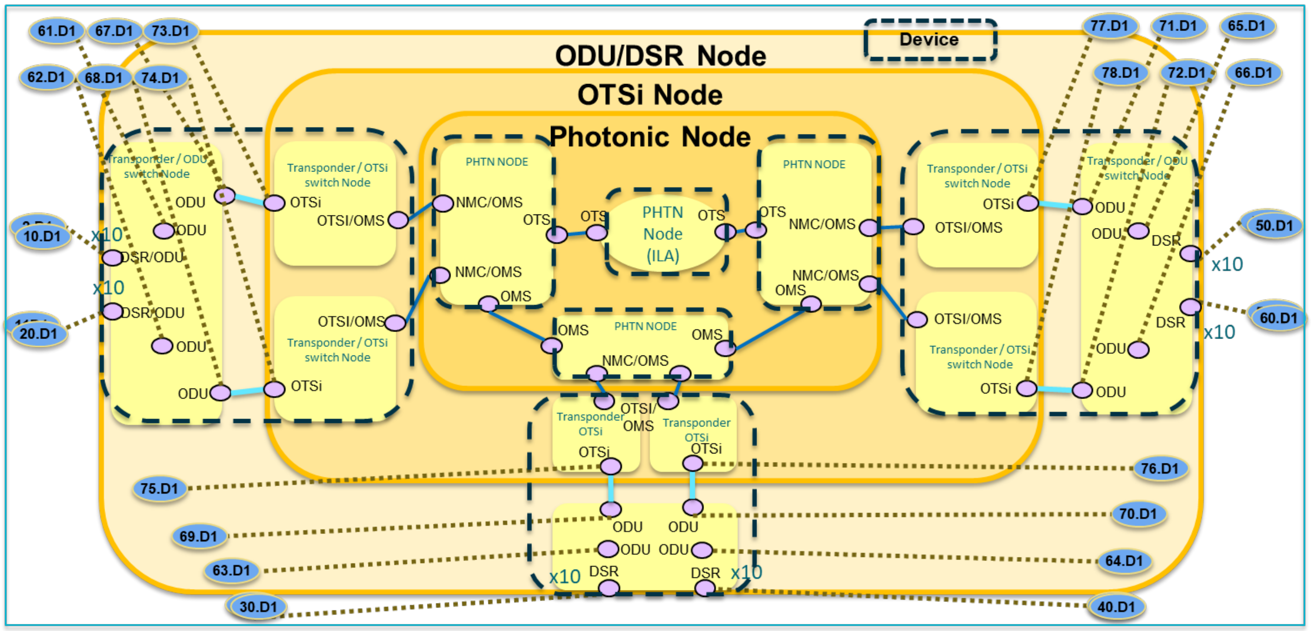


Figure 4-4 NS-1. T0: TAPI Topology Flat Abstraction model, transitional link approach (Device view).

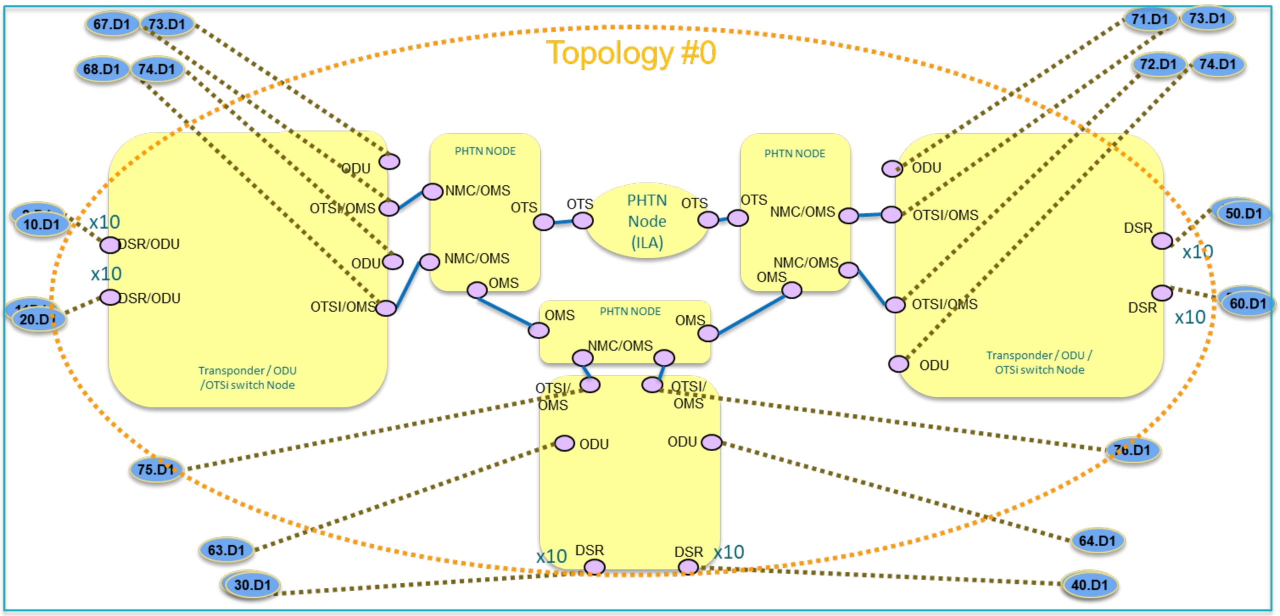


Figure 4-5 NS-1.T0: TAPI Topology Flat Abstraction model multi-layer node approach.

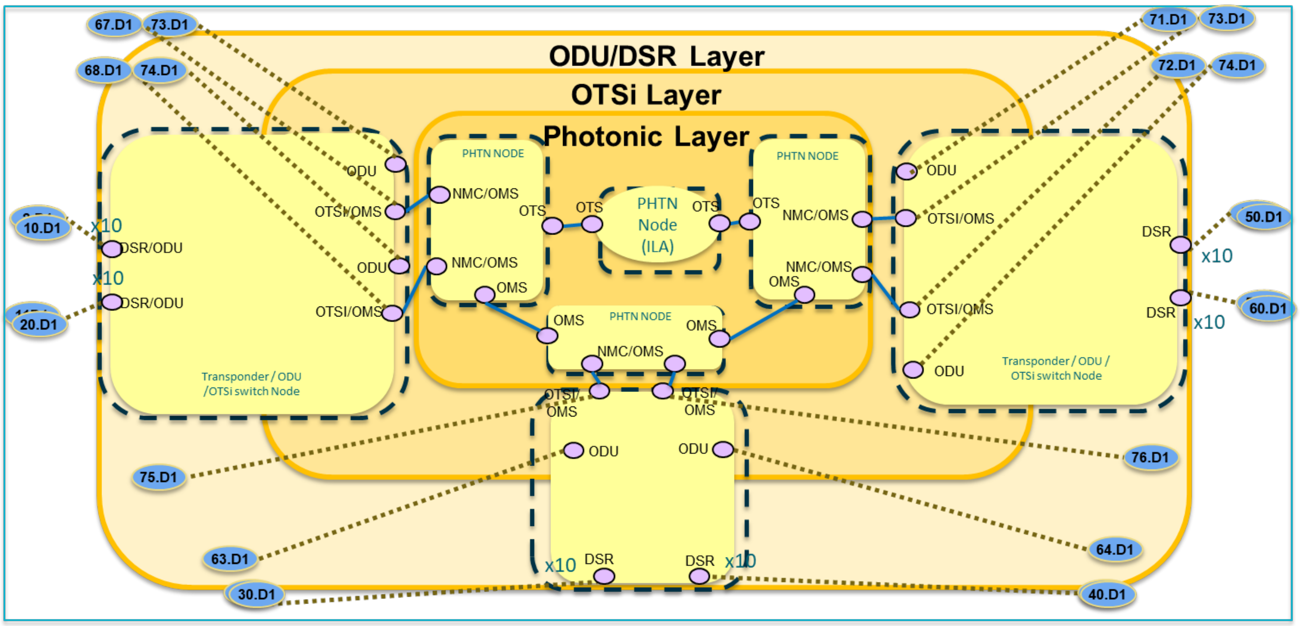


Figure 4-6 NS-1.T0: TAPI Topology Flat Abstraction model multi-layer node approach (Device view).

4.3.2 Scenario 2: Point-to-point DWDM link + Mesh DWDM network

The second scenario consists of two separated networks: one network is a point-to-point DWDM link separated by 2 In-Line-Amplifiers (ILAs) and terminated in Fixed OADM structures consisting on Mux/Demux; and the second is a similar three-ROADM network this time equipped with Transponder/Muxponder cards, the two transponders present in the network are connected to a Line Side Optical Line Protection (OLP) structures to provide optical 1+1 protection resilient schema.

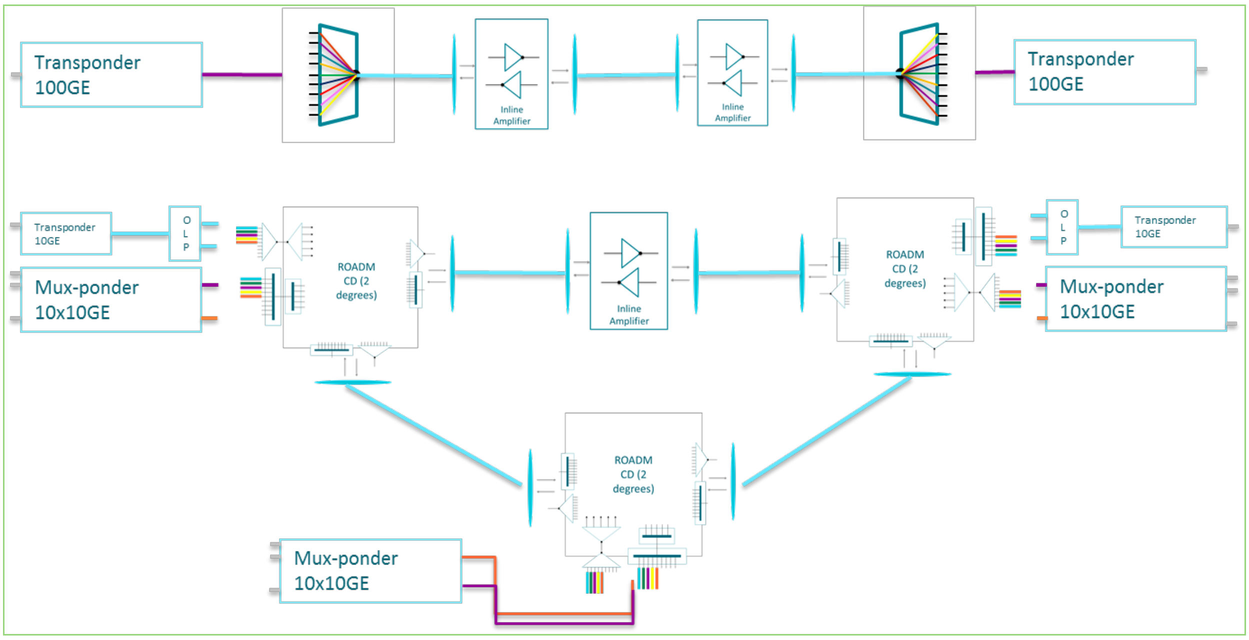


Figure 4-7 NS-2: OTN/WDM Network scenario 2.

4.3.2.1 Model representation

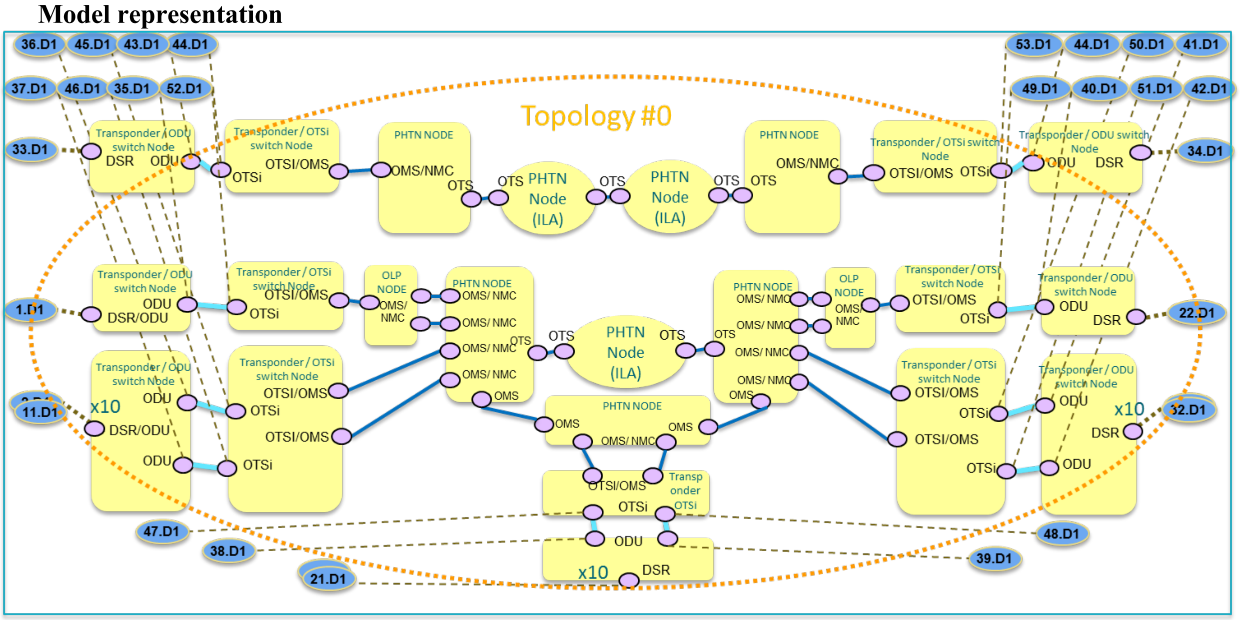


Figure 4-8 NS-2. T0: TAPI Topology Flat Abstraction Transitional Link model.

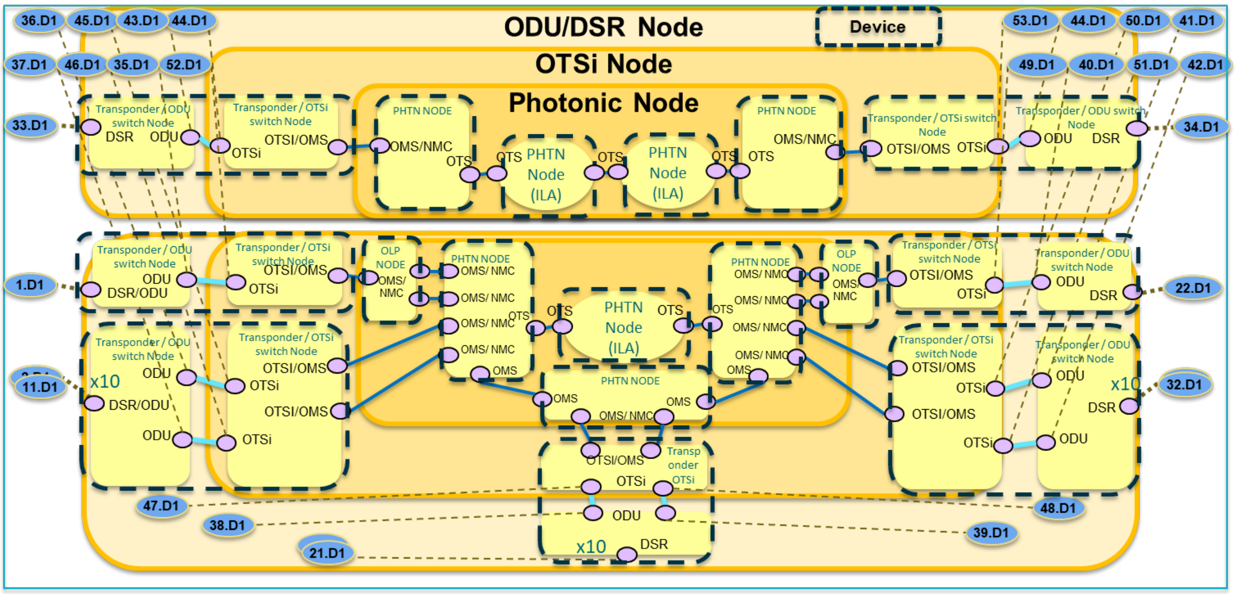


Figure 4-9 NS-2. T0: TAPI Topology Flat Abstraction Transitional Link model (Device view).

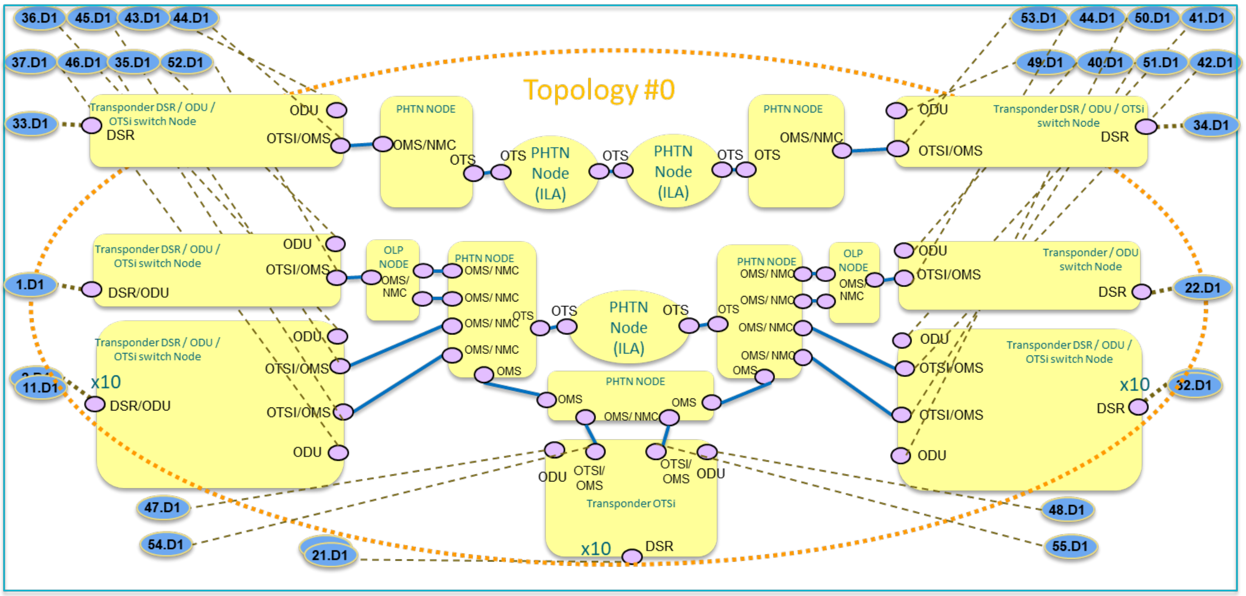


Figure 4-10 NS-2. T0: TAPI Topology Flat Abstraction Multi-Layer Node model.

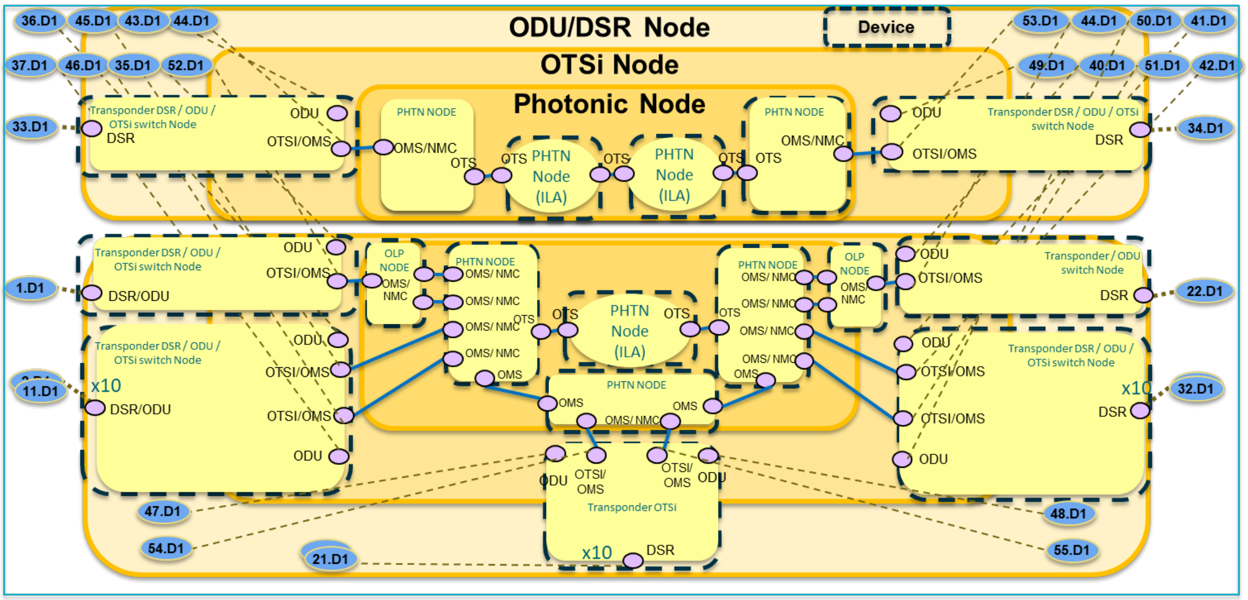


Figure 4-11 NS-2. T0: TAPI Topology Flat Abstraction Multi-Layer Node model (Device view).

5 Connectivity service model

In this chapter the complete connectivity service model will be described. The intention is to clarify some gaps which might not be clear just by reading the current description included in TAPI YANG models which need to be enhanced in order to provide a uniform understanding on the use of the TAPI information models. Thus, in this section a set of reference design guidelines are stated in order to constrain the possibilities or interpretations of the current proposed models.

The topology absence model is excluded. TAPI model covers connectivity-service without connections but in this reference implementation agreement this option is not covered.

5.1 Model guidelines

The following guidelines MUST be met by all implementations following the current specification.

[TAPI-CONN-MODEL-REQ-1] The solution exposing the proposed NBI based on RESCONF/TAPI MUST expose the capability of creating Connectivity-Service at all proposed network layers.

- **DSR Layer:** Models a Digital Signal of an unspecified format. This value can be used when the intent is to represent a generic digital layer signal without making any statement on its format or overhead (processing) capabilities.
- **ODU Layer:** Models the ODU layer as per described in [ITU-T G.709].
- **PHOTONIC_MEDIA Layer:** Models the OTSi,/ OTSiA, and Media Channels (OTSIMC, MC) as per described in [ITU-T G.872].

[TAPI-CONN-MODEL-REQ-2] The connectivity model MUST include the concept of **Top Connection,** which is defined as end-to-end connections between CEPs within the same layer which may belong to different Forwarding-Domains (TAPI Nodes). They can be either terminated ("infrastructure trails") or non-terminated (connecting client signals). Please see section 3.2.4.2 for more complete information.

[TAPI-CONN-MODEL-REQ-3] A ***tapi-connectivity:connectivity-service*** MUST, after being successfully provisioned by the TAPI Server, always include a reference to all Top Connection ***tapi-connectivity:connection*** objects supporting/composing the CS between its Connectivity-Service-End-Points (CSEPs), within its connection list (***tapi-connectivity:connectivity-service/connection***). These connections describe the end-to-end connectivity across the network at every network layers traversed by the connectivity-service (represented as the combination of the ***tapi-common:layer-protocol-name*** and ***tapi-common: layer-protocol-qualifier*** attributes).

```
module: tapi-connectivity
  augment /tapi-common:context:
      +--rw connectivity-context
            +--rw connectivity-service* [uuid]
            |  +--ro connection* [connection-uuid]
            |  |  +--ro connection-uuid -> /tapi-common:context/tapi-
connectivity:connectivity-context/connection/uuid
```

[TAPI-CONN-MODEL-REQ-4] The Top Connection object MUST represent how the requested service has been implemented within a network layer. It shall include the ***tapi-connectivity:connection/route*** object containing the list of connection-end-points (CEPs) traversed by the service at that layer.

```
module: tapi-connectivity
```

```
  augment /tapi-common:context:
      +--rw connectivity-context
          +--ro connection* [uuid]
             +--ro route* [local-id]
          |  +--ro connection-end-point* [topology-uuid node-uuid node-edge-point-uuid
connection-end-point-uuid]
          |  |  +--ro topology-uuid               -> /tapi-common:context/tapi-
topology:topology-context/topology/uuid
          |  |  +--ro node-uuid                   -> /tapi-common:context/tapi-
topology:topology-context/topology/node/uuid
          |  |  +--ro node-edge-point-uuid        -> /tapi-common:context/tapi-
topology:topology-context/topology/node/owned-node-edge-point/uuid
          |  |  +--ro connection-end-point-uuid   -> /tapi-common:context/tapi-
topology:topology-context/topology/node/owned-node-edge-point/tapi-connectivity:cep-
list/connection-end-point/uuid
          |  +--ro local-id              string
          |  +--ro name* [value-name]
          |     +--ro value-name    string
             |     +--ro value? string
```

[TAPI-CONN-MODEL-REQ-5] The ***tapi-connectivity:connection/route*** is modelled as a YANG List object of CEPs which is, by the default, ordered by the system (i.e., the TAPI server which produces it). The TAPI Server SHALL maintain the logical order of the CEP, however the absolute source of truth, to infer the logical order of the connection-end-points within the Route object by the TAPI client by the knowledge of the Topology information (links) and the NEP and CEP associations, which MUST univocally represent the correct sequence of CEPs for each Top Connection.

[TAPI-CONN-MODEL-REQ-6] The Top Connection MUST include a reference to all the lower connections generated in the in the same network layer and rate. These references MUST be included within the ***tapi-connectivity:connection/tapi-connectivity:lower-connection*** list. Please note that the use of the lower-connection attribute is used to represent the partitioning of the Top Connection and does not introduce any layering relationship.

```
module: tapi-connectivity
  augment /tapi-common:context:
      +--rw connectivity-context
          +--ro connection* [uuid]
             +--ro lower-connection* [connection-uuid]
             |   +--ro connection-uuid -> /tapi-common:context/tapi-
connectivity:connectivity-context/connection/uuid
```

[TAPI-CONN-MODEL-REQ-7] Top Connections MAY represent two different cases:

- **Non-terminated Top Connections**: between CEPs with a parent-NEPs (**tapi-topology:owned-node-edge-point/tapi-connectivity:cep-list/connection-end-point/parent-node-edge-point**) directly associated to the SIPs which has been referenced by the Service-End-Points of the Connectivity-Service associated to this Top Connection.

```
augment /tapi-common:context/tapi-topology:topology-context/tapi-
topology:topology/tapi-topology:node/tapi-topology:owned-node-edge-point:
    +--ro cep-list
       +--ro connection-end-point* [uuid]
          +--ro parent-node-edge-point
          |  +--ro topology-uuid?          -> /tapi-common:context/tapi-
topology:topology-context/topology/uuid
```

```
          |  +--ro node-uuid?              -> /tapi-common:context/tapi-
topology:topology-context/topology/node/uuid
          |  +--ro node-edge-point-uuid?   -> /tapi-common:context/tapi-
topology:topology-context/topology/node/owned-node-edge-point/uuid
```

- **Infrastructure Trails as defined in [ITU-T g.805]**: between CEPs representing Trail Termination Points (TTPs) which handover a signal of a given layer to a higher layer. These CEPs also produce associated client-NEPs (**tapi-topology:owned-node-edge-point/tapi-connectivity:cep-list/connection-end-point/client-node-edge-point**), to represent the generated pool of resources at a higher network layer or rate. E.g., an ODUk CEP producing a lower order ODUj NEP or an ODUk CEP producing DSR NEP.

```
augment /tapi-common:context/tapi-topology:topology-context/tapi-
topology:topology/tapi-topology:node/tapi-topology:owned-node-edge-point:
    +--ro cep-list
       +--ro connection-end-point* [uuid]
          +--ro client-node-edge-point* [topology-uuid node-uuid node-edge-point-uuid]
          |  +--ro topology-uuid           -> /tapi-common:context/tapi-
topology:topology-context/topology/uuid
          |  +--ro node-uuid               -> /tapi-common:context/tapi-
topology:topology-context/topology/node/uuid
          |  +--ro node-edge-point-uuid -> /tapi-common:context/tapi-topology:topology-
context/topology/node/owned-node-edge-point/uuid
```

5.1.1 Multi-layer connectivity service provisioning and connection generation

In the proposed model, the TAPI server MUST include a reference to each layer Top Connection within the Connectivity Service's Connection list as stated in 39[TAPI-CONN-MODEL-REQ-3].

The Connectivity Service routing across different layers MUST be inferred the Top-connections and its lower-connections, composing/supporting the connectivity-service (referenced within the ***tapi-connectivity:connectivity-service/connection*** list attribute) and by the tapi-topology <-> tapi-connectivity model relationships. These relationships are described in the following requirements:

[TAPI-CONN-MODEL-REQ-8] Every layer-protocol or layer-protocol-rate transition MUST be represented as a stack of ***tapi-topology:node-edge-point*** and ***tapi-connectivity:connection-end-points*** related to each other by ***tapi-connectivity:connection-end-point/parent-node-edge-point*** and ***tapi-connectivity:connection-end-point/client-node-edge-point*** attributes:

```
augment /tapi-common:context/tapi-topology:topology-context/tapi-
topology:topology/tapi-topology:node/tapi-topology:owned-node-edge-point:
    +--ro cep-list
       +--ro connection-end-point* [uuid]
          +--ro parent-node-edge-point
          |  +--ro topology-uuid?        -> /tapi-common:context/tapi-topology:topology-
context/topology/uuid
          |  +--ro node-uuid?            -> /tapi-common:context/tapi-topology:topology-
context/topology/node/uuid
          |      +--ro   node-edge-point-uuid?        ->   /tapi-common:context/tapi-
topology:topology-context/topology/node/owned-node-edge-point/uuid
          +--ro client-node-edge-point* [topology-uuid node-uuid node-edge-point-uuid]
          |  +--ro topology-uuid         -> /tapi-common:context/tapi-topology:topology-
context/topology/uuid
```

```
          |  +--ro node-uuid              -> /tapi-common:context/tapi-topology:topology-
context/topology/node/uuid
          |      +--ro   node-edge-point-uuid          ->   /tapi-common:context/tapi-
topology:topology-context/topology/node/owned-node-edge-point/uuid
```

Please note that the previous statement is valid for all layer-protocol and layer-protocol-rate transitions but for the HO-ODUk <-> OTSi layer transition which MAY be represented also by a ***tapi-topology:link*** object including a ***tapi-topology:transitioned-layer-protocol-name*** attribute as described in [TAPI-TOP-MODEL-REQ-9]. See a descriptive example in the following figure:

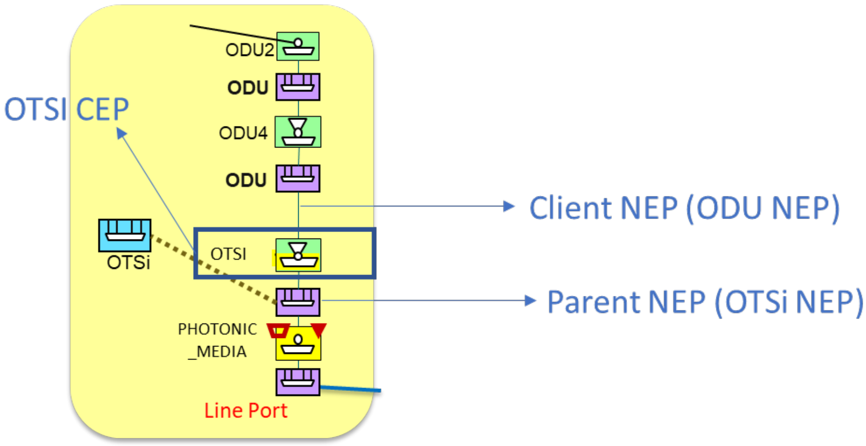


Figure 5-1 Client/parent NEP relations of CEP objects for multi-layer transitions representation.

[TAPI-CONN-MODEL-REQ-9] Additionally every ***tapi-topology:link*** object generated to represent the adjacency between every pair of NEPs connected by the sequence of cross-connections included as lower-connections of a Top Connection object MUST be referenced by the ***tapi-connectivity:connection/supported-client-link*** attribute. The rest of the section states in which conditions these links SHALL be generated.

```
module: tapi-connectivity
  augment /tapi-common:context:
   +--rw connectivity-context
       +--ro connection* [uuid]
                +--ro supported-client-link* [topology-uuid link-uuid]
                |  +--ro topology-uuid    -> /tapi-common:context/tapi-
topology:topology-context/topology/uuid
                |  +--ro link-uuid -> /tapi-common:context/tapi-topology:topology-
context/topology/link/uuid
```

Now, assuming a DSR Connectivity-Service has been requested between two SIPs associated to NEPs of the DSR layer, the multi-layer connections created are described by the following assumptions:

At DSR layer:

[TAPI-CONN-MODEL-REQ-10] The CS triggers the creation of the Top Connection at the DSR layer:

- Once DSR Top Connection is operational (***tapi-connectivity:connection/tapi-common:operational-state = ENABLED***), the DSR CS becomes operational too and includes DSR Top Connection within its connection list.

- DSR Top Connection MUST include the explicit route referencing CEPs associated to NEPs at the DSR layer.

[TAPI-CONN-MODEL-REQ-11] One or more, DSR XC Connections, describing the lower partitioning level of DSR Top Connection and MUST be included within its lower-connection list.

- When operational (***tapi-connectivity:connection/tapi-common:operational-state = ENABLED***), the DSR Top Connection becomes operational too.

At the ODU layer the CS triggers the creation of:

[TAPI-CONN-MODEL-REQ-12] 0-N Top Connections at the ODUj layers, which describes the multiplexing of Low Order (LO)-ODU signals into High Order (HO)-ODU signals.

- Once each 0-N Top Connections at the ODUj layers are operational (***tapi-connectivity:connection/tapi-common:operational-state = ENABLED***), all TOP connections MUST be included within the CS connection list.
- Each ODUj Top Connection MUST include the list of ODUj lower connections describing the lower partitioning level of ODUj Top Connection.
- When each ODUj Top Connection become operational (***tapi-connectivity:connection/tapi-common:operational-state = ENABLED***) in increasing order from lower ODU layer (higher ODU rate) to the DSR layer, the immediately upper layer adjacency is enabled (a higher layer NEP is created "over" the operational CEP) allowing the upper layer Top Connection to be realized.
- Moreover, when all ODUj Top Connections are operational, a new ***tapi-topology:link*** at the DSR layer (**layer-protocol-name=DSR**) **MAY be optionally** generated between the NEPs produced by the CEPs (Trail Termination Points) of the Top Most Connection with the ODUj Layer rate and referenced by the ***tapi-connectivity: supported-client-link*** attribute**.**

[TAPI-CONN-MODEL-REQ-13] One Top Connection at the HO-ODUk rate, which describe the highest order ODU (supported by this node) which transport by the optical OTSi layer.

- Once the HO-ODUk Top Connection is operational (***tapi-connectivity:connection/tapi-common:operational-state = ENABLED***), it MUST be included within the CS connection list.
- When the HO-ODUk Top Connection become operational (***tapi-connectivity:connection/tapi-common:operational-state = ENABLED***) the lower-rate ODU layer adjacency is enabled (client layer NEPs is created "over" the operational CEPs) allowing the upper layer Top Connection to be realized.

[TAPI-CONN-MODEL-REQ-14] Possibly multiple HO-ODUk XC Connections, describing the lower partitioning level of DSR Top Connection and they MUST be included within its lower-connection list.

- When all XCs become operational, the HO-ODUk Top Connection becomes operational too.

At the PHOTONIC_LAYER_QUALIFER_OTSI layer the CS triggers the creation of:

[TAPI-CONN-MODEL-REQ-15] One or more Top Connection at the OTSi layer between the two NEPs supporting the HO-ODUk NEPs.

- Once the OTSi Top Connection is operational (***tapi-connectivity:connection/tapi-common:operational-state = ENABLED***), it MUST be included within the CS connection list.
- ***If the multi-layer node modelling approach defined in [TAPI-TOP-MODEL-REQ-10] is followed*** (OTSI<>ODU layer transition is represented by the NEPs <> CEPs stack relationship instead of transitional links) and related HO-ODUk NEPs are not present in the multi-layer node, then these NEPs MUST be generated to allow the HO-ODUk Top Connection to be realized.
- When the OTSi Top Connection becomes operational, an ODU ***tapi-topology:link*** between the related HO-ODUk NEPs **MAY be optionally** generated, representing the potential adjacency between these two NEPs at the HO-ODUk layer/rate. The new generated link MUST be referenced by the Top Connection, which realizes it, as a ***tapi-connectivity: supported-client-link.***
- OTSi/OTSiA Top Connection MUST include the explicit route referencing CEPs associated to NEPs of PHOTONIC_MEDIA Layer with OTSI supported-layer-qualifier.

[TAPI-CONN-MODEL-REQ-16] Possibly multiple OTSi/OTSiA, XC Connections, describing the lower partitioning level of OTSI Top Connection and MUST be included within its lower-connection list.

- When operational, the OTSi/OTSiA Top Connection becomes operational too.
- ***If the multi-layer node modelling approach defined in [TAPI-TOP-MODEL-REQ-10] is followed*** (OTSI<>ODU layer transition is represented by the NEPs <> CEPs stack relationship instead of transitional links) OTSi XC Connections may not be generated, nor referenced by the lower-connection list. In this case, the Top Connection's route MUST consists of the CEPs referenced at **tapi-connectivity:connection/connection-end-point** list.

At the Photonic Media Layer, the CS triggers the creation of one or more PHOTONIC_LAYER_QUALIFER_MC and optionally a PHOTONIC_LAYER_QUALIFER_OTSIMC Top Connections at the Media-Channel layer.

[TAPI-CONN-MODEL-REQ-17] One MC Top Connection at the Photonic Media layer between the two CEPs supported by the MC NEPs facing the OTSi NEPs. At server PHOTONIC_MEDIA layer, the NEPs are linked by PHOTONIC_MEDIA link.

- Once the MC Top Connection is operational (***tapi-connectivity:connection/tapi-common:operational-state = ENABLED***), it MUST be included within the CS connection list.
- When the MC Top Connection becomes operational (***tapi-connectivity:connection/tapi-common:operational-state = ENABLED***), an OTSi ***tapi-topology:link*** between the OTSi NEPs facing the MC NEPs **MAY be optionally** generated, representing the potential adjacency between these two NEPs at the OTSi layer. Thus, allowing ALL OTSi Top Connections supported by the MC to be realized. The new generated link[6] MUST be referenced by the CEPs of the Top Connection which generates it as a ***tapi-connectivity:supported-client-link.***
- MC Top Connection MUST include the explicit route referencing CEPs associated to NEPs of PHOTONIC_MEDIA Layer with MC supported-layer-qualifier.

[6] Please note this may not be generated in case of an Open Line System disaggregated scenario where OTSi layer NEP facing the MC NEPs are not managed by the TAPI Server.

[TAPI-CONN-MODEL-REQ-18] Possibly multiple MC XC Connections, describing the lower partitioning level of MC Top Connection and MUST be included within its lower-connection list.

- When all XC become operational, the MC Top Connection becomes operational too.

OTSiMC layer representation, including Top Connections, XCs and CEPs, is only required to represent monitoring capabilities at the filters but not switching (switching is just happening at the lower MC layer). Thus, the representation of the OTSiMC CEPs is optional and it depends upon the monitoring capabilities of the photonic_media layer filters at the OTSiMC level (signal positioning within the actual spectrum filtered MC).

[TAPI-CONN-MODEL-REQ-19] One or more OTSiMC Top Connections (depending on the number of OTSI signals composing the OTSiA) at the Photonic Media layer between the two client NEPs generated by MC CEPs logically linked to PHOTONIC_MEDIA Add/Drop ports connected to the OT's line ports.

- Once the each OTSiMC Top Connection is operational (***tapi-connectivity:connection/tapi-common:operational-state = ENABLED***), they MUST be included within the CS connection list.
- Each OTSiMC Top Connection MUST include the explicit route referencing CEPs associated to NEPs of PHOTONIC_MEDIA Layer with OTSiMC supported-layer-qualifier.

[TAPI-CONN-MODEL-REQ-20] Possibly multiple OTSiMC XC Connections, describing the lower partitioning level of OTSiMC Top Connection and MUST be included within its lower-connection list.

OTSiMC CEPs are present for two purposes, forwarding and monitoring. In the photonic model is never relevant for forwarding (strict fate sharing with MC), but only for monitoring.

- When all XC become operational, the OTSiMC Top Connection becomes operational too.

Following a complete generic example of a DSR connectivity-service is presented, to clearly identify the connection association hierarchy described by the previous set of requirements.


```
{"context":{
      "connectivity-context":{
            "connectivity-service":[
                  {"uuid":"CS_UUID",
                   "end-point":[
{"local_id":"LOCAL_ID_A","service-interface-point":{"service-interface-point-
uuid":"SIP_UUID_A"}},
{"local_id":"LOCAL_ID_B","service-interface-point":{"service-interface-point-
uuid":"SIP_UUID_B"}}],
                   "connection":[
                              {"connection_uuid":"DSR_TOP_1"},
                              {"connection-uuid":"ODUj_TOP_1"},
                              …
                              {"connection-uuid":"ODUj+N_TOP_N"},
                              {"connection-uuid":"HO-ODUk_TOP_1"},
                              {"connection-uuid":"OTSi_TOP_1"},
                              {"connection-uuid":"OTSIMC_TOP_1"},
                              {"connection-uuid":"MC_TOP_1"}
                  ]}
            ],

            "connection":[
                  {"uuid": "DSR_TOP_1",
```

```
                 "lower-connection":[
                      {"connection-uuid":"DSR_XC_1"},
                      {"connection-uuid":"DSR_XC_2"}
                 ]},
                {"uuid": "ODUj_TOP_1",
                 "lower-connection":[
                      {"connection-uuid":"ODUj_XC_1"},
                      {"connection-uuid":"ODUj_XC_2"},
                 ]},

                … (repeated for N ODUj layer rates)

                {"uuid": "ODUj_TOP_N",
                 "lower-connection":[
                      {"connection-uuid":"ODUj_XC_1"},
                      {"connection-uuid":"ODUj_XC_2"},
                 ]},
                {"uuid": "HO-ODUk_TOP_1",
                 "lower-connection":[
                      {"connection-uuid":"HO-ODUk_XC_1"},
                      {"connection-uuid":"HO-ODUk_XC_2"}
                 ]},
                {"uuid": "OTSi_TOP_1",
                 "lower-connection":[
                      {"connection-uuid":"OTSi_XC_1"},
                      {"connection-uuid":"OTSi_XC_2"}
                 ]},
                {"uuid": "OTSiMC_TOP_1",
                 "lower-connection":[
                      {"connection-uuid":"OTSiMC_XC_1"},
                      {"connection-uuid":"OTSiMC_XC_2"},
                            …
                      {"connection-uuid":"OTSiMC_XC_N"}
                 ]} ,
                {"uuid": "MC_TOP_1",
                 "lower-connection":[
                      {"connection-uuid":"MC_XC_1"},
                      {"connection-uuid":"MC_XC_2"},
                            …
                      {"connection-uuid":"MC_XC_N"}
                 ]}

          ]}
     }
}
```


5.1.2 Resiliency mechanism at connectivity service

[TAPI-CONN-MODEL-REQ-21] To implement different mechanism of protection the TAPI Server MUST support the following protection and restoration policies (***tapi-topology:protection-type***) at the Connectivity Service level:

- ONE_PLUS_ONE_PROTECTION
- ONE_PLUS_ONE_PROTECTION_WITH_DYNAMIC_RESTORATION
- ONE_PLUS_ONE_PROTECTION_WITH_PRE_COMPUTED_RESTORATION
- PERMANENT_ONE_PLUS_ONE_PROTECTION

- ONE_FOR_ONE_PROTECTION
- DYNAMIC_RESTORATION
- PRE_COMPUTED_RESTORATION

```
module: tapi-connectivity
  augment /tapi-common:context:
    +--rw connectivity-context
       +--rw connectivity-service* [uuid]
       |  +--rw resilience-type
       |  |  +--rw restoration-policy?   restoration-policy
       |  |  +--rw protection-type? protection-type
```

[TAPI-CONN-MODEL-REQ-22] The TAPI server, for all protected services with restoration capabilities, SHALL implement the PER_DOMAIN_RESTORATION policy by default, which implies it is responsible of activating the required control mechanisms to guarantee the restoration of the service autonomously.

[TAPI-CONN-MODEL-REQ-23] At the Connection level, the switch control among lower-connections, which implements the route diversity for the different levels of protection policies listed in [TAPI-CONN-MODEL-REQ-21], MUST be implemented by the TAPI server. The TAPI server MUST be able to describe these mechanisms by the ***tapi-connectivity:connection/switch-control.*** The specific switch-control mechanisms for each protection schema will be detailed in Section 6

```
module: tapi-connectivity
  augment /tapi-common:context:
    +--rw connectivity-context
       +--rw connection* [uuid]
          +--ro switch-control* [uuid]
          |  +--ro sub-switch-control* [connection-uuid switch-control-uuid]
          |  |  +--ro connection-uuid        -> /tapi-common:context/tapi-
connectivity:connectivity-context/connection/uuid
          |  |  +--ro switch-control-uuid    -> /tapi-common:context/tapi-
connectivity:connectivity-context/connection/switch-control/uuid
          |  +--ro switch* [local-id]
          |  |  +--ro selected-connection-end-point* [topology-uuid node-uuid node-
edge-point-uuid connection-end-point-uuid]
          |  |  |  +--ro connection-end-point-uuid    -> /tapi-common:context/tapi-
topology:topology-context/topology/node/owned-node-edge-point/tapi-connectivity:cep-
list/connection-end-point/uuid
          |  |  +--ro selected-route* [connection-uuid route-local-id]
          |  |  |  +--ro connection-uuid    -> /tapi-common:context/tapi-
connectivity:connectivity-context/connection/uuid
          |  |  |  +--ro route-local-id     -> /tapi-common:context/tapi-
connectivity:connectivity-context/connection/route/local-id
          |  |  +--ro selection-control?  selection-control
          |  |  +--ro selection-reason?   selection-reason
          |  |  +--ro switch-direction?   tapi-common:port-direction
          |  |  +--ro local-id                         string
          |  |  +--ro name* [value-name]
          |  +--ro uuid                                  uuid
          |  +--ro name* [value-name]
          |  +--ro resilience-type
          |  |  +--ro restoration-policy?   restoration-policy
          |  |  +--ro protection-type?      protection-type
          |  +--ro restoration-coordinate-type?          coordinate-type
```

```
          |  +--ro restore-priority?                          uint64
          |  +--ro reversion-mode?                            reversion-mode
          |  +--ro wait-to-revert-time?                       uint64
          |  +--ro hold-off-time?                             uint64
          |  +--ro is-lock-out?                               boolean
          |  +--ro is-frozen?                                 boolean
          |  +--ro is-coordinated-switching-both-ends?   boolean
          |  +--ro max-switch-times?                          uint64
          |  +--ro preferred-restoration-layer* tapi-common:layer-protocol-name
```

5.1.3 Topology and service constrains for connectivity services

[TAPI-CONN-MODEL-REQ-24] In order to implement different use cases that imply the constrain of the service path along the network topology, the following attributes of the ***tapi-onnectivity:connectivity-service*** object MUST be supported.

```
module: tapi-connectivity
  augment /tapi-common:context:
    +--rw connectivity-context
      +--rw connectivity-service* [uuid]
      |  +--rw coroute-inclusion
      |  |  +--rw connectivity-service-uuid?   -> /tapi-common:context/tapi-
connectivity:connectivity-context/connectivity-service/uuid
      |  +--rw diversity-exclusion* [connectivity-service-uuid]
      |  |  +--rw connectivity-service-uuid -> /tapi-common:context/tapi-
connectivity:connectivity-context/connectivity-service/uuid
      |  +--rw diversity-policy? diversity-policy
      |  +--rw include-topology*                        tapi-common:uuid
      |  +--rw avoid-topology*                          tapi-common:uuid
      |  +--rw include-path*                            tapi-common:uuid
      |  +--rw exclude-path*                            tapi-common:uuid
      |  +--rw include-link*                            tapi-common:uuid
      |  +--rw exclude-link*                            tapi-common:uuid
      |  +--rw include-node*                            tapi-common:uuid
      |  +--rw exclude-node*                            tapi-common:uuid
```

6 Use cases Low Level designs (LLDs)

6.1 Topology and services discovery

This use case consists on retrieving all information available from TAPI servers (SDN-C) including service-interface-points and topology. Is intended to be performed by any NBI client controller, module or application which intends to discover the logical representation of the network done by the SDN-C.

There are two modes of operations foreseen for this use case:

- **Polling mode** - based on periodic pooling retrieval operations and after each service creation to reconcile the actual state of the network.
- **Event triggered mode (Notifications)** - based on an initial proactive synchronization done from the NBI client module and a connection oriented notification subscription session based on the NBI Notification mechanism described in section 2.6 and 3.4.

In this version at least the pooling mechanism is foreseen to be fully supported by any provider SDN-C compliant with this specification.

The use case is divided in two

- Use Case 0a: Context & Service Interface Points discovery
- Use Case 0b: Topology discovery

6.1.1 Use Case 0a: Context & Service Interface Points discovery (polling mode)

Number	**UC0a**
Name	**Context & Service Interface Points discovery (polling mode)**
Technologies involved	Optical
Process/Areas Involved	Planning and Operations
Brief description	The TAPI Context and Service Interface Points are the relevant network service information required before any connectivity-service creation operation. The discovery of this information is intended to be requested periodically and/or on-demand basis, proactively from the TAPI client role, to synchronize the context information.
Layers involved	DSR/ODU/PHOTONIC_MEDIA
Type	Discovery
Description & Workflow	This use case consists on retrieving context and service-interface-point information (Figure 6-1). If the first operation (1) is correctly supported by the NBI server, it MUST retrieve the context filtered subtree up to three levels down into the hierarchy (2). The response operation MUST contain the attributes included in Table 6 which are marked as Mandatory (M) in the Support (Sup) column. The second operation (3) retrieves the list of service-interface-point (SIP) “uuid” (4), to recursively retrieve the full content of each SIP object in operation (5) which employs the “fields” query parameter to obtain only the desired filtered information. The response operation (6) MUST contain the attributes included in Table 7 which are marked as Mandatory (M) in the Support (Sup) column.

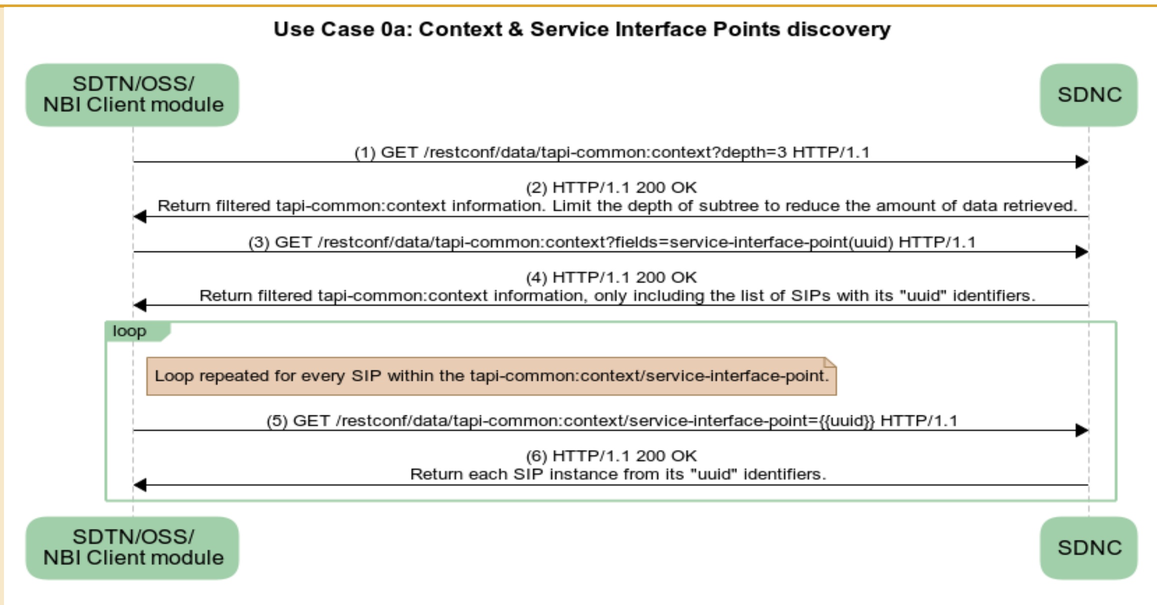


Figure 6-1 UC-0a: Context and Service Interface Point - LLD Workflow.

6.1.1.1 Required parameters

Following the required parameters for each object which is retrieved in this use case.

Table 6: Context object definition

Context	**/tapi-common:context**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
uuid	"[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}"	RO	M	• As per RFC 4122 • Provided by *tapi-server*
name	List of {value-name, value} • "value-name": "CONTEXT_NAME" "value": " [0-9a-zA-Z_]{64} • "value-name": "VENDOR_NAME" "value": "[0-9a-zA-Z_]{64}"	RO	M	• Provided by *tapi-server* • CONTEXT_NAME is a user legible unstructured string tag to uniquely identify the tapi-server context. • VENDOR_NAME is a user legible unstructured string tag to uniquely identify the tapi-server owner or supplier.
service-interface-point	List of {service-interface-point}	RO	M	• Provided by *tapi-server* • Direct modification disallowed
notification-context	• List of {notif-subscription} • List of {notification}	RO	O	• Provided by *tapi-server*
topology-context	• {network-topology-service} • List of {topology}	RO	M	• Provided by *tapi-server*
connectivity-context	• List of {**connectivity-service**} • List of {**connection**}	RO	M	• Provided by *tapi-server*

Table 7: Service Interface Point (SIP) object definition

Service-interface-point	**/tapi-common:context/service-interface-point**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
uuid	"[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}"	RW	M	• As per RFC 4122 • Provided by *tapi-server*
name	List of {value-name, value} • "value-name": "INVENTORY_ID", "value": " [0-9a-zA-Z_]{64}"	RW	M	• Initial value provided by *tapi-server* • INVENTORY_ID format is described in section 4.2 • Subsequent updates provided by *tapi-client*
layer-protocol-name	["DSR", "ETH", "ODU", "PHOTONIC_MEDIA"]	RO	M	
supported-layer-protocol-qualifier	List of ["DIGITAL_SIGNAL_TYPE", "ODU_TYPE", "PHOTONIC_LAYER_QUALIFIER"]	RO	M	• Provided by tapi-server • All children identities defined for ["DIGITAL_SIGNAL_TYPE", "ODU_TYPE", "PHOTONIC_LAYER_QUALIFIER"] MUST be supported.
administrative-state	["UNLOCKED", "LOCKED"]	RW	M	• Initial value provided by *tapi-server* • Subsequent updates provided by *tapi-client*
operational-state	["ENABLED", "DISABLED"]	RO	M	• Provided by *tapi-server*
lifecycle-state	["PLANNED", "POTENTIAL_AVAILABLE", "POTENTIAL_BUSY", "INSTALLED", "PENDING_REMOVAL"]	RO	M	• Provided by *tapi-server*
total-potential-capacity	"total-size": {value, unit} • "value": "[0-9]{8}", • "unit": ["TB", "TBPS", "GB", "GBPS", "MB", "MBPS", "KB", "KBPS"]	RO	M	• Provided by *tapi-server*
available-capacity	"total-size": {value, unit} • "value": "[0-9]{8}", • "unit": ["TB", "TBPS", "GB", "GBPS", "MB", "MBPS", "KB", "KBPS"]	RO	M	• Provided by *tapi-server*
tapi-photonic-media:otsi-service-interface-point-spec/ otsi-capability	• List of "supportable-central-frequency-spectrum-band":{ lower/upper-central-frequency, frequency-constraint: {adjustment-granularity, grid-type} } ○ "central-frequency": " [0-9]{9}", ○ "adjustment-granularity":["UNCONSTRAINED", "G_3_125GHZ", "G_6_25GHZ", "G_12_5GHZ", "G_25GHZ", "G_50GHZ", "G_100GHZ",] ○ "grid-type": ["GRIDLESS", "FLEX", "CWDM", "DWDM"]	RO	M	• Provided by *tapi-server* • This block of attributes MUST augment SIPs attached to PHOTONIC_MEDIA NEPs exposing OTSI/OTSiG service provisioning capabilities. • The lower/upper-**central-frequency** of the laser specified in MHz. Those correspond to the lower and upper central frequencies of the band. • **frequency-constraint** specify the rest of feasible central frequencies within the band. • **Adjustment-granularity** in Gigahertz. As per ITU-T G.694.1, it

	• List of "supportable-application-identifier":{ application-identifier-type, application-code} ○ "application-identifier-type":["PROPRIETARY", "ITUT_G959_1", "ITUT_G698_1", "ITUT_G698_2", "ITUT_G696_1", "ITUT_G695",] ○ "application-code": " [0-9a-zA-Z_]{64}" • List of "supportable-modulation":["RZ", "NRZ", "BPSK", "DPSK", "QPSK", "8QAM", "16QAM"] • "total-power-warn-threshold": {total-power-lower-warn-threshold-default /min/max, total-power-upper-warn-threshold-default /min/max"} ○ "total-power-*-warn-threshold": "[0-9].[0-9]{7}"			is used to calculate nominal central frequency". • The **grid-type** specifies the reference set of frequencies used to denote allowed nominal central frequencies that may be used for defining applications.
tapi-photonic-media:otsi-service-interface-point-spec/ power-management-capability	• "supportable-maximum-output-power":{total-power, power-spectral-density} ○ "Total-power":"[0-9].[0-9]{64}", ○ "power-spectral-density":"[0-9].[0-9]{64}" • "supportable-minimum-output-power":{total-power, power-spectral-density} ○ "Total-power":"[0-9].[0-9]{64}", ○ "power-spectral-density":"[0-9].[0-9]{64}" • "tolerable-maximum-output-power":{total-power, power-spectral-density} ○ "Total-power":"[0-9].[0-9]{64}", ○ "power-spectral-density":"[0-9].[0-9]{64}" • "tolerable-minimum-output-power":{total-power, power-spectral-density} ○ "Total-power":"[0-9].[0-9]{64}", ○ "power-spectral-density":"[0-9].[0-9]{64}"	RW	M	• Provided by *tapi-server* • Can be updated by the *tapi-client*
tapi-photonic-media: media-channel-service-interface-point-spec/mc-pool	• List of "supportable/available/occupied-spectrum":{upper/lower-frequency, frequency-constraint: {adjustment-granularity, grid-type} } ○ "upper/lower-frequency": " [0-9]{9}", ○ "adjustment-granularity":["UNCONSTRAINED", "G_3_125GHZ", "G_6_25GHZ",	RO	M	• Provided by *tapi-server* • This block of attributes MUST augment SIPs attached to PHOTONIC_MEDIA NEPs exposing Media_Channel service provisioning capabilities. • The **upper/lower-frequency** bound of the media channel spectrum specified in MHz.. • **Adjustment-granularity** in Gigahertz. As per ITU-T G.694.1, it is used to calculate nominal central frequency".

	“G_12_5GHZ”, “G_25GHZ”, “G_50GHZ”, “G_100GHZ”,] ○ “grid-type”: [“GRIDLESS”, “FLEX”, “CWDM”, “DWDM”]			• The **grid-type** specifies the reference set of frequencies used to denote allowed nominal central frequencies that may be used for defining applications.
tapi-photonic-media: media-channel-service-interface-point-spec/power-management-capability	• "supportable-maximum-output-power":{total-power, power-spectral-density} ○ "Total-power":"[0-9].[0-9]{64}", ○ "power-spectral-density":"[0-9].[0-9]{64}" • "supportable-minimum-output-power":{total-power, power-spectral-density} ○ "Total-power":"[0-9].[0-9]{64}", ○ "power-spectral-density":"[0-9].[0-9]{64}" • "tolerable-maximum-output-power":{total-power, power-spectral-density} ○ "Total-power":"[0-9].[0-9]{64}", ○ "power-spectral-density":"[0-9].[0-9]{64}" • "tolerable-minimum-output-power":{total-power, power-spectral-density} ○ "Total-power":"[0-9].[0-9]{64}", ○ "power-spectral-density":"[0-9].[0-9]{64}"	RW	M	• Provided by *tapi-server* • Can be updated by the *tapi-client*

6.1.2 Use Case 0b: Topology discovery (synchronous mode)

Number	**UC0b**
Name	**Topology discovery (synchronous mode)**
Technologies involved	Optical
Process/Areas Involved	Planning and Operations
Brief description	The TAPI Topology is the relevant network logical representation information required for inventory, traffic-engineering or provisioning purposes. The discovery of this information is intended to be requested periodically and/or on-demand basis, proactively from the TAPI client role, to synchronize the context information.
Layers involved	DSR/ODU/PHOTONIC_MEDIA
Type	Discovery
Description & Workflow	The topology discover use case consists on the workflow and operations depicted in Figure 6-2. The first operation (1) retrieves the list of Topology references (UUID) included in the ***tapi-topology:topology-context/nw-topology-service/*** (2). For each Topology reference found, operation (3) is repeated to obtain a Topology object instance filtered subtree up to three levels (4). The details of the Topology object mandatory attributes are included in Table 8. The response operation MUST contain the attributes included in Table 8 which are marked as Mandatory (M) in the Support (Sup) column. For each node found in operation (4), operation (5) is repeated to retrieve each ***tapi-topology:node*** object (6) which MUST contain the attributes included in Table 9 which are marked as Mandatory (M) in the Support (Sup) column. Recursively, for each node object a filtered operation (7) is used to retrieve each ***owned-node-edge-point/uuid*** reference. For each owned-node-edge-point reference discovered in operation (7), operation (9) is performed to retrieve each owned-node-edge-point object (10) for each node. Each ONEP object MUST contain the attributes included in Table 10 which are marked as Mandatory (M) in the Support (Sup) column. For each topology found in operation (4), operation (11) is repeated to retrieve each ***tapi-topology:link*** object (12) which MUST contain the attributes included in Table 13 which are marked as Mandatory (M) in the Support (Sup) column.

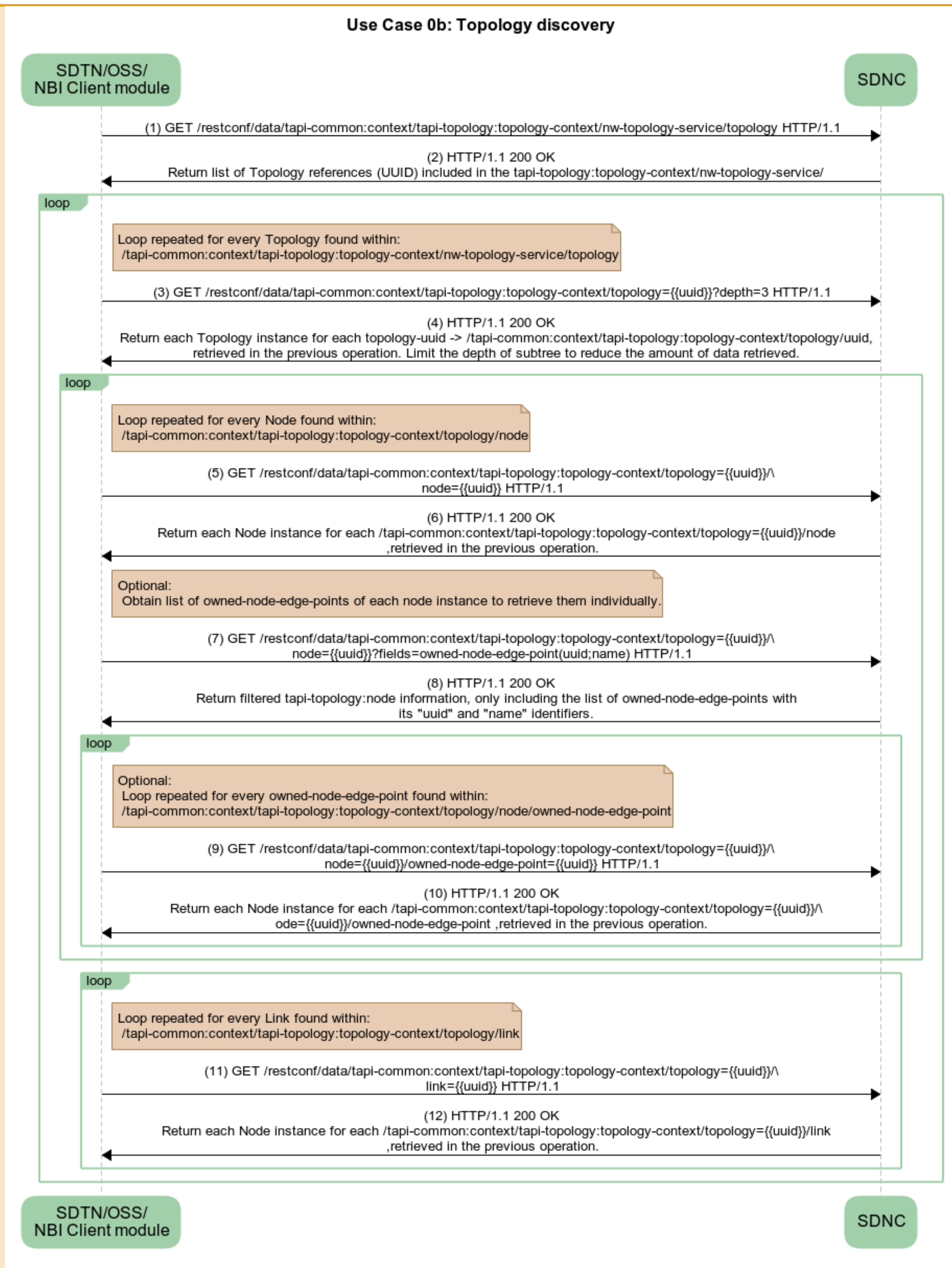


Figure 6-2 UC-0b: Topology discovery - LLD Workflow.

6.1.2.1 Required parameters

Following we include the required parameters for each object which is retrieved in the previously described RESTCONF operations.

Table 8: Topology object definition

Topology	**/tapi-common:context/tapi-topology:topology-context/topology**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
uuid	"[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}"	RO	M	• As per RFC 4122 • Provided by ***tapi-server***
name	List of {value-name: value} • "value-name": "TOPOLOGY_NAME" "value": " [0-9a-zA-Z_]{64}"	RO	M	• Provided by ***tapi-server*** • TOPOLOGY_NAME is a user legible unstructured string tag to uniquely identify the tapi-server topology.
layer-protocol-name	List of ["DSR", "ODU", "PHOTONIC_MEDIA"]	RO	M	• Provided by tapi-server
link	List of {**link**}	RO	M	• Provided by ***tapi-server***
node	List of {**node**}	RO	M	• Provided by ***tapi-server***

Table 9: Node object definition

Node	**/tapi-common:context/tapi-topology:topology-context/topology/node**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
uuid	"[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}"	RO	M	• As per RFC 4122 • Provided by tapi-server
name	List of {value-name: value} • "value-name": "NW-NE-NAME" "value": " [0-9a-zA-Z_]{64}"	RO	M	• Provided by tapi-server • NW-NE-NAME format is described in section 4.2
layer-protocol-name	List of ["DSR", "ODU", "PHOTONIC_MEDIA"]	RO	M	• Provided by tapi-server
administrative-state	["UNLOCKED", "LOCKED"]	RO	M	• Provided by tapi-server
operational-state	["ENABLED", "DISABLED"]	RO	M	• Provided by tapi-server
lifecycle-state	["PLANNED", "POTENTIAL_AVAILABLE", "POTENTIAL_BUSY", "INSTALLED", "PENDING_REMOVAL"]	RO	M	• Provided by tapi-server
total-potential-capacity	“total-size”: {value: unit} • "value": "[0-9]{8}", • "unit": ["TB", "TBPS", "GB", "GBPS", "MB", "MBPS", "KB", "KBPS"]	RO	O	• Provided by tapi-server
available-capacity	“total-size”: {value: unit} • "value": "[0-9]{8}", • "unit": ["TB", "TBPS", "GB", "GBPS", "MB", "MBPS", "KB", "KBPS"]	RO	O	• Provided by tapi-server
cost-characteristic	List of {cost-name: cost-value} • "cost-name": "HOP_COUNT"	RO	O	• Provided by tapi-server

	"cost-value": "[0-9]{8}"			
latency-characteristic	List of { traffic-property-name: fixed-latency-characteristic } • "traffic-property-name": "FIXED_LATENCY" "fixed-latency-characteristic": "[0-9]{8}"	RO	O	• Provided by tapi-server
encapsulated-topology	{"***topology-ref***"}	RO	O	• Provided by tapi-server • Needed if encapsulated-topology is supported
aggregated-node-edge-point	List of {"***node-edge-point-ref***"}	RO	O	• Provided by tapi-server • Needed if encapsulated-topology is supported
owned-node-edge-point	List of {***node-edge-point***}	RO	M	• Provided by tapi-server • See Table 10
node-rule-group	List of {***node-rule-group***}	RO	O	• Provided by tapi-server • See Table 11

Table 10: Node-edge-point object definition

Node-edge-point	**/tapi-common:context/tapi-topology:topology-context/topology/node/owned-node-edge-point**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
uuid	"[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}"	RO	M	• As per RFC 4122 • Provided by *tapi-server*
name	List of {value-name: value} • "value-name": "INVENTORY_ID", "value": " [0-9a-zA-Z_]{64}"	RO	M	• Provided by *tapi-server* • INVENTORY_ID format is described in section 4.2
layer-protocol-name	["DSR", "ODU", "PHOTONIC_MEDIA"]	RO	M	• Provided by tapi-server
supported-cep-layer-protocol-qualifier	List of ["DIGITAL_SIGNAL_TYPE", "ODU_TYPE", "PHOTONIC_LAYER_QUALIFIER ”]	RO	M	• Provided by tapi-server • All children identities defined for ["DIGITAL_SIGNAL_TYPE", "ODU_TYPE", "PHOTONIC_LAYER_QUALIFIER ”] MUST be supported when applicable.
administrative-state	["UNLOCKED", "LOCKED"]	RO	M	• Provided by *tapi-server*
operational-state	["ENABLED", "DISABLED"]	RO	M	• Provided by *tapi-server*
lifecycle-state	["PLANNED", "POTENTIAL_AVAILABLE", "POTENTIAL_BUSY", "INSTALLED", "PENDING_REMOVAL"]	RO	M	• Provided by *tapi-server*
termination-state	["LP_CAN_NEVER_TERMINATE" , “LT_NOT_TERMINATED”, “TERMINATED_SERVER_TO_CL	RO	O	• Provided by *tapi-server*

	IENT_FLOW", "TERMINATED_CLIENT_TO_SERVER_FLOW", "TERMINATED_BIDIRECTIONAL", "LT_PERMENANTLY_TERMINATED", "TERMINATION_STATE_UNKNOWN"]			
termination-direction	["BIDIRECTIONAL", "SINK", "SOURCE"]	RO	M	• Provided by *tapi-server*
link-port-direction	["BIDIRECTIONAL", "INPUT","OUTPUT"]	RO	M	• Provided by *tapi-server*
link-port-role	["SYMMETRIC"]	RO	O	• Provided by *tapi-server*
total-potential-capacity	"total-size": {value: unit} • "value": "[0-9]{8}", • "unit": ["TB", "TBPS", "GB", "GBPS", "MB", "MBPS", "KB", "KBPS"]	RO	M	• Provided by *tapi-server*
available-capacity	"total-size": {value: unit} • "value": "[0-9]{8}", • "unit": ["TB", "TBPS", "GB", "GBPS", "MB", "MBPS", "KB", "KBPS"]	RO	M	• Provided by *tapi-server*
aggregated-node-edge-point	List of {***node-edge-point-ref}***	RO	O	• Provided by *tapi-server*
mapped-service-interface-point	List of {***"/tapi-common:context/service-interface-point/uuid***"}	RO	O	• Provided by *tapi-server*
cep-list	List of {***connection-end-point***}	RO	M	• Provided by *tapi-server*
tapi-photonic-media: media-channel-node-edge-point-spec	"mc-pool": {supportable/available/occupied-spectrum} • List of "supportable/available/occupied-spectrum":{upper/lower-frequency, frequency-constraint: {adjustment-granularity, grid-type} } ○ "upper/lower-frequency": " [0-9]{9}", ○ "adjustment-granularity":["UNCONSTRAINED", "G_3_125GHZ", "G_6_25GHZ", "G_12_5GHZ", "G_25GHZ", "G_50GHZ", "G_100GHZ",] ○ "grid-type": ["GRIDLESS", "FLEX", "CWDM", "DWDM"]	RO	O	• Provided by *tapi-server* • This block of attributes MUST augment PHOTONIC_MEDIA NEPs exposing Media_Channel characteristics. • The **upper/lower-frequency** bound of the media channel spectrum specified in MHz.. • **Adjustment-granularity** in Gigahertz. As per ITU-T G.694.1, it is used to calculate nominal central frequency". • The **grid-type** specifies the reference set of frequencies used to denote allowed nominal central frequencies that may be used for defining applications.

tapi-odu:odu-node-edge-point-spec	"odu-pool": {client capacity, max-client-instances, max-client-size}	RO	O	• Provided by *tapi-server*

Table 11: Node-rule-group object definition

Node-rule-group	/tapi-common:context/tapi-topology:topology-context/topology/node/node-rule-group			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
uuid	"[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}"	RO	M	• As per RFC 4122 • Provided by *tapi-server*
name	List of {value-name: value} • "value-name": "NRG_NAME" "value": " [0-9a-zA-Z_]{64}"	RO	M	• Provided by *tapi-server*
node-edge-point	List of {***node-edge-point-ref}***	RO	M	• Provided by *tapi-server*
rule	List of {**rule**}	RO	M	• Provided by *tapi-server* • See Table 12

Table 12: Rule object definition

Rule	/tapi-common:context/tapi-topology:topology-context/topology/node/node-rule-group/rule			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
local-id	"[0-9a-zA-Z_]{32}"	RO	M	• Provided by *tapi-server*
name	List of {value-name: value} • "value-name": "RULE_NAME" "value": " [0-9a-zA-Z_]{64}"	RO	M	• Provided by *tapi-server*
rule-type	["FORWARDING"]	RO	M	• Provided by *tapi-server* • Support other rule types?
forwarding-rule	["MAY_FORWARD_ACROSS_GROUP", "MUST_FORWARD_ACROSS_GROUP", "CANNOT_FORWARD_ACROSS_GROUP", "NO_STATEMENT_ON_FORWARDING"]	RO	M	• Provided by *tapi-server*

Table 13: Link object definition

Link	/tapi-common:context/tapi-topology:topology-context/topology/link			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**

uuid	"[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}"	RO	M	• As per RFC 4122 • Provided by *tapi-server*
name	List of {value-name: value} • "value-name": "LINK_NAME" "value": " [0-9a-zA-Z_]{64}"	RO	M	• Provided by *tapi-server*
layer-protocol-name	List of ["DSR", "ODU", "PHOTONIC_MEDIA"]	RO	M	
administrative-state	["UNLOCKED", "LOCKED"]	RO	M	• Provided by *tapi-server*
operational-state	["ENABLED", "DISABLED"]	RO	M	• Provided by *tapi-server*
lifecycle-state	["PLANNED", "POTENTIAL_AVAILABLE", "POTENTIAL_BUSY", "INSTALLED", "PENDING_REMOVAL"]	RO	M	• Provided by *tapi-server*
direction	["BIDIRECTIONAL"]	RO	M	• Provided by *tapi- server*
total-potential-capacity	"total-size": {value: unit} • "value": "[0-9]{8}", • "unit": ["TB", "TBPS", "GB", "GBPS", "MB", "MBPS", "KB", "KBPS"]	RO	M	• Provided by *tapi-server*
available-capacity	"total-size": {value: unit} • "value": "[0-9]{8}", • "unit": ["TB", "TBPS", "GB", "GBPS", "MB", "MBPS", "KB", "KBPS"]	RO	M	• Provided by *tapi-server*
transitioned-layer-protocol-name	List of {**layer-protocol-name**}	RO	M	• Provided by *tapi-server* • Only applicable to transitional-links.
cost-characteristic	List of {cost-name: cost-value} • "cost-name": "HOP_COUNT" "cost-value": "[0-9]{8}"	RO	O	
latency-characteristic	List of { traffic-property-name: fixed-latency-characteristic } • "traffic-property-name": "FIXED_LATENCY" "fixed-latency-characteristic": "[0-9]{8}"	RO	O	
Risk-characteristic	List of {risk-characteristic-name: risk-identifier-list} • "risk-characteristic-name": ["SRLG", "SRNG"] "risk-identifier-list": List of "[0-9a-zA-Z_]{64}"	RO	O	
node-edge-point	List of {***"node-edge-point-ref"***}	RO	M	• Provided by *tapi-server*

6.1.2.2 Expected results

In this section we introduce the detail TAPI-Topology modelling expected at "Day 0" (i.e., after the commissioning stage of the network devices into the SDN-C, but before any service is configured).

6.1.2.2.1 UC0b - Example 0: TAPI topology representation at "day 0" following Transitional Link modelling approach

The detailed description of the assumptions assumed by the Reference Implementation to compose the Topology at Day '0' based on a Transitional-Link model, are described in [TAPI-TOP-MODEL-REQ-9].

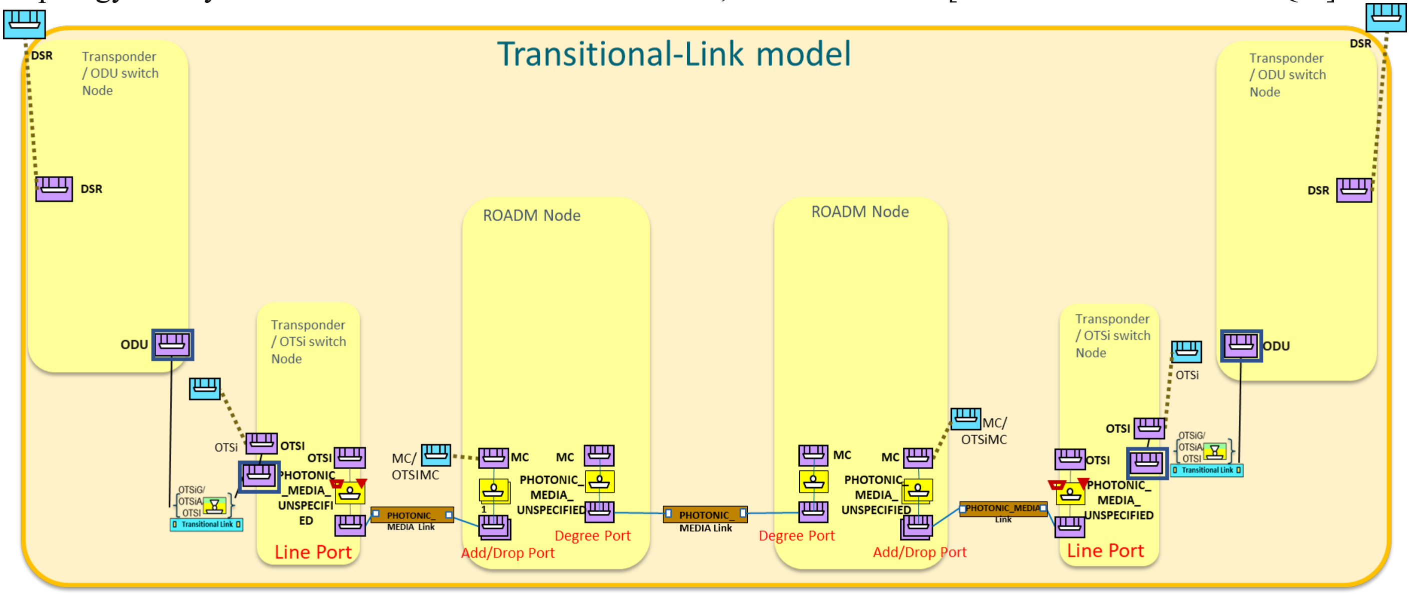


Figure 6-3 TAPI topology representation at "day 0" following Transitional Link modelling approach.

6.1.2.2.2 UC0b - Example 0: TAPI topology representation at "day 0" following Multi-layer Node modelling approach

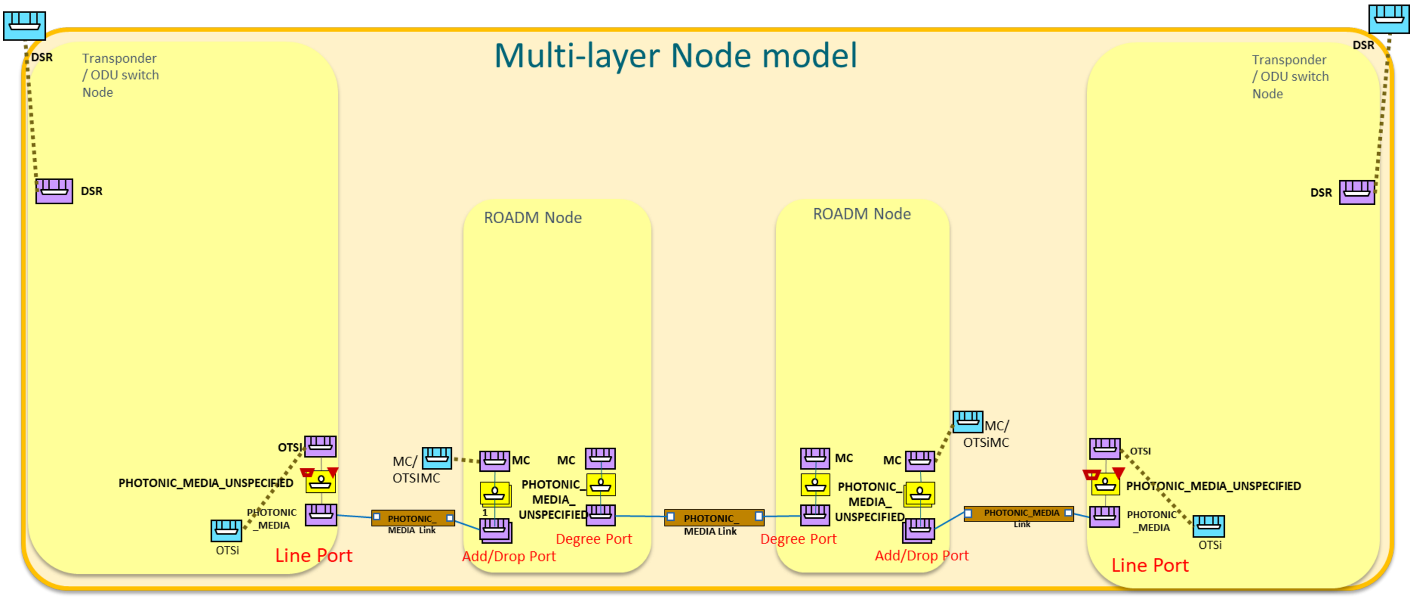


Figure 6-4 TAPI topology representation at "day 0" following Multi-layer Node modelling approach.

6.1.3 Use case 0c: Connectivity Service discovery (synchronous mode)

Number	**UC 0C**
Name	Connectivity Service discovery (synchronous mode)
Technologies involved	Optical
Process/Areas Involved	Planning and Operations
Brief description	The TAPI Connectivity Service is a relevant network service information required for the operation. The discovery of this information is intended to be requested periodically and/or on-demand basis, proactively from the TAPI client role, in order to synchronize the connectivity information.
Layers involved	DSR, ODU, PHOTONIC_MEDIA
Type	Planning
Description & Workflow	The Use Case 0c: Connectivity Service discovery consists on the retrieve of a connectivity-service at the DSR/ODU/PHOTONIC_MEDIA layers and the retrieval of the generated connections information. The complete workflow is shown in Figure 6-5. The NBI Client first retrieves the connectivity-context trimmed by the *?fields=connectivity-service* filter to retrieve all connectivity-services deployed in the TAPI Server (2). Then, iteratively the information of each Connectivity-Service (3) is requested, and also its list of Connection references (5). For all Connection reference a Connection retrieval operation is performed to get the Connection object details (7). The NBI server MUST return a valid object, if previous operations (4)(6)(8) succeed, which are compliant with the definition of the objects included in Table 14 (Connectivity-Service) and Table 16 (Connection). Please note the Connection object MUST include all attributes marked as Mandatory (M) in the Support (Sup) in all Connection sub-objects defined in Table 17, Table 18, Table 19, Table 20 and Table 22.

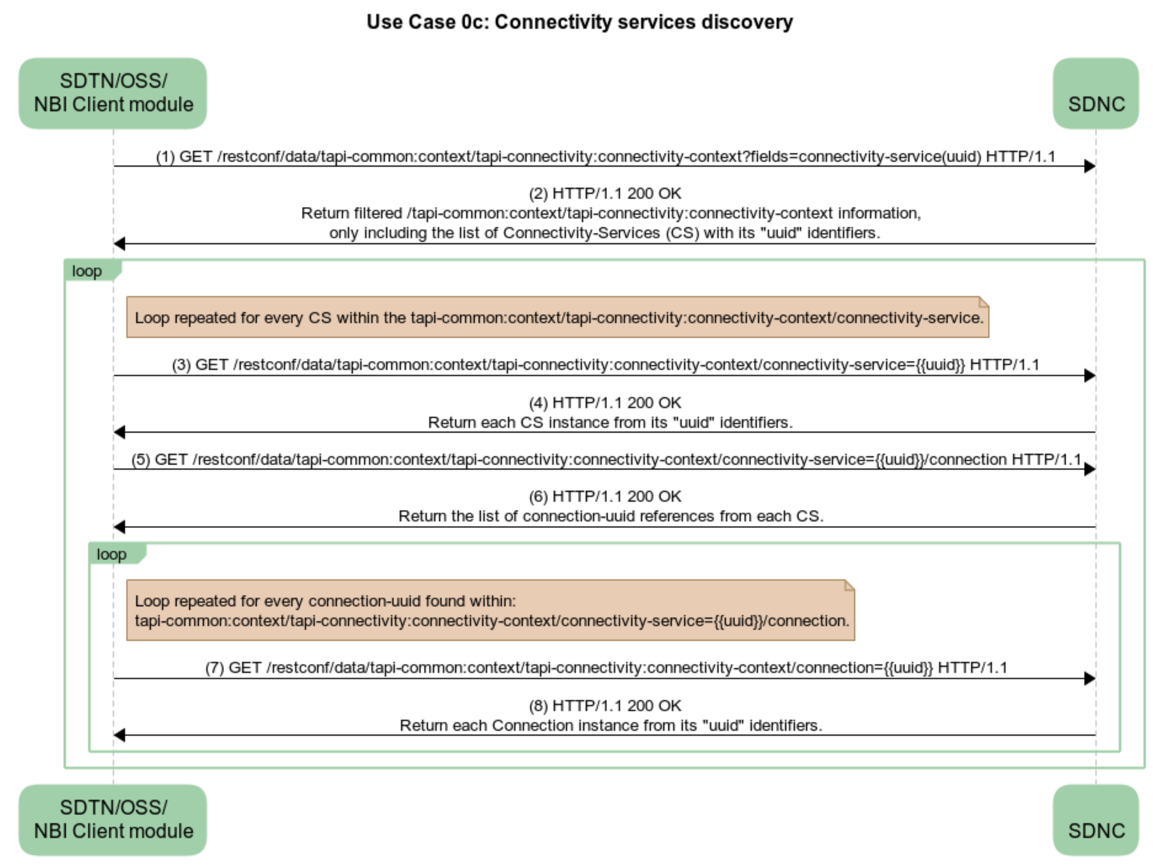


Figure 6-5 UC-0c: Connectivity Service - LLD Workflow.

6.1.3.1 Required parameters

For the details about the required parameters for each object retrieved in the previously described RESTCONF call operation, please refer to the UC1a Required Parameters Section 6.2.1.1

6.2 Unconstrained E2E Service Provisioning

Please note that this use case can divided in five sub use cases depending on the layer on which it is applied:

- **UC1a: Unconstrained DSR Service Provisioning single wavelength (≤100G).**
- **UC1b: Unconstrained DSR Service Provisioning multi wavelength (beyond 100G).**
- **UC1c: Unconstrained ODU Service Provisioning**
- **UC1d: Unconstrained PHOTONIC_MEDIA/OTSi Service Provisioning**
- **UC1e: Unconstrained PHOTONIC_MEDIA/OTSiA Service Provisioning**
- **UC1f: Unconstrained PHOTONIC_MEDIA/OTSiMC Service Provisioning**

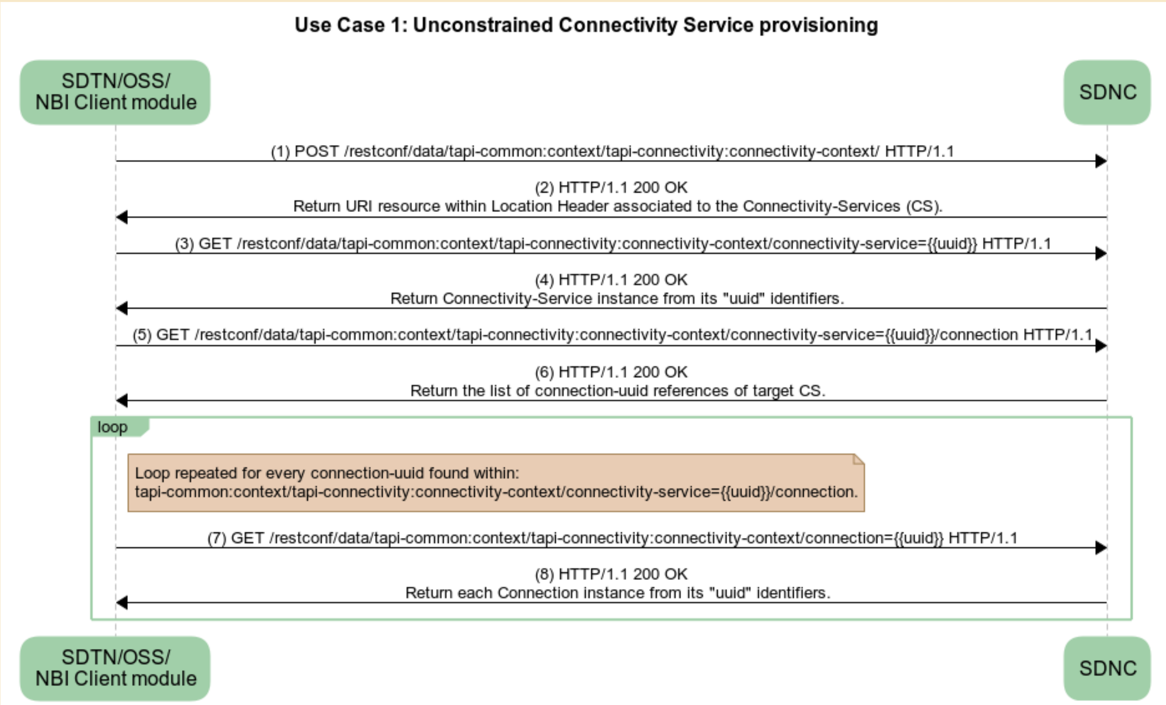
UC1c is left to be considered in future version of this specification.

The workflow definition and required parameters described in these sections applies to all the different sub-use cases but the results in terms of the number of connections generated are different depending on the layer to which the unconstrained connectivity-service has been requested.

The main different between the different sub-use cases described here, is the selection of the Connectivity-Service-End-Points, i.e., the layer of their associated SIPs. Depending on this selection the connections generated in the network are up to a networking layer or another. E.g., for the provisioning of a **PHOTONIC_MEDIA/OTSi Connectivity-Service,** the TAPI Server will only generates connection up to the PHOTONIC_MEDIA/OTSi network layer.

6.2.1 Use case 1a: Unconstrained DSR Service Provisioning single wavelength (<100G).

Number	**UC1a**
Name	**Unconstrained DSR Service Provisioning single wavelength (<100G).**
Technologies involved	Optical
Process/Areas Involved	Planning and Operations
Brief description	The UC1 describes the provisioning of a ***tapi-connectivity:connectivity-service*** instance between service-interface-points exposed by the TAPI-Server at the DSR networking layer. **This service can include intermediate regeneration if necessary.** The underlying connection provisioning and management (including lower layer connections e.g., ODU, OTSi, MC, OTSiMC including intermediate regeneration connections if needed) is performed by the SDN Domain controller. The path of each lower layer connection (e.g., ODU or OCh/OTSi, OMS) across the network topology is calculated by the controller and the connection automatically provisioned. The “unconstrained” term refers that the TAPI-Client is not introducing any routing constraint in the service request, thus rely completely into the routing capabilities of the TAPI-Server to select the network resources employed to provide the desired service characteristics. Moreover, the TAPI-Client is not providing technology specific Traffic-Engineering constrains such time-slot selection at the ODU layer, or optical-spectrum selection for the routing of OTSi connections.
Layers involved	DSR/ODU/PHOTONIC_MEDIA

Type	Provisioning
Description & Workflow	The Use Case 1: Connectivity Service provisioning consists on the creation of a connectivity-service between Service-Interface-Points at the DSR/ODU/PHOTONIC_MEDIA layers and the retrieval of the generated connections information. The complete workflow is shown in Figure 6-6. The first operation (1) triggers the creation of Connectivity-Service in the NBI Server. The TAPI server MUST accept any valid combination of the attributes marked as Mandatory (M) in the Support (Sup) column of Table 14 and Table 15. If the operation is successful, the NBI server MUST return an http response message with the Location Header as specified in https://www.w3.org/Protocols/rfc2616/rfc2616-sec9.html#sec9.5. Once the resource has been created (here a pooling or event-trigger mechanism need to be defined in order to reconciliate the information),) the NBI Client may retrieve all information of the Connectivity-Service (3), Connection list of references (5) and all its Connection objects (7). The NBI server MUST return a valid object, if previous operations (4)(6)(8) succeed, which are compliant with the definition of the objects included in Table 14 (Connectivity-Service) and Table 16 (Connection). Please note the Connection object MUST include all attributes marked as Mandatory (M) in the Support (Sup) in all Connection sub-objects defined in Table 17, Table 18, Table 19, **Error! Reference source not found.**Table 20 and Table 22.  **Figure 6-6 UC-1: Unconstrained end-to-end service provisioning.**

6.2.1.1 Required parameters

Table 14: Connectivity-service object definition

Connectivity-service	**/tapi-common:context/tapi-connectivity:connectivity-context/connectivity-service**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
uuid	"[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}"	RW	M	• As per RFC 4122 • Provided by ***tapi-client***
name	List of {value-name: value} • "value-name": "SERVICE_NAME" "value": " [0-9a-zA-Z_]{64}"	RW	M	• Provided by *tapi-client*
administrative-state	["UNLOCKED", "LOCKED"]	RW	O	• Provided by *tapi-client*
operational-state	["ENABLED", "DISABLED"]	RO	M	• Provided by *tapi-server*
lifecycle-state	["PLANNED", "POTENTIAL_AVAILABLE", "POTENTIAL_BUSY", "INSTALLED", "PENDING_REMOVAL"]	RO	M	• Provided by *tapi-server*
requested-capacity	"total-size": {value: ,unit:} • "value": "[0-9]{8}", • "unit": ["TB", "TBPS", "GB", "GBPS", "MB", "MBPS", "KB", "KBPS"]	RW	M	• Provided by *tapi-client*
service-type	["POINT_TO_POINT_CONNECTIVITY"]	RW	O	• Provided by *tapi-client* • Support P2P only
service-layer	List of ["DSR", "ODU", "PHOTONIC_MEDIA"]	RW	M	• Provided by *tapi-client*
preferred-transport-layer	List of ["DSR", "ODU", "PHOTONIC_MEDIA"]	RW	O	• Provided by *tapi-client*
connection	List of {**connection-ref** - */tapi-common:context/tapi-connectivity:connectivity-context/connection/uuid*}	RO	M	• Provided by *tapi-server*
end-point	List of {**connectivity-service-end-point**}	RW	M	• Provided by *tapi-client* • Min elements 2. • See Table 15

Table 15: Connectivity-service-end-point object definition

Connectivity-service-end-point	**/tapi-common:context/tapi-connectivity:connectivity-service/end-point**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
local-id	"[0-9a-zA-Z_]{32}"	RW	M	• Provided by *tapi-client*
layer-protocol-name	List of ["DSR", "ODU", "PHOTONIC_MEDIA"]	RW	M	• Provided by *tapi-client*

supported-layer-protocol-qualifier	List of ["DIGITAL_SIGNAL_TYPE", "ODU_TYPE", "PHOTONIC_LAYER_QUALIFIER"]	RW	M	• Provided by tapi-server • All children identities defined for ["DIGITAL_SIGNAL_TYPE", "ODU_TYPE", "PHOTONIC_LAYER_QUALIFIER"] MUST be supported.
administrative-state	["UNLOCKED", "LOCKED"]	RW	O	• Provided by *tapi-client*
operational-state	["ENABLED", "DISABLED"]	RO	O	• Provided by *tapi-server*
lifecycle-state	["PLANNED", "POTENTIAL_AVAILABLE", "POTENTIAL_BUSY", "INSTALLED", "PENDING_REMOVAL"]	RO	O	• Provided by *tapi-server*
direction	["BIDIRECTIONAL", "INPUT", "OUTPUT"]	RW	O	• Provided by *tapi-client*
role	["SYMMETRIC", "TRUNK"]	RW	O	• Provided by *tapi-client* • Support only P2P
capacity	"total-size": {value: unit} • "value": "[0-9]{8}", • "unit": ["TB", "TBPS", "GB", "GBPS", "MB", "MBPS", "KB", "KBPS"]	RW	O	• Provided by *tapi-client*
service-interface-point	*"/tapi-common:context/service-interface-point/uuid"*	RW	M	• Provided by *tapi-client*
connection-end-point	List { ***connection-end-point}***	RO	M	• Provided by *tapi-server* • See Table 17

Table 16: Connection object definition

connection	**/tapi-common:context/tapi-connectivity:connection**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
uuid	"[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}"	RO	M	• As per RFC 4122 • Provided by *tapi-server*
name	List of {value-name, value} • "value-name": "CONNECTION_NAME" "value": " [0-9a-zA-Z_]{64}"	RO	M	• Provided by *tapi-server*
layer-protocol-name	List of ["DSR", "ODU", "PHOTONIC_MEDIA"]	RO	M	• Provided by *tapi-client*
operational-state	["ENABLED", "DISABLED"]	RO	M	• Provided by *tapi-server*
lifecycle-state	["PLANNED", "POTENTIAL_AVAILABLE", "POTENTIAL_BUSY", "INSTALLED", "PENDING_REMOVAL"]	RO	M	• Provided by *tapi-server*
direction	["BIDIRECTIONAL"]	RO	M	• Provided by *tapi- server* • Support UNIDIRECTIONAL?

lower-connection	List of {**connectivity-ref** - */tapi-common:context/tapi-connectivity:connectivity-context/connection/uuid}*	RO	M	• Provided by *tapi-server*
connection-end-point	List of {"connection-end-point-ref - */tapi-common:context/tapi-topology:topology-context/topology/node/owned-node-edge-point/tapi-connectivity:cep-list/connection-end-point/uuid* "}	RO	M	• Provided by *tapi-server*
route	List of { **route** }	RO	M	• Provided by *tapi-server* • See Table 22
switch-control	List of { **switch-control** }	RO	M	• Provided by *tapi-server*
supported-client-link	List of {link-ref - topology-uuid + link-uuid	RO	O	• Provided by *tapi-server*

Table 17: Connection-end-point object definition

Connection-end-point	**/tapi-common:context/tapi-topology:topology-context/topology/node/owned-node-edge-point/tapi-connectivity:cep-list/connection-end-point**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
uuid	"[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}"	RO	M	• As per RFC 4122 • Provided by *tapi-server*
name	List of {value-name: value} • "value-name": "CEP_NAME" "value": " [0-9a-zA-Z_]{64}"	RO	M	• Provided by *tapi-server*
layer-protocol-name	List of ["DSR", "ODU", "PHOTONIC_MEDIA"]	RO	M	Provided by *tapi-server*
supported-layer-protocol-qualifier	List of ["DIGITAL_SIGNAL_TYPE", "ODU_TYPE", "PHOTONIC_LAYER_QUALIFIER"]	RO	M	• Provided by tapi-server All children identities defined for ["DIGITAL_SIGNAL_TYPE", "ODU_TYPE", "PHOTONIC_LAYER_QUALIFIER"] MUST be supported.
administrative-state	["UNLOCKED", "LOCKED"]	RO	M	• Provided by *tapi-server*
operational-state	["ENABLED", "DISABLED"]	RO	M	• Provided by *tapi-server*
lifecycle-state	["PLANNED", "POTENTIAL_AVAILABLE", "POTENTIAL_BUSY", "INSTALLED", "PENDING_REMOVAL"]	RO	M	• Provided by *tapi-server*
termination-state	["LP_CAN_NEVER_TERMINATE", "LT_NOT_TERMINATED", "TERMINATED_SERVER_TO_CLIENT_FLOW", "TERMINATED_CLIENT_TO_SERVER_FLOW", "TERMINATED_BIDIRECTIONAL", "LT_PERMENANTLY_TERMINATED	RO	M	• Provided by *tapi-server*

	", "TERMINATION_STATE_UNKNOWN "]			
termination-direction	["BIDIRECTIONAL", "SINK", "SOURCE"]	RO	M	• Provided by *tapi-server*
connection-port-direction	["BIDIRECTIONAL","INPUT","OUTPUT"]	RO	M	• Provided by *tapi-server*
connection-port-role	["SYMMETRIC"]	RO	M	• Provided by *tapi-server*
aggregated-connection-end-point	List of {***node-edge-point-ref***}	RO	O	• Provided by *tapi-server*
parent-node-edge-point	List of {***node-edge-point-ref***}	RO	M	• Provided by *tapi-server*
client-node-edge-point	List of {***node-edge-point-ref***}	RO	M	• Provided by *tapi-server*
tapi-odu:odu-connection-end-point-spec	**{odu-connection-end-point-spec}**	RO	O	• Provided by tapi-server • MUST augment CEPs at the ODU layer • See Table 18
tapi-photonic-media:otsi-connection-end-point-spec	**{ otsi-connection-end-point-spec }**	RO	O	• Provided by tapi-server • MUST augment CEPs at the PHOTONIC_MEDIA layer with OTSI layer-protocol-qualifier. • See Table 19
tapi-photonic-media:media-channel-connection-end-point-spec	**{media-channel-connection-end-point-spec}**	RO	O	• Provided by tapi-server • MUST augment CEPs at the PHOTONIC_MEDIA layer with MC/OTSiMC layer-protocol-qualifier. • See Table 20
tapi-photonic-media:ots-media-channel-connection-end-point-spec	**{ots-media-channel-connection-end-point-spec}**			• Provided by tapi-server • MUST augment CEPs at the PHOTONIC_MEDIA layer with layer-protocol-qualifier:LAYER_PROTOCOL_QUALIFIER_UNSPECIFIED • See Table 21

Table 18: ODU-Connection-end-point-spec object definition

odu-connection-end-point-spec	**/tapi-common:context/tapi-topology:topology-context/topology/node/owned-node-edge-point/tapi-connectivity:cep-list/connection-end-point/tapi-odu:odu-connection-end-point-spec**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
odu-common	{ odu-type, odu-rate, odu-rate-tolerance } • "odu-type": [ODU_TYPE], • "odu-rate": [0-9]{12}, • "odu-rate-tolerance": [0-9]{12},	RO	M	• Provided by *tapi-server* • All children identities defined for ["ODU_TYPE"] MUST be supported. • **odu-rate-tolerance** Standardized values are defined in Table 7-2/G.709
odu-term-and-adapter	{ opu-tributary-slot-list, auto-payload-type, configured-client-type, configured-	RO	M	• Provided by *tapi-server*

	mapping-type, accepted-payload-type, named-payload-type, hex-payload-type} • opu-tributary-slot-size: ["1G25", "2G5"] • auto-payload-type? boolean • configured-client-type: [DIGITAL_SIGNAL_TYPE] • configured-mapping-type: ["AMP", "BMP", "GFP-F", "GMP", "TTP_GFP_BMP", "NULL"] • accepted-payload-type ○ "named-payload-type": ["UNKNOWN", "UNINTERPRETABLE"] ○ "hex-payload-type": "[0-9]{64}", • fec-parameters : { pre-fec-ber, post-fec-ber, corrected-bytes, corrected-bits, uncorrectable-bytes, uncorrectable-bits} ○ "pre-fec-ber": "[0-9]{64}", ○ "post-fec-ber": "[0-9]{64}", ○ "corrected-bytes": "[0-9]{64}", ○ "corrected-bits": "[0-9]{64}", ○ "uncorrectable-bytes": "[0-9]{64}", ○ "uncorrectable-bits": "[0-9]{64}", • odu-cn-effective-time-slot: List of "[0-9]{64}"			• *Configured-client-type* accepts any child identities defined for ["DIGITAL_SIGNAL_TYPE"]
odu-ctp	{tributary-slot-list, tributary-port-number, accepted-msi} • tributary-slot-list : List of "[0-9]{64}" • tributary-port-number: "[0-9]{64}" • accepted-msi? string	RO	M	• Provided by *tapi-server*
odu-protection	{aps-enable, aps-level} • aps-enable : Boolean • aps-level: "[0-9]{64}"	RO	O	• Provided by tapi-server

Table 19: otsi-connection-end-point-spec object definition

otsi-connection-end-point-spec	**/tapi-common:context/tapi-topology:topology-context/topology/node/owned-node-edge-point/tapi-connectivity:cep-list/connection-end-point/tapi-photonic-media:otsi-connection-end-point-spec**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
otsi-termination	{selected-central-frequency, selected-application-identifier, selected-modulation, selected-spectrum,	RO	M	• Provided by *tapi-server*

	transmitted-power, received-power, laser-properties}			
selected-central-frequency	{ central-frequency, frequency-constraint: {adjustment-granularity, grid-type} } • "central-frequency": "[0-9]{9}", • "adjustment-granularity":["UNCONSTRAINED", "G_3_125GHZ", "G_6_25GHZ", "G_12_5GHZ", "G_25GHZ", "G_50GHZ", "G_100GHZ",] • "grid-type": ["GRIDLESS", "FLEX", "CWDM", "DWDM"]	RO	M	• Provided by *tapi-server* • The **central-frequency** of the laser specified in MHz. It is the oscillation frequency of the corresponding electromagnetic wave. • **Adjustment-granularity** in Gigahertz. As per ITU-T G.694.1, it is used to calculate nominal central frequency". • The **grid-type** specifies the reference set of frequencies used to denote allowed nominal central frequencies that may be used for defining applications.
selected-application-identifier	{ application-identifier-type, application-code} • "application-identifier-type":["PROPRIETARY", "ITUT_G959_1", "ITUT_G698_1", "ITUT_G698_2", "ITUT_G696_1", "ITUT_G695",] • "application-code":"[0-9a-zA-Z_]{64}"	RO	M	• Provided by tapi-server
selected -modulation	["RZ", "NRZ", "BPSK", "DPSK", "QPSK", "8QAM", "16QAM"]	RO	M	• Provided by *tapi-server*
selected-spectrum	{lower-frequency, upper-frequency, frequency-constraint: {adjustment-granularity, grid-type} } • "upper/lower-frequency": "[0-9]{9}", • "adjustment-granularity":["UNCONSTRAINED", "G_3_125GHZ", "G_6_25GHZ", "G_12_5GHZ", "G_25GHZ", "G_50GHZ", "G_100GHZ",] • "grid-type": ["GRIDLESS", "FLEX", "CWDM", "DWDM"]	RO	M	• Provided by *tapi-server* • The **central-frequency** of the laser specified in MHz. It is the oscillation frequency of the corresponding electromagnetic wave. • **Adjustment-granularity** in Gigahertz. As per ITU-T G.694.1, it is used to calculate nominal central frequency". • The **grid-type** specifies the reference set of frequencies used to denote allowed nominal central frequencies that may be used for defining applications.
transmited-power	{total-power, power-spectral-density} • "total-power":"[0-9].[0-9]{7}", • "power-spectral-density": "[0-9].[0-9]{7}",	RO	M	• Provided by *tapi-server*
received-power	{total-power, power-spectral-density} • "total-power":"[0-9].[0-9]{7}", • "power-spectral-density": "[0-9].[0-9]{7}",	RO	M	• Provided by *tapi-server*

laser-properties	{laser-status, laser-application-type, laser-bias-current, laser-temperature} • "laser-status": ["ON","OFF","PULSING", "UNDEFINED"] • "laser-application-type": ["PUMP", "MODULATED", "PULSE"] • "laser-bias-current": "[0-9].[0-9]{7}", • "laser-temperature": "[0-9].[0-9]{7}",	RO	M	• Provided by *tapi-server*

Table 20 media-channel-connection-end-point-spec object definition

media-channel-connection-end-point-spec	**/tapi-common:context/tapi-topology:topology-context/topology/node/owned-node-edge-point/tapi-connectivity:cep-list/connection-end-point/tapi-photonic-media:media-channel-connection-end-point-spec**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
media-channel	{occupied-spectrum, measured-power-ingress, measured-power-egress}	R0	M	**Mandatory only for PHOTONIC_LAYER_QUALIFIER_MC layer-protocol-qualified CEPs**
occupied-spectrum	{lower-frequency, upper-frequency, frequency-constraint} • "upper/lower-frequency": "[0-9]{9}", • "frequency-constraint": {adjustment-granularity, grid-type} o "adjustment-granularity": ["UNCONSTRAINED", "G_3_125GHZ", "G_6_25GHZ", "G_12_5GHZ", "G_25GHZ", "G_50GHZ", "G_100GHZ",] o "grid-type": ["GRIDLESS", "FLEX", "CWDM", "DWDM"]	RO	M	• Provided by *tapi-client* • The **upper/lower-frequency** boundaries of the band specified in MHz. • **Adjustment-granularity** in Gigahertz. As per ITU-T G.694.1, it is used to calculate nominal central frequency". • The **grid-type** specifies the reference set of frequencies used to denote allowed nominal central frequencies
measured-power-ingress	{total-power, power-spectral-density} • "total-power":"[0-9].[0-9]{7}", • "power-spectral-density": "[0-9].[0-9]{7}",	RO	M	• Provided by *tapi-server*
measured-power-egress	{total-power, power-spectral-density} • "total-power":"[0-9].[0-9]{7}", • "power-spectral-density": "[0-9].[0-9]{7}",	RO	M	• Provided by *tapi-server*

Table 21 ots-media-channel-connection-end-point-spec object definition

ots-media-channel-connection-end-point-spec	**/tapi-common:context/tapi-topology:topology-context/topology/node/owned-node-edge-point/tapi-connectivity:cep-list/connection-end-point/tapi-photonic-media:ots-media-channel-connection-end-point-spec**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
ots-media-channel	{occupied-spectrum, measured-power-ingress, measured-power-egress}	R0	M	**Mandatory only for LAYER_PROTOCOL_QUALIFIER_ UNSPECIFIED layer-protocol-qualified CEPs corresponding to OMS-OTS abstraction in the photonic_media layer**
occupied-spectrum	{lower-frequency, upper-frequency, frequency-constraint} • "upper/lower-frequency": "[0-9]{9}", • "frequency-constraint": {adjustment-granularity, grid-type} o "adjustment-granularity": ["UNCONSTRAINED", "G_3_125GHZ", "G_6_25GHZ", "G_12_5GHZ", "G_25GHZ", "G_50GHZ", "G_100GHZ",] o "grid-type": ["GRIDLESS", "FLEX", "CWDM", "DWDM"]	RO	M	• Provided by *tapi-client* • The **upper/lower-frequency** boundaries of the band specified in MHz. • **Adjustment-granularity** in Gigahertz. As per ITU-T G.694.1, it is used to calculate nominal central frequency". • The **grid-type** specifies the reference set of frequencies used to denote allowed nominal central frequencies
measured-power-ingress	{total-power, power-spectral-density} • "total-power":"[0-9].[0-9]{7}", • "power-spectral-density": "[0-9].[0-9]{7}",	RO	M	• Provided by *tapi-server*
measured-power-egress	{total-power, power-spectral-density} • "total-power":"[0-9].[0-9]{7}", • "power-spectral-density": "[0-9].[0-9]{7}",	RO	M	• Provided by *tapi-server*

Table 22: Route object definition

route	**/tapi-common:context/tapi-connectivity:connection/route**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
local-id	"[0-9a-zA-Z_]{32}"	RO	M	• Provided by *tapi-server*
name	List of {value-name: value} • "value-name": "ROUTE_NAME" "value": " [0-9a-zA-Z_]{64}"	RO	M	• Provided by *tapi-server*
connection-end-point	List of {"connection-end-point-ref - */tapi-common:context/tapi-topology:topology-context/topology/node/owned-node-edge-*	RO	M	• Provided by *tapi-server*

	point/tapi-connectivity:cep-list/connection-end-point/uuid "}

Table 23: otsia-connectivity-service-end-point-spec object definition

otsia-connectivity-service-end-point-spec	**/tapi-common:context/tapi-connectivity:connectivity-service/end-point/otsia-connectivity-service-end-point-spec**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
otsi-config	List of {otsi-config [local-id]} • otsi-config: {central-frequency, spectrum, application-identifier, modulation, laser-control, transmit-power, total-power-warn-threshold-upper, total-power-warn-threshold-lower, local-id, name}	RW	M	• Provided by tapi-client
number-of-otsi	[0-9]{9}	RW	M	• Provided by tapi-client
central-frequency	• central-frequency: "[0-9]{9}", • "frequency-constraint": {adjustment-granularity, grid-type} ○ "adjustment-granularity": ["UNCONSTRAINED", "G_3_125GHZ", "G_6_25GHZ", "G_12_5GHZ", "G_25GHZ", "G_50GHZ", "G_100GHZ",] ○ "grid-type": ["GRIDLESS", "FLEX", "CWDM", "DWDM"]	RW	M	• Provided by *tapi-client* • The **central-frequency** of the laser specified in MHz. It is the oscillation frequency of the corresponding electromagnetic wave. • **Adjustment-granularity** in Gigahertz. As per ITU-T G.694.1, it is used to calculate nominal central frequency". • The **grid-type** specifies the reference set of frequencies used to denote allowed nominal central frequencies
spectrum	{lower-frequency, upper-frequency, frequency-constraint} • "upper/lower-frequency": "[0-9]{9}", • "frequency-constraint": {adjustment-granularity, grid-type} ○ "adjustment-granularity": ["UNCONSTRAINED", "G_3_125GHZ", "G_6_25GHZ", "G_12_5GHZ", "G_25GHZ", "G_50GHZ", "G_100GHZ",] ○ "grid-type": ["GRIDLESS", "FLEX", "CWDM", "DWDM"]	RW	M	• Provided by *tapi-client* • The **upper/lower-frequency** boundaries of the band specified in MHz. • **Adjustment-granularity** in Gigahertz. As per ITU-T G.694.1, it is used to calculate nominal central frequency". • The **grid-type** specifies the reference set of frequencies used to denote allowed nominal central frequencies
application-identifier	{application-identifier-type, application-code} • "application-identifier-type":["PROPRIETARY", "ITUT_G959_1", "ITUT_G698_1", "ITUT_G698_2", "ITUT_G696_1", "ITUT_G695",] "application-code":"[0-9a-zA-Z_]{64}"	RW	M	• Provided by tapi-client

modulation	["RZ", "NRZ", "BPSK", "DPSK", "QPSK", "8QAM", "16QAM"]	RW	M	• Provided by tapi-client
transmit-power	{total-power, power-spectral-density} • "total-power":"[0-9].[0-9]{7}", • "power-spectral-density": "[0-9].[0-9]{7}",	RW	M	• Provided by tapi-client
laser-control	[FORCED-ON, FORCED-OFF, AUTOMATIC-LASER-SHUTDOWN, UNDEFINED]	RW	O	• Provided by tapi-client
total-power-warn-threshold-upper	[0-9].[0-9]{7}	RW	M	• Provided by tapi-client
total-power-warn-threshold-lower	[0-9].[0-9]{7}	RW	M	• Provided by tapi-client
local-id	"[0-9a-zA-Z_]{32}"	RW	M	• Provided by tapi-client
name	List of {value-name: value} • "value-name": "OTSI_CONN_NAME" "value": " [0-9a-zA-Z_]{64}"	RW	M	• Provided by tapi-server

Table 24: mca-connectivity-service-end-point-spec object definition

mca-connectivity-service-end-point-spec	**/tapi-common:context/tapi-connectivity:connectivity-service/end-point/mca-connectivity-service-end-point-spec**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
number-of-mc	[0-9]{9}	RW	M	• Provided by tapi-client
capacity	{value: unit} • "value": "[0-9]{8}", • "unit": ["TB", "TBPS", "GB", "GBPS", "MB", "MBPS", "KB", "KBPS"]	RW	M	• Provided by tapi-client
mc-config	List of {otsi-config [local-id]} • otsi-config:{spectrum, power-management-config-pac }	RW	M	• Provided by tapi-client
spectrum	{lower-frequency, upper-frequency, frequency-constraint} • "upper/lower-frequency": "[0-9]{9}", • "frequency-constraint": {adjustment-granularity, grid-type}	RW	M	• Provided by *tapi-client* • The **upper/lower-frequency** boundaries of the band specified in MHz. • **Adjustment-granularity** in Gigahertz. As per ITU-T G.694.1, it is used to calculate nominal central frequency".

	○ "adjustment-granularity": ["UNCONSTRAINED", "G_3_125GHZ", "G_6_25GHZ", "G_12_5GHZ", "G_25GHZ", "G_50GHZ", "G_100GHZ",] ○ "grid-type": ["GRIDLESS", "FLEX", "CWDM", "DWDM"]			• The **grid-type** specifies the reference set of frequencies used to denote allowed nominal central frequencies
power-management-config-pac	{intended-maximum-output-power, intended-minimum-output-power, expected-maximum-input-power, expected-minimum-input-power} • "intended-maximum-output-power":{total-power, power-spectral-density} ○ "Total-power":"[0-9].[0-9]{64}", ○ "power-spectral-density":"[0-9].[0-9]{64}" • "intended-minimum-output-power":{total-power, power-spectral-density} ○ "Total-power":"[0-9].[0-9]{64}", ○ "power-spectral-density":"[0-9].[0-9]{64}" • "expected-maximum-output-power":{total-power, power-spectral-density} ○ "Total-power":"[0-9].[0-9]{64}", ○ "power-spectral-density":"[0-9].[0-9]{64}" • "expected-minimum-output-power":{total-power, power-spectral-density} ○ "Total-power":"[0-9].[0-9]{64}", ○ "power-spectral-density":"[0-9].[0-9]{64}"	RW	M	• Provided by tapi-client
local-id	"[0-9a-zA-Z_]{32}"	RW	M	Provided by *tapi-client*
name	List of {value-name: value} • "value-name": "OTSI_CONN_NAME" "value": " [0-9a-zA-Z_]{64}"	RW	M	Provided by *tapi-server*

6.2.1.2 Expected results

In the following subsections we include detail examples of the expected results after the successful provisioning of connectivity-services. Please note that all examples follow the rules detailed in section 5.1.1

To show some detailed TAPI connectivity examples, first a simple legend of icons and basic arrangement is included in Figure 6-7.

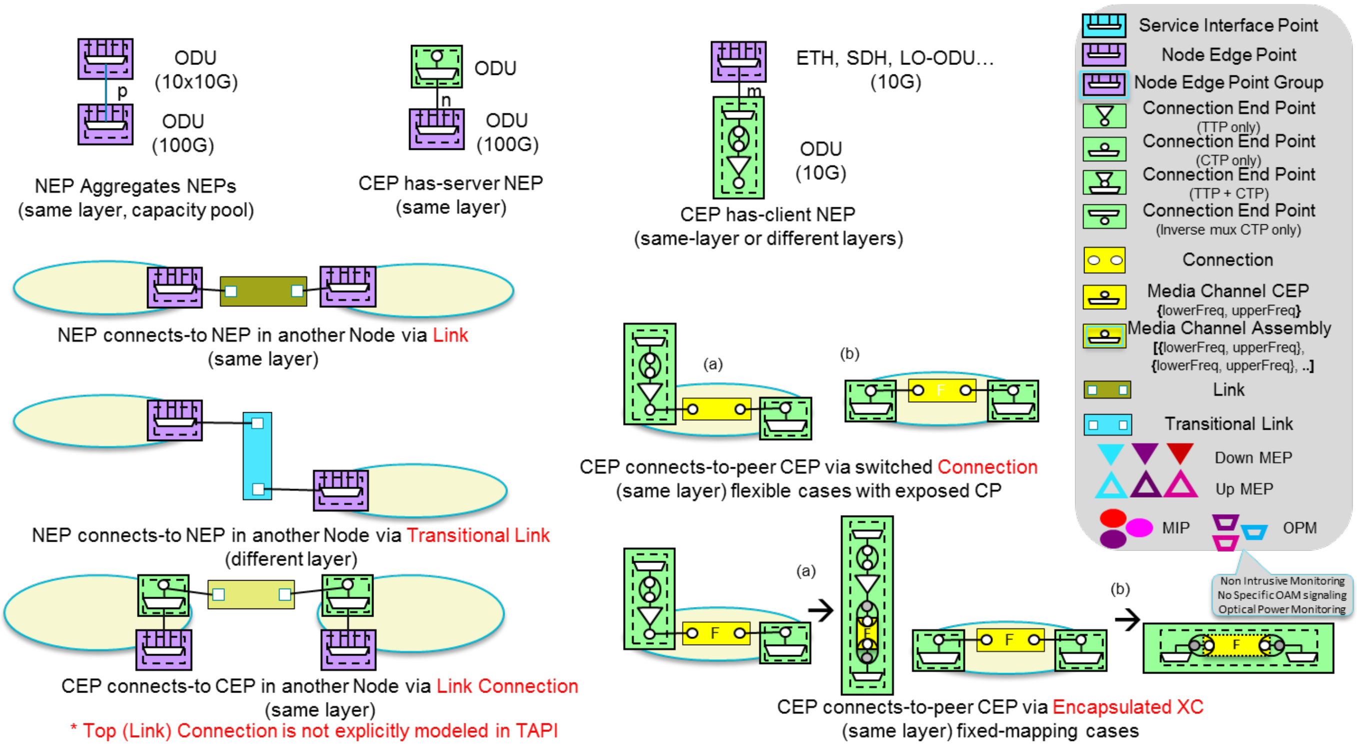


Figure 6-7 TAPI Logical Termination Point Template – Basic Arrangements.

6.2.1.2.1 UC1a - Example 1: 10GE client signal over ODU2 over ODU4 (DSR-ODU Fixed Mapping, flexible ODU allocation)

The following diagrams illustrate a possible sequence of generation of the required TAPI topology and connectivity objects and its relationships according to the rules described in section 5.1.1. However, the internal TAPI server workflow MAY vary and, if notification or streaming services are available, the sequence of asynchronous notifications received by the TAPI client MAY be different.

Thus, the objective of this and the subsequent examples detailing the use cases is to illustrate the object composition and most importantly to define the expected result after a connectivity-service provisioning.

Please note, that in the following examples (all included examples in Section 6), a modelling simplification (represented by blue square) of the client interface (UNI) has been introduced. For the current version of the RIA, the modelling of the UNI client facing side interfaces is not yet covered in detail, so the actual implementation decisions are left to the vendor according to each individual HW capabilities. Thus, please consider the following examples as a guideline of representation the connectivity modelling of the network facing side (e.g., the representation of how the multiplexing of DSR signals over ODU over OTSi shall be modelled). In the following figure we include the general assumed UNI simplifications represented in the examples included in this version of the RIA.

The following figure represents different assumptions done in the UNI representation.

- Option 1: Assumes there is not flexibility at the DSR layer but the fixed cross-connection (at DSR layer) are explicitly represented.
- Option 2: Assumes there is not flexibility at the DSR layer and the fixed cross-connection (at DSR layer) are not explicitly represented.
- Option 3: Assumes there is flexibility at the DSR layer.

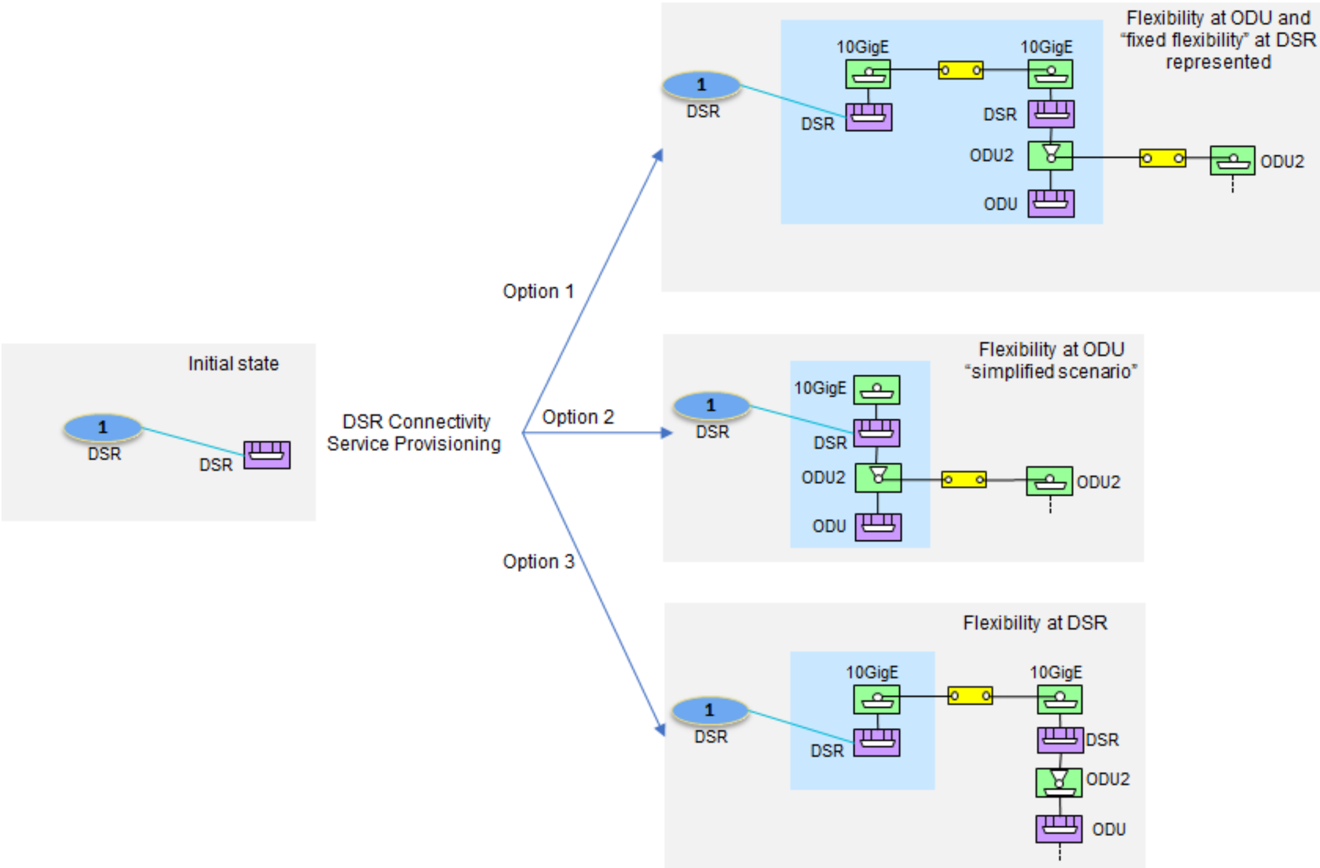


Figure 6-8 UNI Modelling simplifications

Please note, that the following versions of the RIA will address the UNI representation in detail so the previous simplifications may not be longer valid.

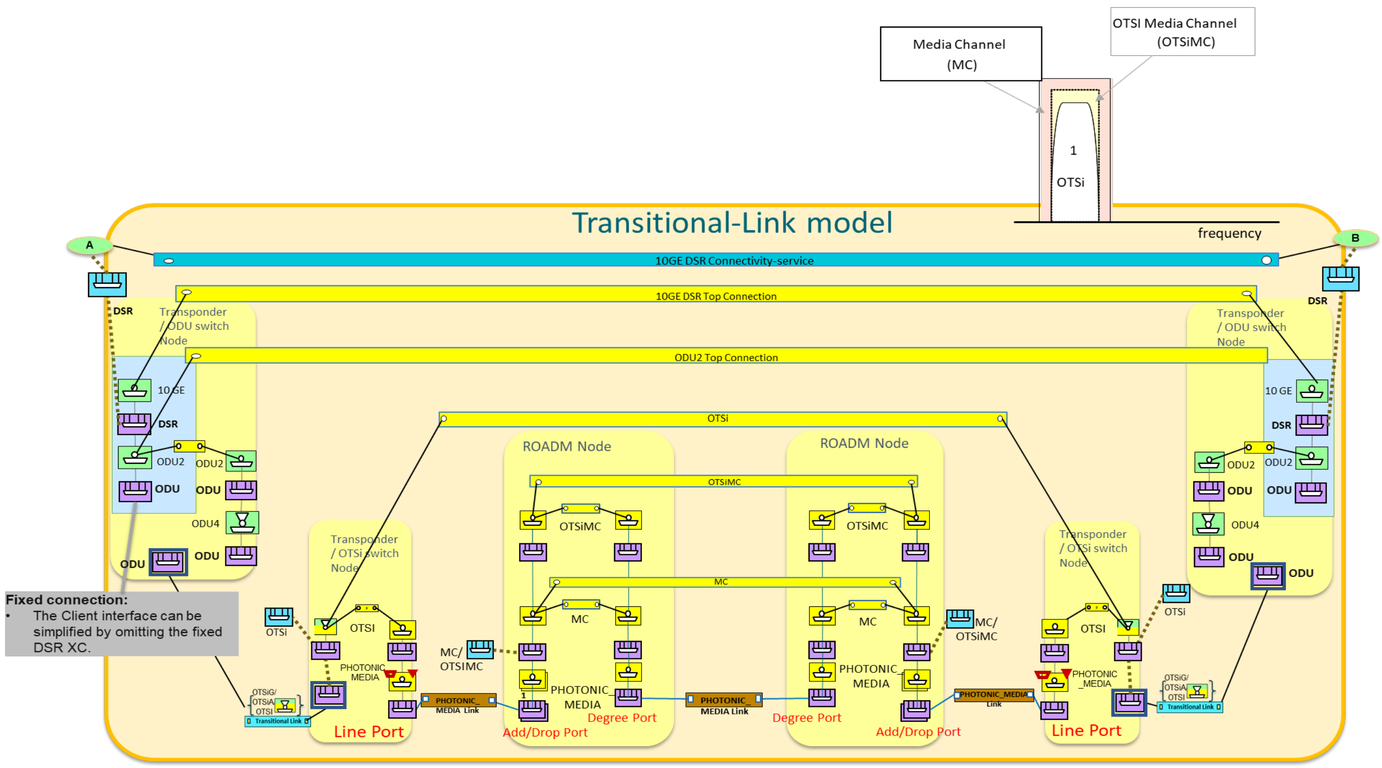


Figure 6-9 Connectivity Service 10GE client signal over ODU2 (DSR-ODU Fixed Mapping) over ODU4 over single OTSi – Transitional Link modelling

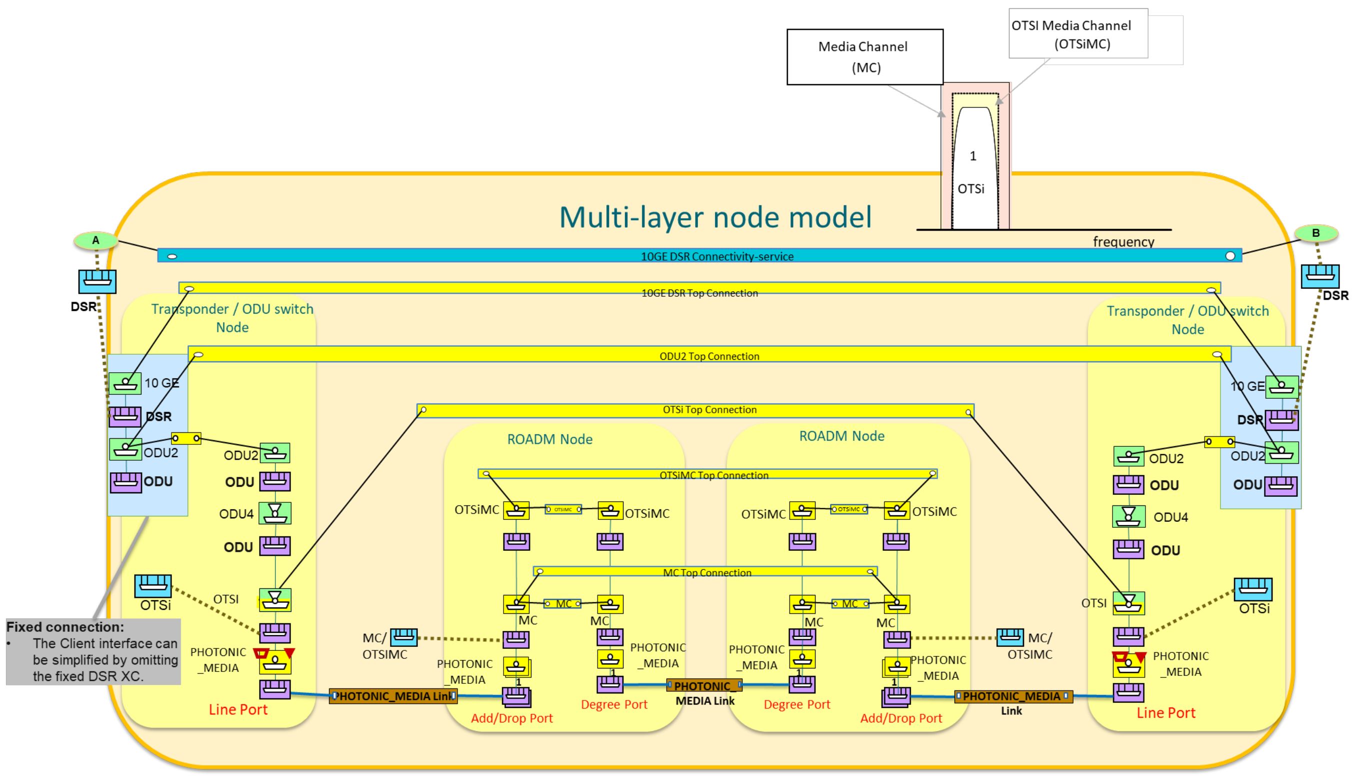


Figure 6-10 Connectivity Service 10GE client signal over ODU2 (DSR-ODU Fixed Mapping) over ODU4 over single OTSi – Multi-layer node modelling.

6.2.1.2.2 UC1a - Example 2: 1GE client signal over ODU0 over ODU2 over ODU4 (Fixed DSR-ODU mapping, flexible ODU allocation)

For simplification OTSi and Photonic Media layers have been omitted from this example, as it is the exact same case depicted in the previous example.

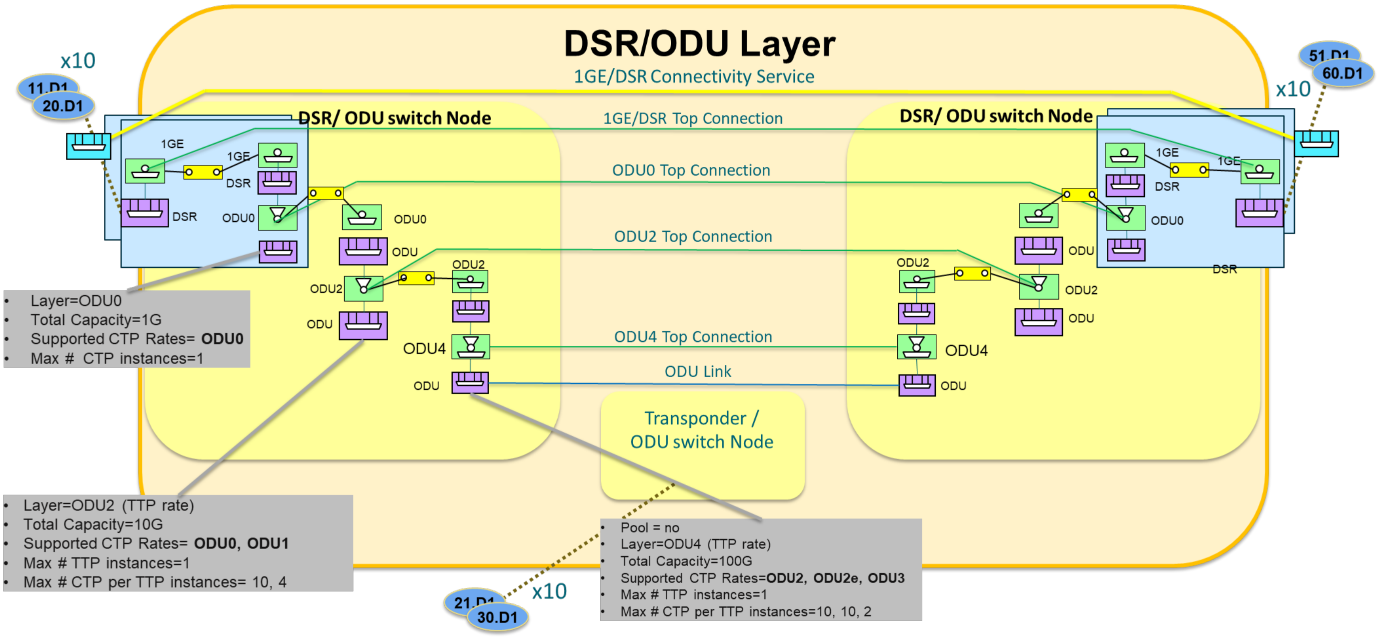


Figure 6-11 Connectivity Service 1GE client signal over ODU0 over ODU2 over ODU4 (Fixed DSR-ODU mapping, flexible ODU allocation)

6.2.1.2.3 UC1a - Example 3: 1GE client signal over ODU0 over ODU2 over ODU4 (Fixed DSR-ODU mapping, flexible ODU allocation) with intermediate ODU0 switching.

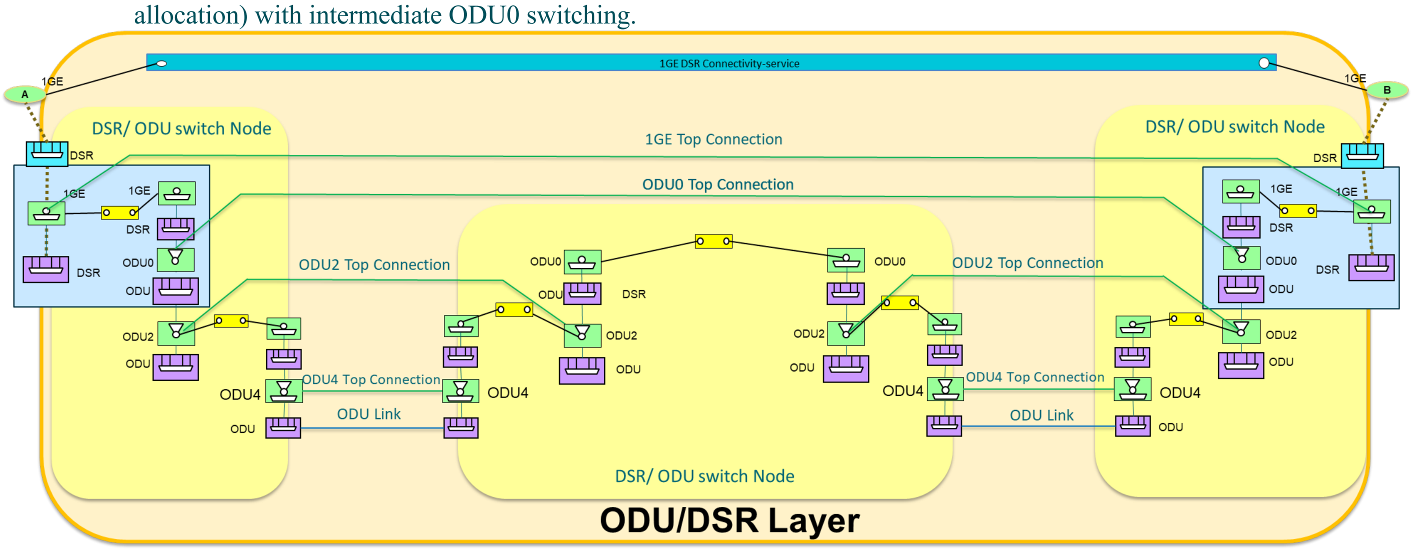


Figure 6-12 Connectivity Service 1GE client signal over ODU0 over ODU2 over ODU4 (Fixed DSR-ODU mapping, flexible ODU allocation) with intermediate ODU0 switching

6.2.1.2.4 UC1a - Example 4: 10GE client signal over ODU2 over ODU4 (Fixed DSR-ODU mapping, flexible ODU allocation) with intermediate transponder regeneration.

In this example introduces the intermediate regeneration (3R) of the optical channel (OTSi) into account from the modelling perspective. Please consider the following:

The TAPI Server MAY or MAY NOT expose SIPs for the regeneration board OTSi/OTSiA:

a) The TAPI Server does not expose SIPs for the regeneration board OTSi/ OTSiA interfaces, thus these resources are only available to be consumed by the internal control plane for regeneration of a higher layer (ODU, DSR) client connectivity-service. Currently we assume this approach.

b) The TAPI Server does expose SIPs for the regeneration board OTSi/OTSiA interfaces. So, the TAPI Server exposes to the user the creation of the two OTSi/ OTSiA connectivity services segments independently, to be used by higher layer services. Moreover, these SIPs also can be used to constrain a HO-ODUk or higher layer connectivity-service to use this 3R point. In this case, the user MUST request a creation of a CS including (for a regenerated service with a single regeneration point) four CSEPs with references to aEND, zEND and intermediate regeneration SIPs, in the CS request.

Assuming the option a), the expected connectivity model result in terms of hierarchy of connections within the CS's connection list, SHALL include a single HO-ODUk supported by N-segments of OTSi/OTSiA connections represented as ODU links between every regeneration stage (in this example a single regeneration results into 2 OTSI/OTSiA segments).

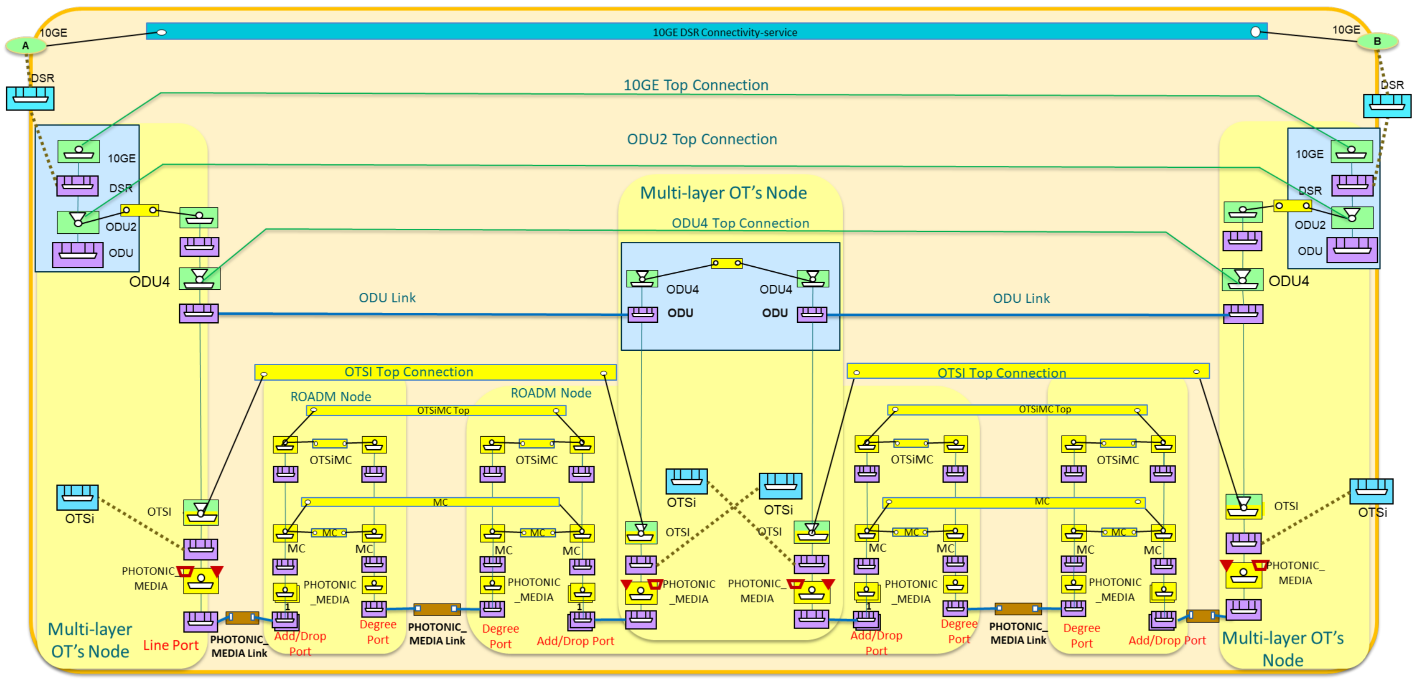


Figure 6-13 Connectivity Service 10GE client signal over ODU2 over ODU4 (Fixed DSR-ODU mapping, flexible ODU allocation) with intermediate transponder regeneration.

6.2.2 Use Case 1b: Unconstrained DSR Service Provisioning multi wavelength (beyond 100G).

Number	**UC1b**
Name	**Unconstrained DSR Service Provisioning multi wavelength (beyond 100G).**
Technologies involved	Optical
Process/Areas Involved	Planning and Operations
Brief description	The UC1 describes the provisioning of a ***tapi-connectivity:connectivity-service*** instance between service-interface-points exposed by the TAPI-Server at the DSR networking layer. **This service can include intermediate regeneration if necessary.** The underlying connection provisioning and management (including lower layer connections e.g., ODU, OTSi/OTSiA,MC/OTSiMC including intermediate regeneration connections if needed) is performed by the SDN Domain controller. The path of each lower layer connection (e.g., ODU or OCh/OTSi, OMS) across the network topology is calculated by the controller and the connection automatically provisioned. The "unconstrained" term refers that the TAPI-Client is not introducing any routing constraint in the service request, thus relays completely into the routing capabilities of the TAPI-Server to select the network resources employed to provide the desired service characteristics. Moreover, the TAPI-Client is not providing technology specific Traffic-Engineering constrains such time-slot selection at the ODU layer, or optical-spectrum selection for the routing of OTSi connections.
Layers involved	DSR/ODU/PHOTONIC_MEDIA
Type	Provisioning
Description & Workflow	This UC is implemented following the same workflow described in "Description & Workflow" of UC1a described in section 6.2.1

6.2.2.1 Expected results

The connection generation follows the rules detailed in section 5.1.1. In this use case it is expected that originally, the ODU and PHOTONIC_MEDIA layers are connected through a transitional link or collapsed in a multi-layer node.

This use case triggers the generation of an ODUcN Top Connection which is realize by N number of OTSi Top Connections required to transport the service. N Top OTSi Connctions are thus, generated over the same parent NEP (which only includes PHOTONIC_LAYER_QUALIFIER_OTSi within its ***supported-layer-qualifier*** list).

The ODUcN CEPs must include the FEC attributes specified in ODU-Connection-end-point-spec object definition in Table 17. **Please note that these attributes have been duplicated at the ODU layer from the ones defined at the OTSI-Assembly-connection-end-point-spec object definition which is now marked as deprecated in the new TAPI v2.1.3, its use at the OTSiA layer is not anymore recommended**.

Additionally, the model may include an OTSiA/OTSiG MEP for monitoring purposes, however the monitoring capabilities of the model will be described in next releases of this specification.

Please see the detail graphical description according to the Transitional-link and Multi-Layer models in Figure 6-14 and Figure 6-15 respectively.

6.2.2.1.1 UC1b - Example 5: 200GE over ODUc2 connectivity-service over two OTSi/OTSiMC over a single physical (PHOTONIC-MEDIA_OMS) port. Transitional link.

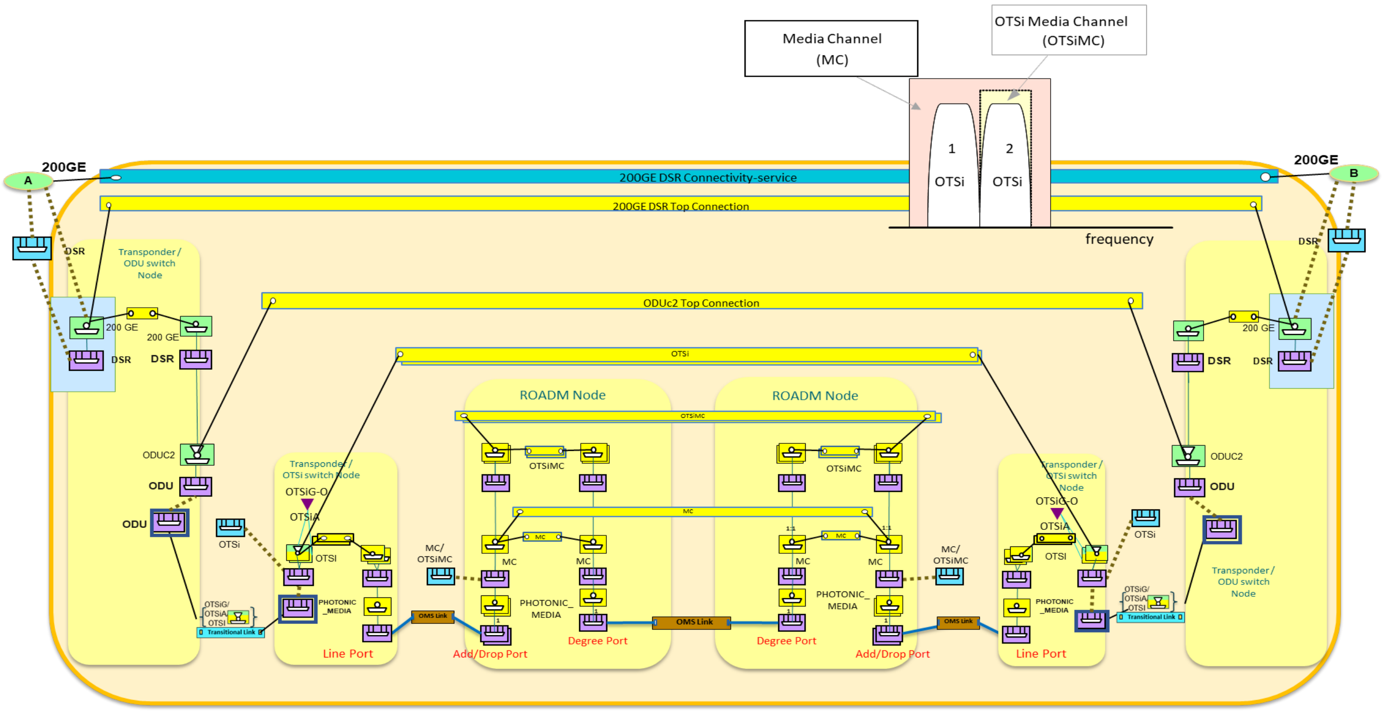


Figure 6-14 200GE over ODUc2 connectivity-service over two OTSi/OTSiMC over a single physical (PHOTONIC-MEDIA_OMS) port (Transitional Link).

6.2.2.1.2 UC1b - Example 5: 200GE over ODUc2 connectivity-service over two OTSi/OTSiMC over a single physical (PHOTONIC-MEDIA_OMS) port. Multi-layer node.

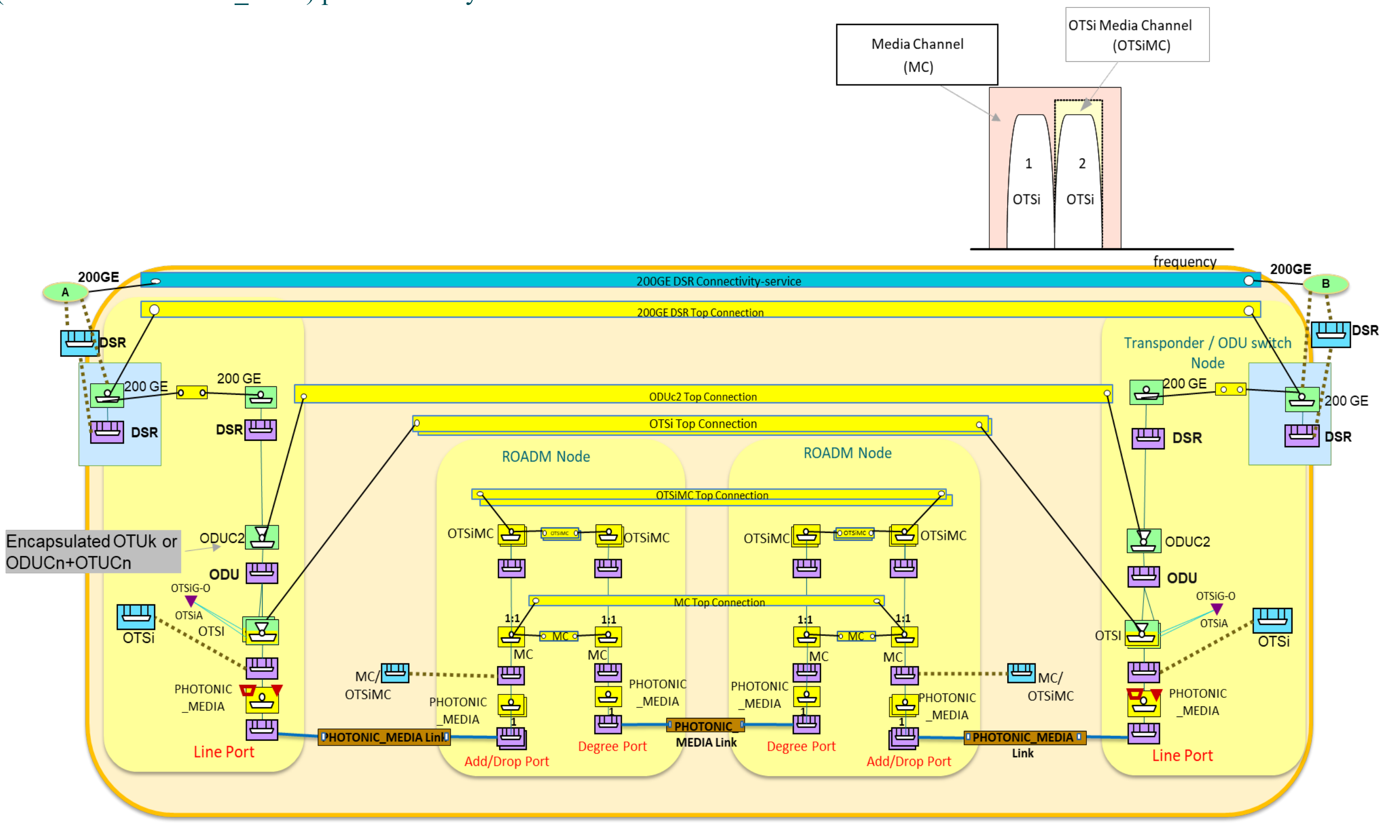


Figure 6-15 200GE over ODUc2 connectivity-service over two OTSi/OTSiMC over a single physical (PHOTONIC-MEDIA_OMS) port (Multi-Layer Node).

6.2.3 Use case 1c: Unconstrained ODU Service Provisioning

[TO BE DEFINED]

6.2.4 Use case 1d: Unconstrained PHOTONIC_MEDIA/OTSi Service Provisioning

Number	**UC1d**
Name	**Unconstrained PHOTONIC_MEDIA/OTSi Service Provisioning**
Technologies involved	Optical
Process/Areas Involved	Planning and Operations
Brief description	The UC1 describes the provisioning of a ***tapi-connectivity:connectivity-service*** instance between service-interface-points exposed by the TAPI-Server at the PHOTONIC_MEDIA networking layer. The underlying connection provisioning and management (including lower layer connections e.g., OTS/OMS, OTSiMC, MC including intermediate regeneration connections if needed) is performed by the SDN Domain controller. The path of each lower layer connection (e.g., OTSiMC, MC, OMS/OTS) across the network topology is calculated by the controller and the connection automatically provisioned. The "unconstrained" term refers that the TAPI-Client is introducing any routing constraint in the service request, thus relays completely into the routing capabilities of the TAPI-Server to select the network resources employed to provide the desired service characteristics. Moreover, the TAPI-Client is not providing technology specific Traffic-Engineering constrains such optical-spectrum selection for the routing of OTSi connections.
Layers involved	PHOTONIC_MEDIA
Type	Provisioning
Description & Workflow	This UC is implemented following the same workflow described in "Description & Workflow" of UC1a described in section 6.2.1

6.2.4.1 Expected results

This use case requires the relevant Service Interface Points (SIPs) attached to the corresponding OTSI NEPs are available and exposed by the TAPI server.

The connection generation MUST follows the rules detailed in section 5.1.1.

6.2.4.1.1 UC1d - Example 6: OTSi Single Wavelength Connectivity service provisioning. Transitional link approach

In this use case it is expected that originally, the ODU and PHOTONIC_MEDIA layers are connected through a transitional link. Please see the detail graphical description in Figure 6-16:

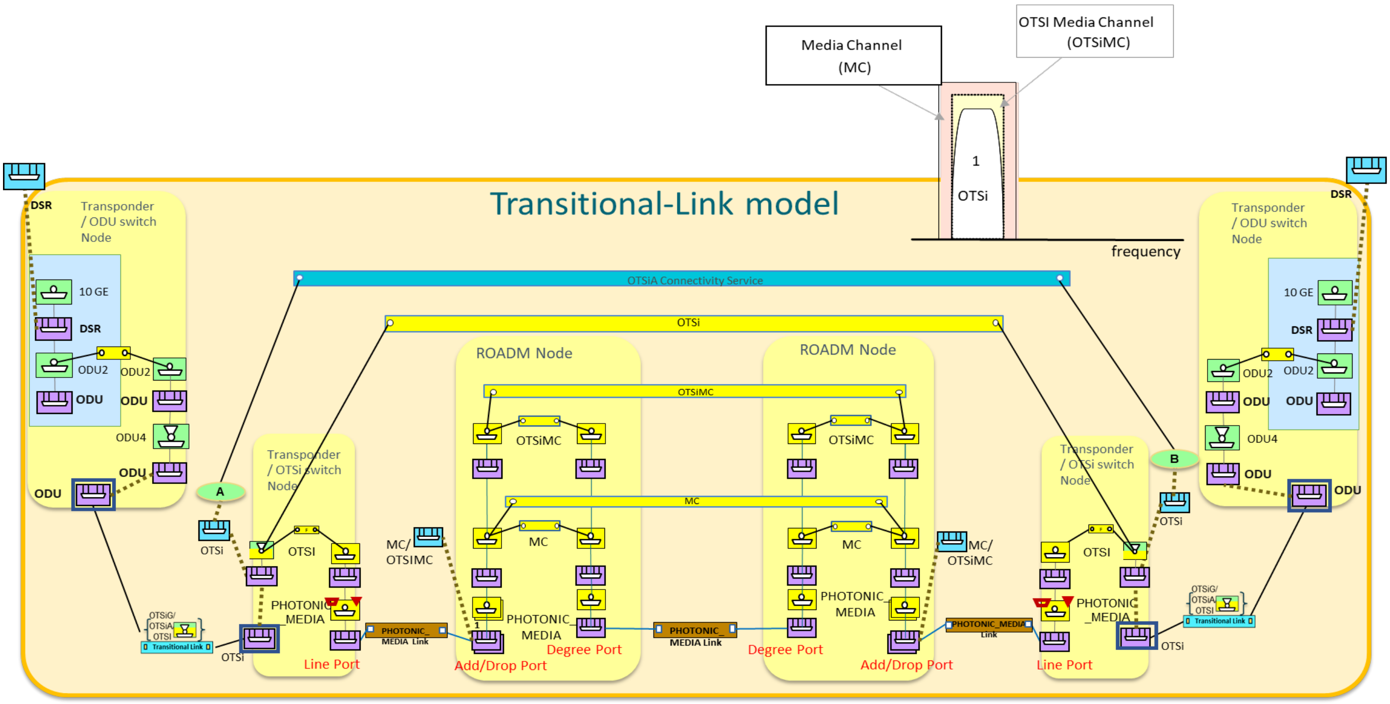


Figure 6-16 OTSi single lambda connectivity-service - Transitional link model.

6.2.4.1.2 UC1d - Example 7: OTSi Single Wavelength Connectivity service provisioning. Multi-layer node approach

In this use case it is expected that originally, the ODU and PHOTONIC_MEDIA layers integrated in a Multi-layer Node. Please see the detail graphical description in Figure 6-17:

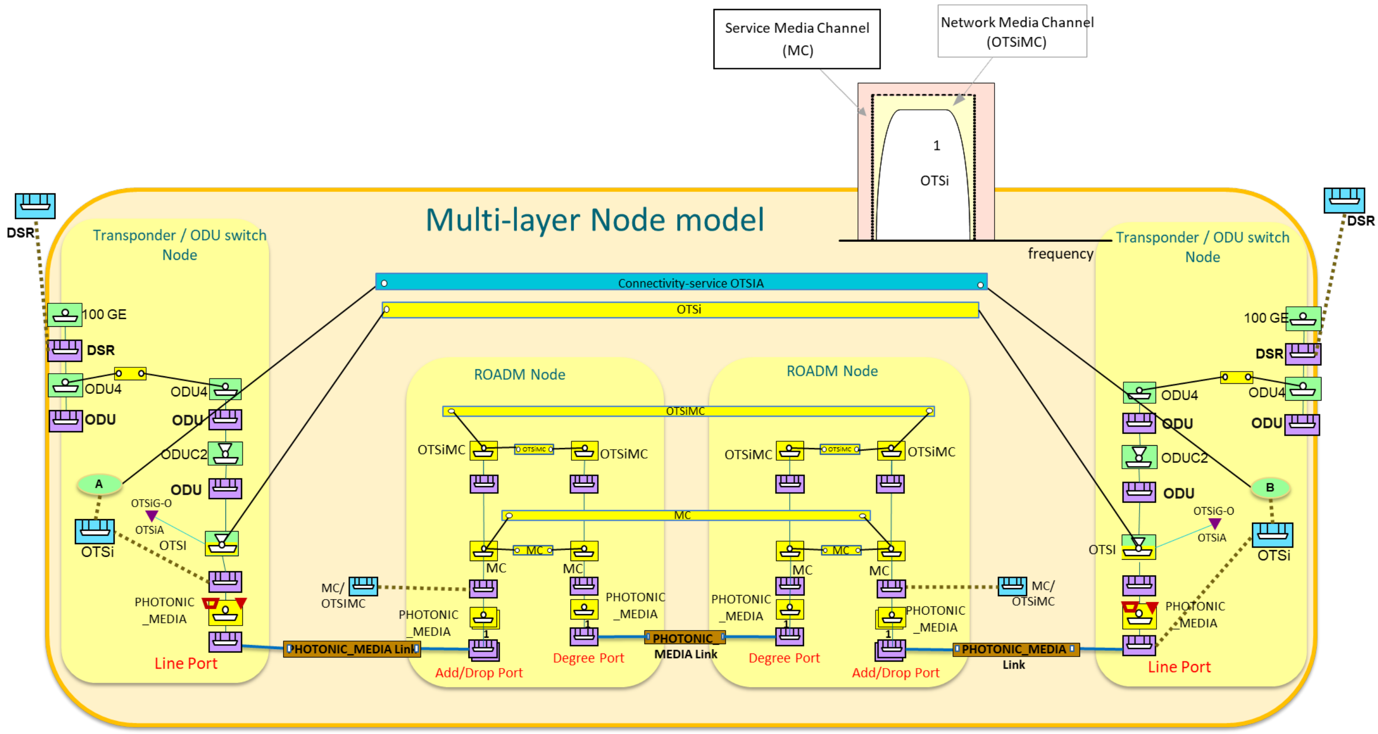


Figure 6-17 OTSi single lambda connectivity-service - Multi-layer node approach.

6.2.5 Use case 1e: Unconstrained PHOTONIC_MEDIA/OTSiA Service Provisioning

Number	**UC1e**
Name	**Unconstrained PHOTONIC_MEDIA/OTSiA Service Provisioning**
Technologies involved	Optical
Process/Areas Involved	Planning and Operations
Brief description	The UC1 describes the provisioning of a ***tapi-connectivity:connectivity-service*** instance between service-interface-points exposed by the TAPI-Server at the PHOTONIC_MEDIA networking layer. The underlying connection provisioning and management (including lower layer connections e.g., OTS/OMS, OTSiMC, MC including intermediate regeneration connections if needed) is performed by the SDN Domain controller. The path of each lower layer connection (e.g., OTS/OMS, OTSiMC, MC) across the network topology is calculated by the controller and the connection automatically provisioned. The "unconstrained" term refers that the TAPI-Client is introducing any routing constraint in the service request, thus relays completely into the routing capabilities of the TAPI-Server to select the network resources employed to provide the desired service characteristics. Moreover, the TAPI-Client is not providing technology specific Traffic-Engineering constrains such as optical-spectrum selection for the routing of OTSi connections.
Layers involved	PHOTONIC_MEDIA
Type	Provisioning
Description & Workflow	This UC is implemented following the same workflow described in "Description & Workflow" of UC1a described in section 6.2.1

6.2.5.1 Expected results

This use case requires the relevant Service Interface Points (SIPs) attached to the corresponding OTSI NEPs are available and exposed by the TAPI server.

The connection generation follows the rules detailed in section 5.1.1. In this use case it is expected that originally, the ODU and PHOTONIC_MEDIA layers are connected through a transitional link.

This case requires the generation of N number of OTSi Top Connections required to transport the service. N Top OTSi Connections are thus, generated over the same parent NEP (which only includes PHOTONIC_LAYER_QUALIFIER_OTSi within its ***supported-layer-qualifier*** list).

Additionally, the model may include an OTSiA/OTSiG MEP for monitoring purposes, however the monitoring capabilities of the model will be described in next releases of this specification.

Please see the detail graphical description in Figure 6-18 and Figure 6-19.

6.2.5.1.1 UC1e - Example 8: OTSiA multi-wavelength connectivity-service.

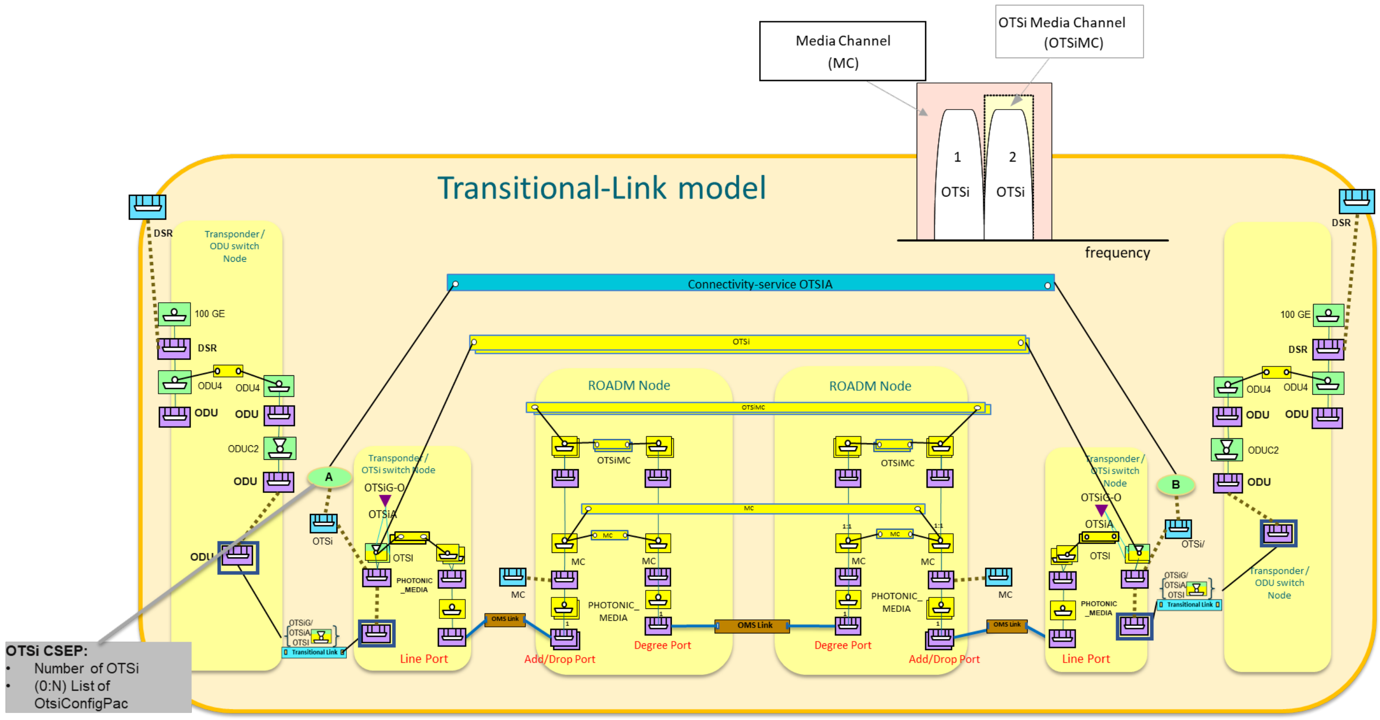


Figure 6-18 OTSiA multi-wavelength connectivity-service (transitional link model abstraction).

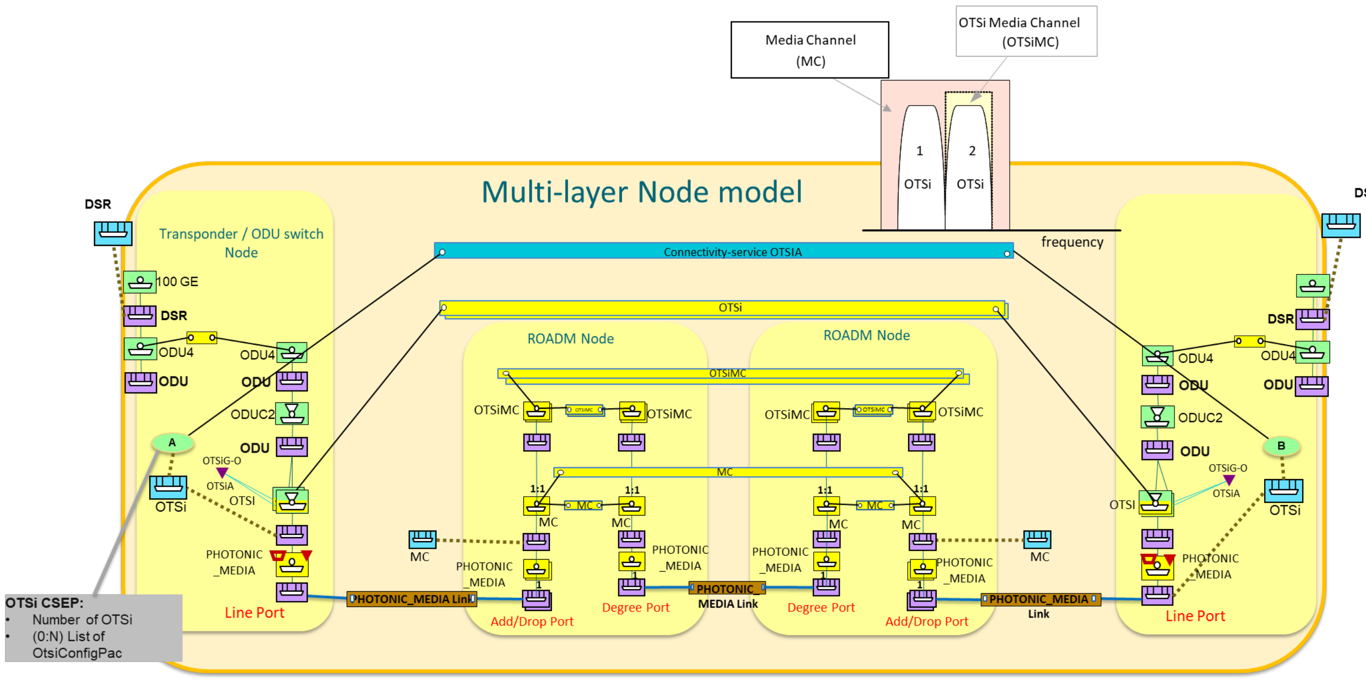
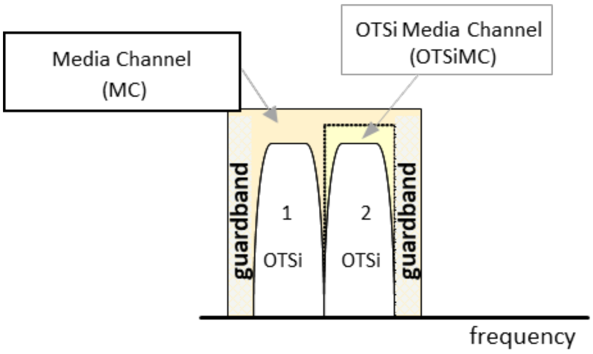


Figure 6-19 OTSiA multi-wavelength connectivity-service (multi-layer node model abstraction).

6.2.6 Use case 1f: Unconstrained PHOTONIC_LAYER_QUALIFER_MC Service Provisioning

Number	**UC1f**
Name	**Use case 1f: Unconstrained PHOTONIC_LAYER_QUALIFER_MC Service Provisioning**
Technologies involved	Optical
Process/Areas Involved	Planning and Operations
Brief description	The UC1 describes the provisioning of a ***tapi-connectivity:connectivity-service*** instance between service-interface-points exposed by the TAPI-Server at the PHOTONIC_LAYER_QUALIFER_MC networking layer. **This service can not include intermediate regeneration.** This use case is intended to define the way the TAPI Client can request the creation of a media-channel service which reserves a portion of optical spectrum across the PHOTONIC_MEDIA layer. This MC is generally wider than the OTSI occupied spectrum which is going to occupy it because it also includes the guardband between channels. The graphical representation of the relationship between MC, OTSIMC and OTSI signal is depicted in Figure 6-20.  **Figure 6-20 Media-channel entities relationship.** The underlying connection provisioning and management is performed by the SDN Domain controller. The path of each lower layer connection (e.g., OMS, OTS) across the network topology is calculated by the controller and the connection automatically provisioned. The "unconstrained" term refers that the TAPI-Client is not allowed to introduce any routing constraint in the service request, thus relays completely into the routing capabilities of the TAPI-Server to select the network resources employed to provide the desired service characteristics. Moreover, the TAPI-Client is not providing technology specific Traffic-Engineering constrains such spectrum-band selection for the MC connections. The PHOTONIC_MEDIA layer topology MUST be represented according to one of the following alternatives: • Bidirectional Topology abstraction. • Unidirectional Topology abstraction. Moreover, the Reconfigurable Optical Add Drop Multiplexers (ROADMs) Add/Drop ports MUST be represented as UNI interfaces with associated SIPs. The UNI interfaces MUST be represented according to one of the following alternatives: • Bidirectional UNI representation. • Unidirectional UNI representation. Please find more details in the following "Expected results" section.

	<u>DISCLAIMER</u>: The PHOTONIC_LAYER_QUALIFER_MC of TAPI v2.1.3 is equivalent to previous PHOTONIC_MEDIA_SMC of v2.1.2
Layers involved	PHOTONIC_MEDIA
Type	Provisioning
Description & Workflow	This UC is implemented following the same workflow described in "Description & Workflow" of UC1a described in section 6.2.1

6.2.6.1 Expected results

This use case accepts different variations according to the model directionality chosen to represent the PHOTONIC_MEDIA layer. The currently agreed solutions are three:

1. Full-bidirectional - UNI and PHOTONIC_MEDIA model.
2. Mixed-scenario - UNI bidirectional and topology unidirectional.
3. Full-unidirectional OLS scenario - UNI and PHOTONIC_MEDIA unidirectional

6.2.6.1.1 UC1f - Model 1: Full Bidirectional - UNI and PHOTONIC_MEDIA Topology

This choice corresponds to a solution exposed by the TAPI server where the relation between the Add/Drop directions of UNI interfaces is known by the TAPI server and the unidirectionality of the rest of the PHOTONIC_MEDIA layer is abstracted as a full-bidirectional topology.

In this approach the MC UNI interfaces are represented as bidirectional SIPs associated to Add/Drop PHOTONIC_MEDIA NEPs.

In this model, it is expected the lower layer NEPs (PHOTONIC_MEDIA), representing the Optical Terminals (OTs - Transponders/Muxponders) line interfaces and the ROADMs Add/Drop interfaces, are connected through a link with a 1:1 relationship at the PHOTONIC_MEDIA layer. The NEPs/CEPs and links at the bottom layer represents the OTS and OMS ITU-T switching layers, which SHALL be augmented by the necessary OAM monitoring capabilities of these layers.

MC Connectivity-Services are bidirectional too with a single bidirectional Top Connection representing the end-to-end route across the PHOTONIC_MEDIA layer.

Moreover, the MC Top Connection includes within the ***tapi-connectivity:lower-connection*** attribute the reference to the Cross-Connections (XCs) between the bidirectional PHOTONIC_LAYER_QUALIFIER_MC CEPs mounted over the bidirectional MC NEPs.

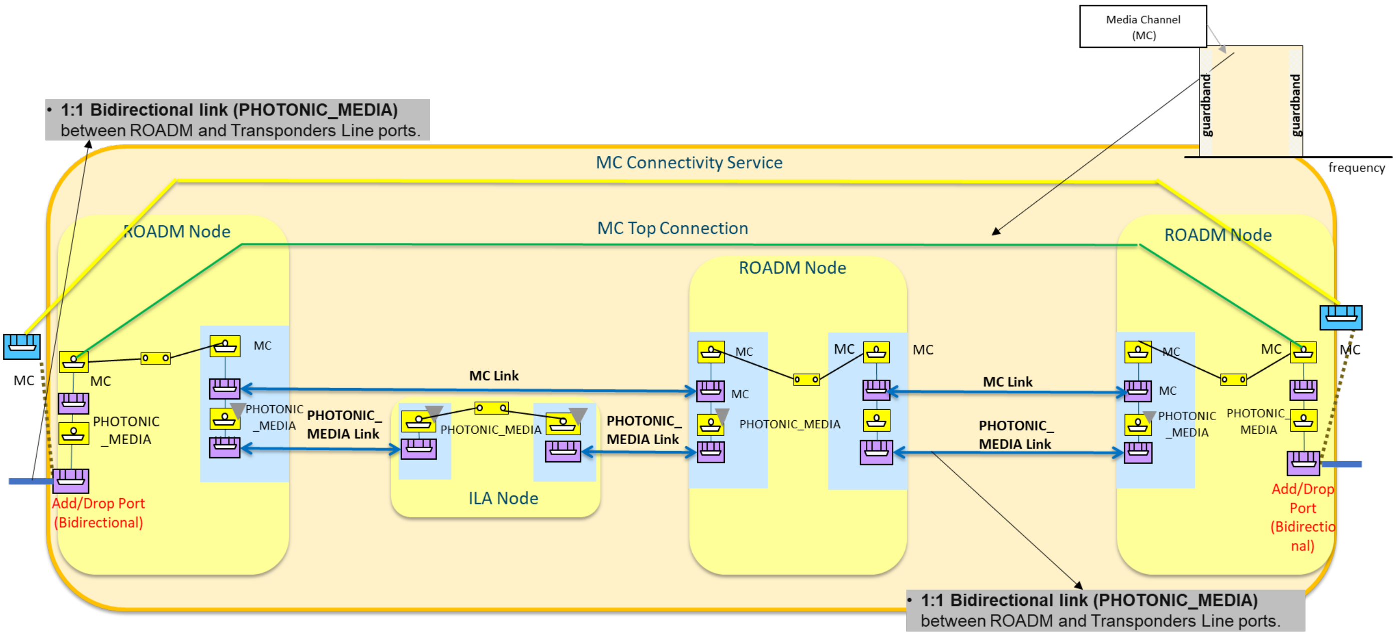


Figure 6-21 Full Bidirectional - UNI and OMS bidirectional scenario.

6.2.6.1.2 UC1f - Model 2: Mixed Scenario - UNI bidirectional and OMS unidirectional

The second alternative corresponds to a mixed solution exposed by the TAPI server where the relation between the Add/Drop directions of UNI interfaces is known by the TAPI server and thus, the MC UNI interfaces are represented as bidirectional SIPs associated to the Add/Drop PHOTONIC_MEDIA NEPs.

However, the PHOTONIC_MEDIA layer is abstracted as a unidirectional link topology.

In this model, it is expected that originally, the NEPs of PHOTONIC_MEDIA (supporting OTSi layer-protocol-qualifier CEPs) and PHOTONIC_MEDIA (supporting MC layer-protocol-qualifier CEPs) are connected through a link with a 1:1 relationship at the PHOTONIC_MEDIA layer.

MC Connectivity-services are bidirectional with two references to the bidirectional Add/Drop SIPs. Once successfully provisioned, the Connectivity-Service MUST reference a single bidirectional Top Connection representing the end-to-end route across the PHOTONIC_MEDIA layer.

The MC Top Connections includes, within the ***tapi-connectivity:lower-connection*** attribute, the references to the two point-to-multipoint Cross-Connections (XCs) connecting the bidirectional Add/Drop UNI interfaces to the ROADM degree unidirectional interfaces. Then the route traverses the remaining unidirectional PHOTONIC_MEDIA nodes till the far end. All XCs in the two directions MUST be included into the MC Top Connection lower-level connection list.

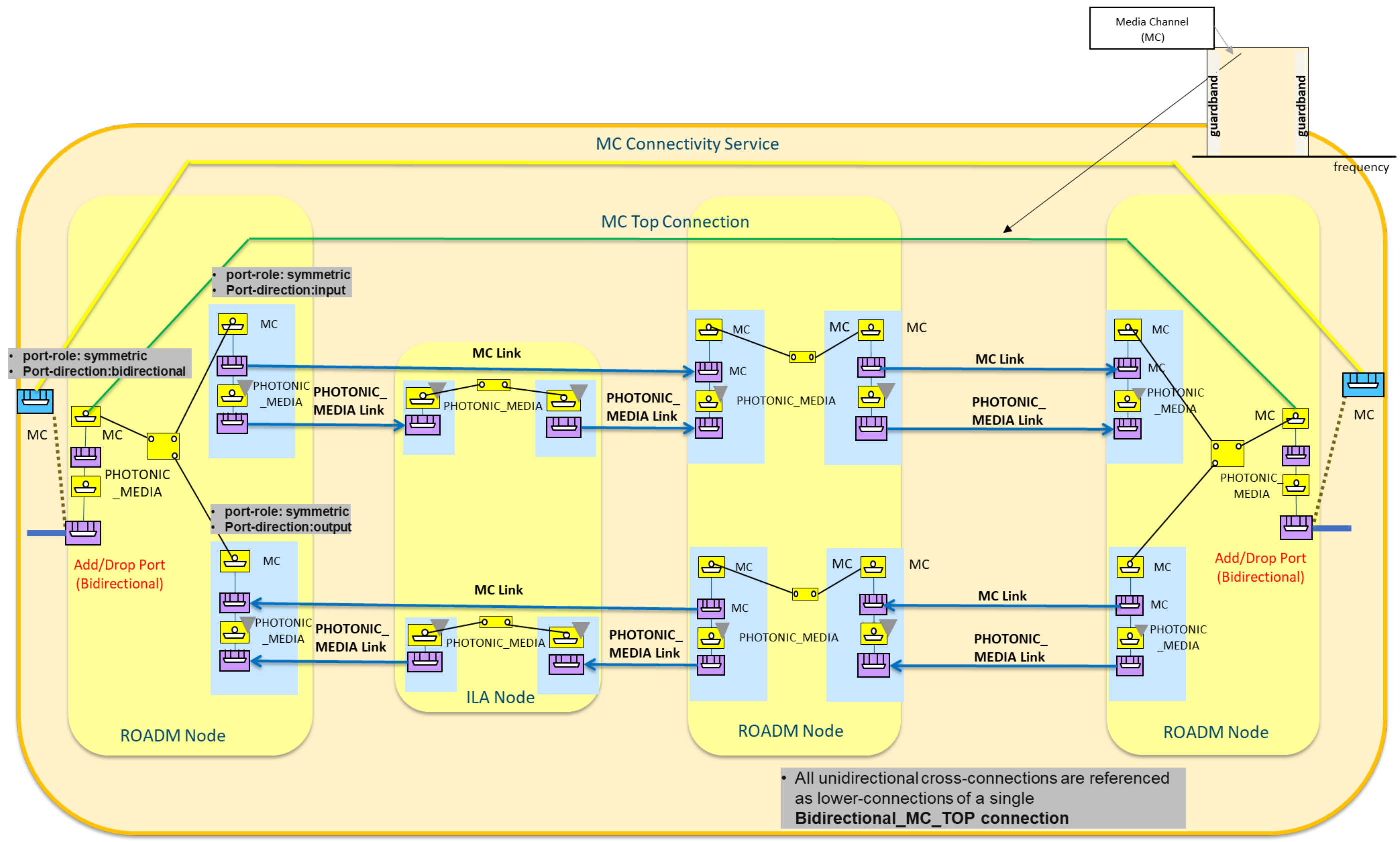


Figure 6-22 Mixed Scenario - UNI bidirectional and OMS unidirectional.

6.2.6.1.3 UC1f - Model 3: Full-unidirectional OLS scenario - UNI and PHOTONIC_MEDIA unidirectional

This model approach is only applicable in case of a realization of an Open Line System (OLS) in a disaggregated network. In this scenario, the OTs are not managed/controlled by the TAPI server and therefore, the relationship between OT's Line and OLS Add/Drop ports is also unknow by the TAPI server.

In case of aggregated domains, where both the OTs and OLS are managed by the same TAPI Server, the relationship between Add/Drop ports is known by the TAPI Server before the service provisioning and thus the **UNI interfaces MUST be represented as bidirectional entities as described in Model 1 or 2.**

Thus, this model corresponds to a solution exposed by the TAPI server where the relation between the Add/Drop directions of UNI interfaces is unknown by the TAPI server. In this case, the TAPI server does not know the relationship between Add and Drop ROADM's interfaces until the TAPI Client correlates them through the creation of a bidirectional Connectivity-Service.

In this modelling approach the MC UNI interfaces are represented as unidirectional SIPs associated to unidirectional Add/Drop OMS NEPs.

In this model, it is expected that once the TAPI Client correlates the PHOTONIC_MEDIA NEPs of OT's Line ports (supporting OTSI layer-protocol-qualifier CEPs) and Open Line System (OLS) Add/Drop

PHOTONIC_MEDIA NEPs (supporting MC layer-protocol-qualifier CEPs), it exposes the connectivity through a 1:2 asymmetric ***tapi-topology:link*** relationship at the PHOTONIC_MEDIA layer.

MC Connectivity-services are bidirectional with four references to the unidirectional Add/Drop SIPs at each CSEP. Once successfully provisioned, the Connectivity-Service MUST reference two unidirectional Top Connection representing the two directions end-to-end route across the PHOTONIC_MEDIA layer.

Moreover, the MC Top Connection includes the reference to the **lower-level** Cross-Connections (XCs) between the UNIDIRECTIONAL PHOTONIC_LAYER_QUALIFIER_MC CEPs mounted over the unidirectional MC supporting NEPs.

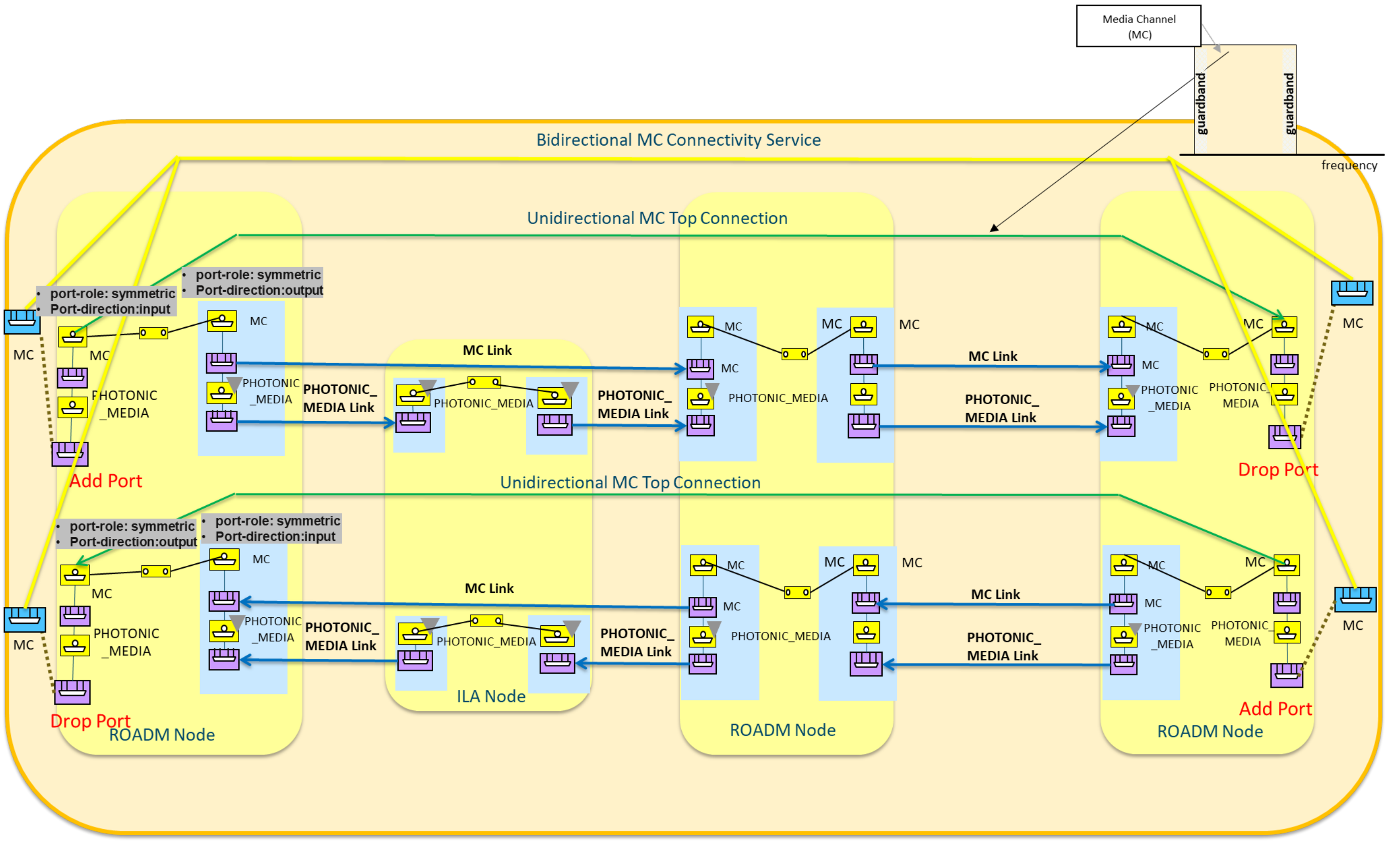


Figure 6-23 Full Unidirectional - UNI and OMS unidirectional scenario.

6.3 Constrained Provisioning

6.3.1 Use case 3a: Include/exclude a node or group of nodes.

Number	**UC3a**
Name	**Constrained provisioning: Include/exclude a node or group of nodes**
Technologies involved	Optical
Process/Areas Involved	Planning and Operations
Brief description	This use case consists on the requests of a connectivity service between two Connectivity Service End Points associated to Service Interface points at different layers (DSR, ODU or OTSi). This service is subject to the inclusion/exclusion of the nodes selected by the TAPI client. **This service can include intermediate regeneration if necessary.** **This topology constrains MUST be processed as a LOOSE constrain and it SHOULD be considered desirable but not mandatory when enter in conflict with other routing policies or re-routing condition such a restoration.** The underlying connection generation (including lower layer connections e.g., OCh/OTSi, Photonic Media and regeneration nodes) is made by the SDN Domain controller. The route of each lower layer connection (e.g., OCh/OTSi, Photonic Media) across the network topology is calculated by the SDN Domain controller considering the constraints introduced in the service request. The route in each layer is recursively calculated from the lower layer to the upper layer, excluding/including **ALL those components** of the topology shared with the connections created to support the selected service. In case the topology-constrains (nodes) are applied to a service with restoration capabilities, any reroute action SHOULD account for any inclusion/exclusion topology-constrain policy defined if possible but, as a general rule, the **restoration MUST always take place even if the topology-constrains enter in conflict with the new route.**
Layers involved	DSR/ODU/PHOTONIC_MEDIA
Type	Provisioning
Description & Workflow	The connectivity-service object sent by the TAPI Client to the Server to request the creation of the service MUST include the **include-node or exclude-node attributes (node-uuid)** The node-uuid referenced by the **inclusion or exclusion** attributes MUST be present in the TAPI Server context at the time of the CS provisioning is request. In case, the referenced CS element is invalid, the TAPI Server MUST return an error message in the RESTConf response message, according to the rules included in https://tools.ietf.org/html/rfc8040#section-7, with the “invalid-value” <error-tag>. This UC is implemented following the same workflow described in “Description & Workflow” of UC1a described in section 6.2.1

6.3.1.1 Required parameters

Table 25 complements the information included in the Use Case 1a definition, with the Connectivity-Service attributes required to implement this use case, this implies all attributes and objects included in Section 6.2.1.1 tables is required for this use case too.

Table 25: Connectivity-service node topology-constrains object definitions.

connectivity-service	**/tapi-common:context/tapi-connectivity:connectivity-context/connectivity-service**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
include-node	LeafList of { "[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}" }	RW	M	• This is a loose constraint - that is it is unordered and could be a partial list
exclude-node	LeafList of { "[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}" }	RW	M	• Reference to an existing node-uuid already present in the TAPI server context MUST be valid.

6.3.2 Use case 3b: Include/exclude a link or group of links.

Number	**UC3b**
Name	**Constrained provisioning: Include/exclude a link or group of links**
Technologies involved	Optical
Process/Areas Involved	Planning and Operations
Brief description	This use case consists on the requests of a connectivity service between two Connectivity Service End Points associated to Service Interface points at different layers (DSR, ODU or OTSi). This service is subject to the inclusion/exclusion of the links selected by the tapi client. **This service can include intermediate regeneration if necessary.** **This topology constrains MUST be processed as a LOOSE constrain and it SHOULD be considered desirable but not mandatory when enter in conflict with other routing policies or re-routing condition such a restoration.** The underlying connection generation (including lower layer connections e.g., OCh/OTSi, Photonic Media and regeneration nodes) is made by the SDN Domain controller. The route of each lower layer connection (e.g., OCh/OTSi, Photonic Media) across the network topology is calculated by the SDN Domain controller considering the constraints introduced in the service request. The route in each layer is recursively calculated from the lower layer to the upper layer, excluding/including **ALL those components** of the topology shared with the connections created to support the selected service. In case the topology-constrains (links) are applied to a service with restoration capabilities, any reroute action SHOULD account for any inclusion/exclusion topology-constrain policy defined if possible but, as a general rule, the **restoration MUST always take place even if the topology-constrains enter in conflict with the new route.**
Layers involved	DSR/ODU/PHOTONIC_MEDIA
Type	Provisioning

Description & Workflow	The connectivity-service object sent by the TAPI Client to the Server to request the creation of the service MUST include the **include-link or exclude-link attributes (link-uuid)** The link-uuid referenced by the **inclusion or exclusion** attributes MUST be present in the TAPI Server context at the time of the CS provisioning is request. In case, the referenced CS element is invalid, the TAPI Server MUST return an error message in the RESTConf response message, according to the rules included in https://tools.ietf.org/html/rfc8040#section-7, with the "invalid-value" <error-tag>. This UC is implemented following the same workflow described in "Description & Workflow" of UC1a described in section 6.2.1

6.3.2.1 Required parameters

Table 26 complements the information included in the Use Case 1a definition, with the Connectivity-Service attributes required to implement this use case, this implies all attributes and objects included in Section 6.2.1.1 tables is required for this use case too.

Table 26: Connectivity-service link topology-constrains object definitions.

connectivity-service	**/tapi-common:context/tapi-connectivity:connectivity-context/connectivity-service**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
include-link	LeafList of { "[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}" }	RW	M	• This is a loose constraint - that is it is unordered and could be a partial list
exclude-link	LeafList of { "[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}" }	RW	M	• Reference to an existing link-uuid already present in the TAPI server context MUST be valid.

6.3.3 Use case 3c: Include/exclude the route used by other service.

Number	**UC3c**
Name	Constrained provisioning: Include/exclude the path used by other service.
Technologies involved	Optical
Process/Areas Involved	Planning and Operations
Brief description	This use case consists on the requests of a connectivity service between two Connectivity Service End Points associated to Service Interface points at different layers (DSR, ODU or OTSi). This service is subject to the inclusion/exclusion of the path associated to one or more already provisioned services. **This service can include intermediate regeneration if necessary.** **This topology constrains MUST be processed as a LOOSE constrain and it SHOULD be considered desirable but not mandatory when enter in conflict with other routing policies or re-routing condition such a restoration.** The underlying connection generation (including lower layer connections e.g., OCh/OTSi, Photonic Media and regeneration nodes) is made by the SDN Domain controller. The route of each lower layer connection (e.g., OCh/OTSi, Photonic Media) across the network topology is calculated by the SDN Domain controller considering the constraints introduced in the service request. The route in each layer is recursively calculated from the lower layer to the upper layer, excluding/including **ALL those components** of the topology shared with the connections created to support the selected service. For the coroute-inclusion capability, the CS's Top Connection's routes MUST share the same potentially shared resources (i.e., links) of the referenced service in all layers from DSR to PHOTONIC_MEDIA_OMS when it applies. Moreover, the Top Connection's routes (including the new path when the restoration is triggered) of a CS implementing coroute-inclusion or diversity-exclusion policies, MUST include/exclude the all the potentially shared resources (i.e., links) related to the route objects referenced by all the connections (Top Connections) conforming the referenced CS. This condition **SHOULD be applied if possible, at any time** the TAPI Server in charge of the service, performs a path computation action for the target service i.e., **at the provisioning or restoration actions**. In case the referenced connectivity-service's path changes due to a restoration or any other reroute action and this new path enters in conflict with the conditions impose to any other service referencing it by the coroute-inclusion or diversity-exclusion attribute, the latter is not forced to proactively reroute but if any user or control plane action force it to re-route then the conditions will need to be meet again.
Layers involved	DSR/ODU/PHOTONIC_MEDIA
Type	Provisioning
Description & Workflow	The connectivity-service object sent by the TAPI Client to the Server to request the creation of the service MUST include the **coroute-inclusion** or **diversity-exclusion** attributes set**.** The connectivity-service/s referenced by the **coroute-inclusion or diversity-exclusion** attributes MUST be present in the TAPI Server context at the time of the CS provisioning is request. In the case the referenced CS element is invalid the TAPI Server MUST return an error message in the RESTConf response message, according to the rules included in https://tools.ietf.org/html/rfc8040#section-7, with the "invalid-value" <error-tag>. In case the referenced CS by the **coroute-inclusion or diversity-exclusion** attributes change its route, this service will not change accordingly, in other words, the TAPI server is not aimed to maintain a stateful relation between the services. This UC is implemented following the same workflow described in "Description & Workflow" of UC1a described in section 6.2.1

6.3.3.1 Required parameters

Table 27 complements the information included in the Use Case 1a definition in Table 14, with the Connectivity-Service attributes required to implement this use case, this implies all attributes and objects included in Section 6.2.1.1 tables is required for this use case too.

Table 27: Connectivity-service diversity-exclusion and coroute-inclusion object definitions.

connectivity-service				
coroute-inclusion	{connectivity-service-uuid: **connectivity-service-ref** - */tapi-common:context/tapi-connectivity:connectivity-context/connectivity-service/uuid* }	RW	M	• Reference to an existing ConnectivityService already present in the TAPI server context MUST be valid.
coroute-exclusion	{connectivity-service-uuid: **connectivity-service-ref** - */tapi-common:context/tapi-connectivity:connectivity-context/connectivity-service/uuid* }	RW	M	• Reference to an existing ConnectivityService already present in the TAPI server context MUST be valid.

6.4 Inventory

6.4.1 Use case 4a: Introduction of references to external inventory model.

Number	**UC4a**
Name	Introduction of references to external inventory model.
Technologies involved	Optical
Process/Areas Involved	Planning and Operations
Brief description	Hardware identifiers currently stored in OSS inventory systems must be correlated with SDN-C NBI logical endpoint's identifiers. This information MUST be provided by the SDN-C suppliers. For every inventory element represented as a logical element in TAPI by the SDN Domain controller, an INVENTORY_ID tapi-common:name property shall be included into the logical element construct. It is expected the INVENTORY_ID tag will be included for the following TAPI objects: • *tapi-topology:node-edge-point* • *tapi-common:service-interface-point* Note: The INVENTORY_ID value format is defined in section 4.2, which defines how to express the relative position of each component.
Layers involved	DSR/ODU/PHOTONIC_MEDIA
Type	Inventory
Description & Workflow	

6.4.2 Use case 4b: Complete Inventory model for NBI Interface.

Number	**UC4b**
Name	Complete Inventory model for NBI Interface.
Technologies involved	Optical
Process/Areas Involved	Planning and Operations
Brief description	There should be no differences between the inventory data that the SDN controller can send to the OSS through the NBI and the inventory data that the OSS could get by developing a direct interface to the network equipment. In case of any differences between both (via NMS/EMS or direct integration), the manufacturer must argue the reason for the difference and indicate the initiatives to close the gaps, associating them to the SDN controller roadmaps. The proposal for mapping the parameters relative to the HW inventory is to use the "equipment" model structure. This structure is described in the following path of T-API 2.1.3 **/tapi-common:context/tapi-equipment:physical-context** **The basic structure of the equipment does not include the rack position as a mandatory field within the NBI. This Rack position should be added to the database of OSS once the information is provided by the installers or from an external database.**
Layers involved	DSR/ODU/PHOTONIC_MEDIA
Type	Inventory
Description & Workflow	The workflow consists of the retrieval of the inventory information required by the user (TAPI client) using different filters. This information MUST be requested at least in the following way: • Full inventory of all "devices" with all their parameters • Full inventory of equipment (chassis, slot, ports/pluggables) and the hierarchy representation of the equipment within a device or a group of devices with their parameters. • Full inventory of the equipment used within a connectivity service or a precalculated-path • Possibility of filtering. By type (category), by part-number, manufacturer or by any parameter included in the TAPI model.

6.4.2.1 Required parameters

All these parameters must be included in the structure of the equipment (cards, chasis, pluggables).

The basic structure of the equipment does not include the rack position as a mandatory field within the NBI. This Rack position should be added to the database of OSSs once the information is provided by the installers or from an external database.

The main parameters required are included in Table 28.

Table 28: Main parameters for equipment model required.

Conceptual parameter	TAPI xPath reference
Part number	/tapi-common:context/tapi-equipment:physical-context/device/equipment/actual-equipment/common-actual-properties/asset-instance-identifier
Serial number	/tapi-common:context/tapi-equipment:device/equipment/actual-equipment/common-actual-properties/serial-number
Name	/tapi-common:context/tapi-equipment:physical-context/device/equipment/name
Description	/tapi-common:context/tapi-equipment:physical-context/device/equipment/actual-equipment/ common-equipment-properties/equipment-type-description
Component Version	/tapi-common:context/tapi-equipment:physical-context/device/equipment/actual-equipment/ common-equipment-properties/equipment-type-version
Type	/tapi-common:context/tapi-equipment:physical-context/device/equipment/category
Relative position of the component into the network element	/tapi-common:context/tapi-equipment:physical-context/device/equipment/contained-holder/actual-holder/common-holder-properties/holder-location
Operational state	/tapi-common:context/tapi-topology:topology-context/topology/node/owned-node-edge-point/operational-state
Admin state	/tapi-common:context/tapi-topology:topology-context/topology/node/owned-node-edge-point administrative-state
Removable	/tapi-common:context/tapi-equipment:physical-context/device/equipment/contained-holder/actual-holder/common-holder-properties/is-guided
Manufacturer	/tapi-common:context/tapi-equipment:physical-context/device/equipment/actual-equipment/ common-equipment-properties/manufacturer-name
Operator_ID_type	/tapi-common:context/tapi-equipment:physical-context/equipment/contained-holder/actual-holder/common-holder-properties/asset-type-identifier

The following parameters must be included for each item and they must be present in the following path**:**
/tapi-common:context/tapi-equipment:physical-context

Table 29: Equipment object's attributes required for UC4b.

equipment	**/tapi-common:context/tapi-equipment:physical-context/device/equipment**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
contained-holder	List of { occupying-fru, expected-holder, actual-holder, uuid , name}	RO	M	• Provided by *tapi-server*

	• occupying-fru {device-uuid, equipment-uuid} • expected-holder/**common-holder-properties** • actual-holder/**common-holder-properties** • uuid • name {value-name, value} ○ "value-name":"HOLDER_NAME" ○ "value":"[0-9a-zA-Z_]{64}"			Represent all the children contained in the equipment
category	[RACK, SUBRACK, CIRCUIT_PACK, SMALL_FORMFACTOR_PLUGGABLE, STAND_ALONE_UNIT]	RO	M	• Provided by *tapi-server*
equipment-location	String	RO	M	• Provided by tapi-server
geographical-location	String	RO	M	• Provided by *tapi-server*
is-expected-actual-mismatch	Boolean	RO	M	• Provided by *tapi-server*
expected-equipment	List of {expected-non-field-replaceable-module, holder, common-equipment-properties}	RO	M	• Provided by *tapi-server*
actual-equipment	{actual-non-field-replaceable-module, common-actual-properties, common-equipment-properties}	RO	M	• Provided by *tapi-server*
name	List of {value-name: value} • "value-name": "EQUPMENT_NAME" "value": " [0-9a-zA-Z_]{64}"	RO	M	• Provided by *tapi-server*
uuid	"[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}"	RO	M	• As per RFC 4122 • Provided by ***tapi-server***

Table 30: Common-holder-properties object's attributes required for UC4b.

common-holder-properties	**/tapi-common:context/tapi-equipment:physical-context/device/equipment/contained-holder/actual-holder/common-holder-properties**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
holder-category	[HOLDER_CATEGORY_SLOT]	RO	M	• Provided by *tapi-server* A guided holder with fixed connectors. The guided holder is designed to take a particular form of CIRCUIT_PACK or SMALL_FORMFACTOR_PLUGGABLE
is-guided	Boolean	RO	M	• Provided by *tapi-server* This attribute indicates whether the holder has guides that constrain the position of the equipment in the holder or not.
holder-location	String	RO	M	• Provided by tapi-server

				The relative position of the holder in the context of its containing equipment along with the position of that containing Equipment (and further recursion).

Table 31: Common-equipment-properties object's attributes required for UC4b.

common-equipment-properties	**/tapi-common:context/tapi-equipment:physical-context/device/equipment/actual-equipment/common-equipment-properties**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
asset-type-identifier	String	RO	O	• Provided by *tapi-server* Represents the invariant properties of the equipment asset allocated by the operator that define and characterize the type **Operator_ID_type**
equipment-type-description	String	RO	M	• Provided by *tapi-server* Text describing the type of Equipment. **Description**
equipment-type-identifier	String	RO	M	• Provided by tapi-server This attribute identifies the part type of the equipment
equipment-type-name	String	RO	M	• Provided by *tapi-server* This attribute identifies the type of the equipment.
equipment-type-version	String	RO	M	• Provided by *tapi-server* This attribute identifies the version of the equipment. **Version**
manufacturer-identifier	String	RO	M	• Provided by *tapi-server* The formal unique identifier of the manufacturer.
manufacturer-name	String	RO	M	• Provided by *tapi-server* The formal name of the manufacturer of the Equipment.

Table 32: Common-actual-properties object's attributes required for UC4b.

common-actual-properties	**/tapi-common:context/tapi-equipment:physical-context/device/equipment/actual-equipment/common-actual-properties**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
asset-instance-identifier	String	RO	M	• Provided by *tapi-server* This attribute represents the asset identifier of this instance from the manufacturer's perspective **Part-number**
is-powered	Boolean	RO	M	• Provided by *tapi-server* The state of the power being supplied to the equipment. Note that this attribute summarizes the power state.

				Full details on the actual power system would be provided from a number of PC instances representing the relevant parts of the Power function (e.g. different voltage supplies)
Manufacture-date	Date-and-time	RO	M	• Provided by tapi-server This attribute represents the date on which this instance is manufactured.
Serial-number	String	RO	M	• Provided by *tapi-server* This attribute represents the serial number of this instance
Temperature	Decimal64	RO	M	• Provided by *tapi-server* The measured temperature of the Equipment.

There are some fields that are not included in the device (/**tapi-common:context/tapi-equipment:physical-context/tapi-equipment:device**) and must be reported to obtain a complete report of the device. Those fields will be included in the **/tapi-common:context/tapi-equipment:physical-context/tapi-equipment:device/tapi-equipment:name** with their respective "value-name".

- The **access-port** is the reference with the logical model (tapi-topology). One of the parameters of access-port is "equipment-uuid".

  Important note: in case the connector-identification and/or pin-identification is not present for a given access-port the key of the connector-pin attribute MUST be the concatenation of empty strings for the missing values and equipment-uuid (according to RESTCONF RFC8040 Sec 3.5.3). Each key leaf value except the last one is followed by a comma character. E.g., for a given access-port's connector-pin entry , the resource URI should be:

  …/tapi-equipment:access-port={uuid}/connector-pin=",,{equipment-uuid}"

- equipment-uuid: is the bridge between the logical model (NEPs etc.) and the PhysicalModel

Table 33: Device object attributes required for UC4b.

device	**/tapi-common:context/tapi-equipment:physical-context/device**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
Name	List of {value-name: value} "value-name": "NE_NAME" "value": " [0-9a-zA-Z_]{64}"	RO	M	Provided by tapi-server o tapi-client during NE creation
NE_ID	name": [{"value-name": "NE_ID", "value": {NE_ID}" }]	RO	M	• Provided by *tapi-server*
GATEWAY	"name": [{"value-name": "GATEWAY", "value": {Name_Gateway_Device}" }]	RO	M	• Provided by *tapi-server*
NE Type	"name": [{"value-name": " NE_TYPE", "value": {Name_NE_type}" }]	RO	M	Provided by *tapi-server*
IP	"name": [{"value_name": "IP", "value": {IP_Device}" }]	RO	M	• Provided by *tapi-server*

Mask	"name": [{"value_name": "MASK", "value": {Mask_Device}" }]	RO	M	• Provided by *tapi-server*
creation_time	"name": [{"value_name": " CREATION_TIME", "value": { Creation_time _Device}" }]	RO	M	• Provided by *tapi-server*
uuid	"[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}"	RO	M	• As per RFC 4122 • Provided by ***tapi-client***
access-port	List of {uuid, connector-pin, name} • uuid {"[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}" • connector-pin. List of {connector-identification, pin-identification, euqipmnet-uuid} • equipment-uuid {"[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}" • "name": [{"value_name": PORT_NUMBER", "value": {" [0-9a-zA-Z_]{64}"}]	RO	M	• **connector-pin**: The list of Pins that support the AccessPort • **equipment-uuid:** is the bridge between the logical model (NEPs etc.) and the PhysicalModel.

6.4.2.2 Relative location of component with TAPI 2.1.3 using holder location

The following picture shows the relative position of each “equipment” (chassis, slot, sublot, port) in a graphical representation.

The relation between TAPI naming and the picture is the following:

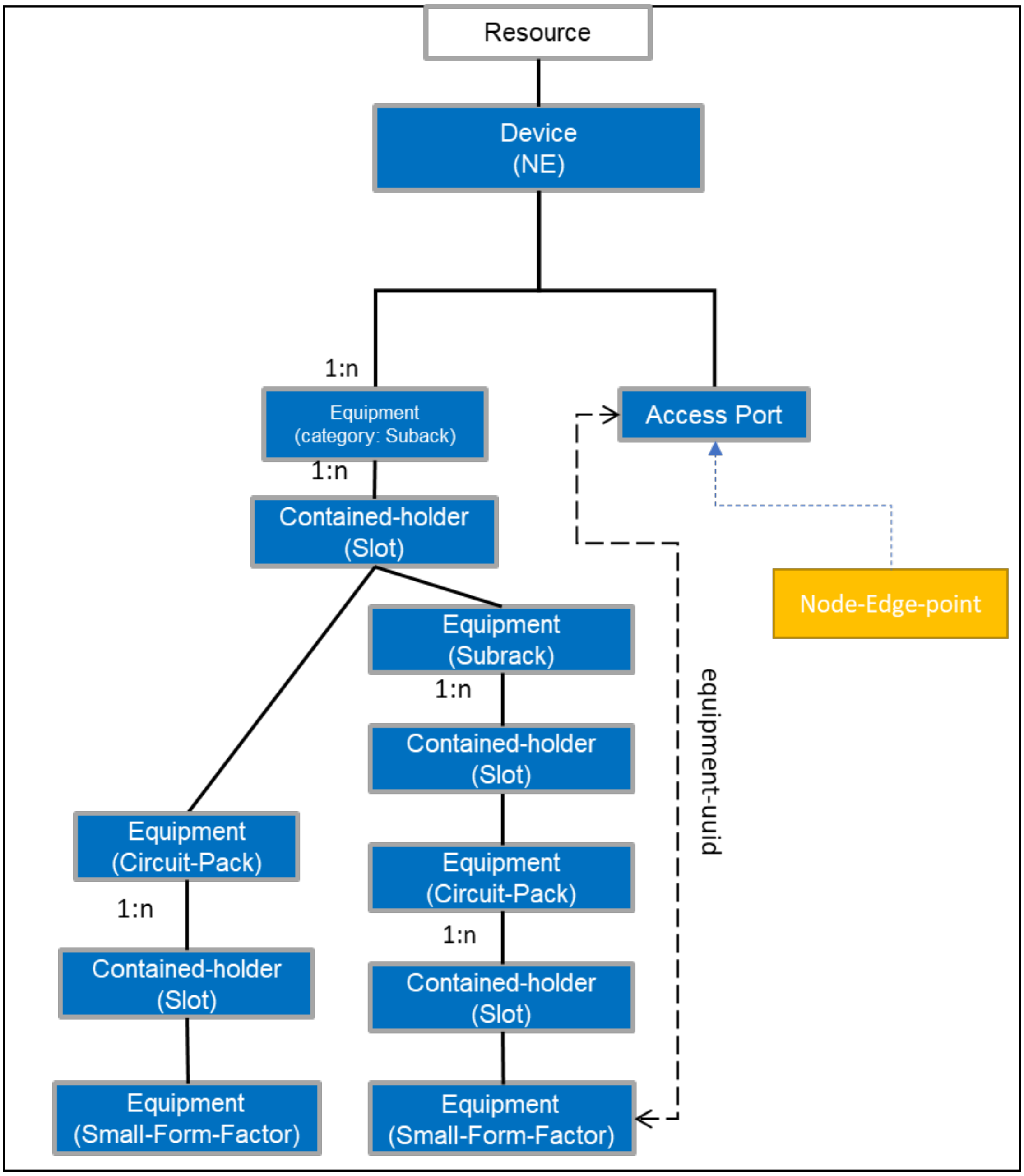


Figure 6-24 UC-4b Hierarchical arrangement of equipment objects with TAPI 2.1.3.

- Chassis=SUBRACK
- Card in slot= CIRCUIT_PACK/ SUBRACK
- Port in circuit pack= SMALL_FORMFACTOR_PLUGGABLE

The TAPI Server MUST use the ***tapi-equipment:contained-holder/actual-holder/common-holder-properties/holder-location*** to represent the **relative position of the contained-holders within the SUBRACK** equipment. The format of the holder-location string MUST be: "*SlotPosition*"-"*SubSlotPosition*" For convention, **if there is not sub-slot within a slot, the sub-slot value must be 0.**

There are some considerations needed to be taken to define a rule convention for filling this attribute. Three different scenarios are considered:

a. **Division**: The equipment slot structure is fixed, there is only one level of Holder objects, which may represent both "full slot" space or "half-sized slot" space cases. In other words, the Holder always represents the smallest granularity occupancy model. In this case, the ***holder-location*** MUST be: "*SlotPosition"-"0"*

b. **Hierarchy**: If the equipment slot structure can change dynamically (i.e., by software configuration of the SUBRACK equipment), an additional dimension of holder-location (i.e., a "sub-slot") must be introduced. In order to represent this sub-slot dimension, the list of ***tapi-equipment:contained-holder*** objects shall be dynamically increased with the new elements representing the partitioning. In this case, the ***holder-location*** MUST be: *"SlotPosition"-"SubSlotPosition"*.

c. **Specific Hardware (HW)**: In this case, a specific hardware is necessary to implement "sub-slotting". In this case, the existing Holder object will host an Equipment object (which MUST be a SUBRACK category equipment object) which at the time it is plugged-in, it enables the sub-slotting capability of the parent hardware. Then, the parent SUBRACK equipment holder-location arrangement shall follow one of the previous two models (depending if specific HW enabling sub-slotting is plugged or not). Please note, this extra-HW equipment is considered not implementing any control logic but just 'enables' the subslots space within the parent SUBRACK.

Then, according to the previous definition, the ***container-location*** **string represents the relative location of the container holder within an equipment**.

The following examples shows all the different possibilities and how to model them.

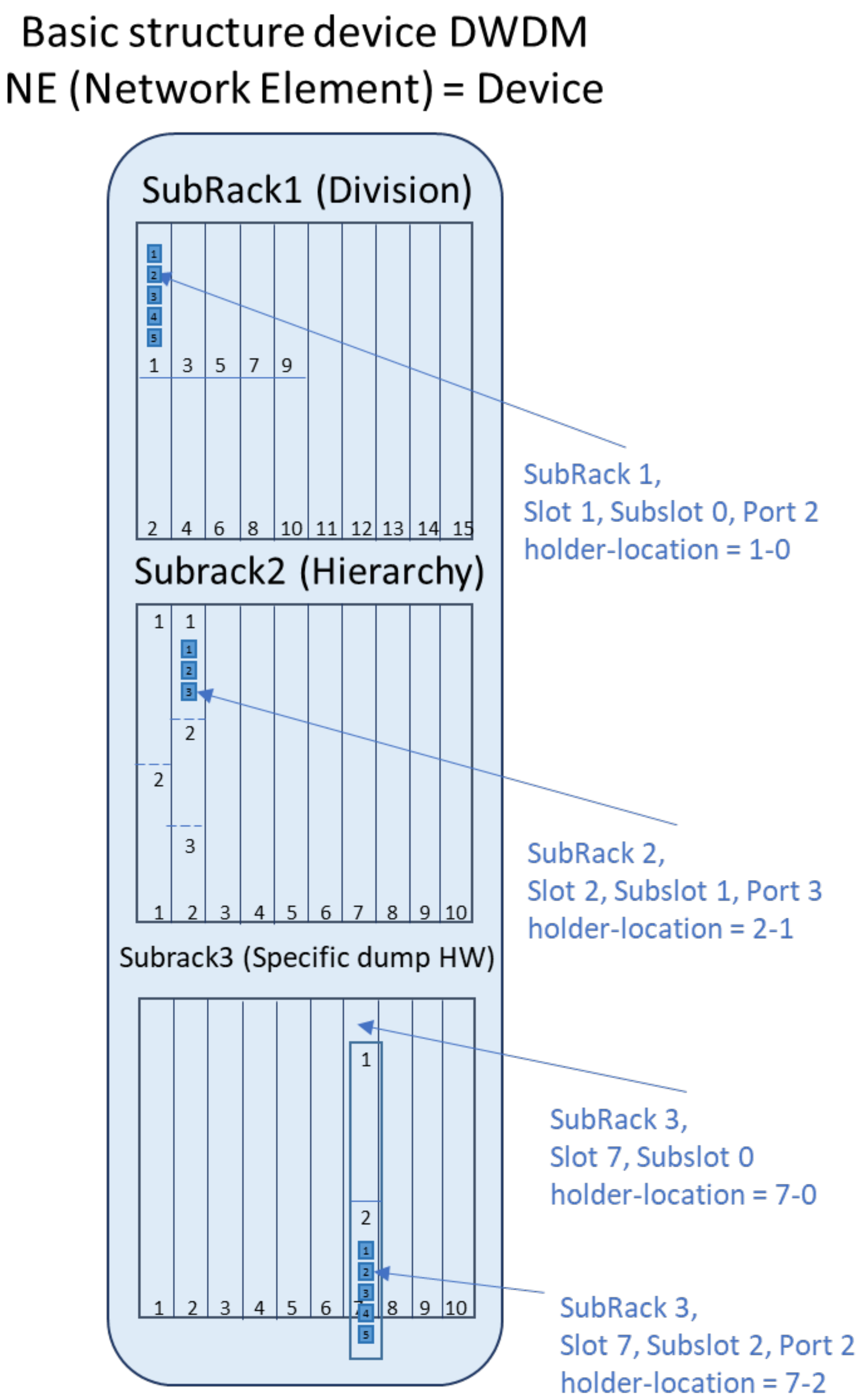


Figure 6-25 UC-4b Network Element Subracks container-holder location examples.

To complete the picture, the examples illustrated in Figure 6-25 are developed in TAPI model, including the holder-location value and the mapping to the INVENTORY_ID format presented in UC4a. Please note that the INVENTORY_ID will represent the absolute location of each equipment component, so it is derived from the position of the equipment within the tree.

Example Subrack1

Linecard holder-location in Subrack1

```
tapi-equipment:equipment[category=SUBRACK]/contained-holder/actual-holder/
    "holder-location": "1-0"
tapi-equipment:equipment[category=SUBRACK]/contained-holder/
    "name": "/ne=MadridNorte/r=1/sh=1/sl=1/s_sl=0"}]
```

Port2 holder-location in Linecard

```
tapi-equipment:equipment[category=CIRCUIT_PACK]/contained-holder/actual-holder/
    "holder-location": "2-0"
tapi-equipment:equipment[category=CIRCUIT_PACK]/contained-holder/
    "name": "/ne=MadridNorte/r=1/sh=1/sl=1/s_sl=0/p=2"}]
```

Example Subrack2

Linecard holder-location in Subrack2

```
tapi-equipment:equipment[category=SUBRACK]/contained-holder/actual-holder/
    "holder-location": "2-1"
tapi-equipment:equipment[category=SUBRACK]/contained-holder/
    "name": "/ne=MadridNorte/r=1/sh=2/sl=2/s_sl=1"}]
```

Port holder-location in Linecard

```
tapi-equipment:equipment[category=CIRCUIT_PACK]/contained-holder/actual-holder/
    "holder-location": "3-0"
tapi-equipment:equipment[category=CIRCUIT_PACK]/contained-holder/
    "name": [{"value_name": "INVENTORY_ID",
             "value": "/ne=MadridNorte/r=1/sh=2/sl=2/s_sl=1/p=3"}]
```

Example Subrack3

Extra HW SUBRACK holder-location in Subrack3

```
tapi-equipment:equipment[category=SUBRACK]/contained-holder/actual-holder/
    "holder-location": "7-0"
tapi-equipment:equipment[category=SUBRACK]/contained-holder/
    "name": [{"value_name": "INVENTORY_ID",
              "value": "/ne=MadridNorte/r=1/sh=3/sl=7/s_sl=0"}]
```

Linecard holder-location in Subrack3

```
tapi-equipment:equipment[category=SUBRACK]/contained-holder/actual-holder/
    "holder-location": "7-2"
tapi-equipment:equipment[category=SUBRACK]/contained-holder/
    "name": [{"value_name": "INVENTORY_ID",
              "value": /ne=MadridNorte/r=1/sh=3/sl=7/s_sl=2"}]
```

Port holder-location in Linecard

```
tapi-equipment:equipment[category=CIRCUIT_PACK]/contained-holder/actual-holder/
    "holder-location":"2-0"
tapi-equipment:equipment[category=CIRCUIT_PACK]/contained-holder/
    "name":[{"value_name": "INVENTORY_ID",
             "value": "/ne=MadridNorte/r=1/sh=3/sl=7/s_sl=2/p=2"}]
```

Some examples of INVENTORY_ID for the node-edge-points potentially mapped to the ports described in the previous examples:

```
Example 1:
"name": [{"value_name": "INVENTORY_ID", "value":
"/ne=MadridNorte/r=1/sh=1/sl=1/s_sl=0"}]

Example 2:
"name": [{"value_name": "INVENTORY_ID", "value":
"/ne=MadridNorte/r=1/sh=2/sl=2/s_sl=1/p=3"}]

Example 3:
"name": [{"value_name": "INVENTORY_ID", "value":
"/ne=MadridNorte/r=1/sh=3/sl=7/s_sl=2/p=2"}]
```

6.5 Resiliency

6.5.1 Use case 5a: 1+1 OLP OMS/OTS Protection with Diverse Service Provisioning

Number	**UC5a**
Name	**1+1 OLP Protection with Diverse Service Provisioning**
Technologies involved	Optical
Process/Areas Involved	Planning and Operations
Brief description	This use case covers the use of OLPs elements for protection services at OMS/OTS layers. Three possible protection schemas were identified, represented in Figure 6-26.

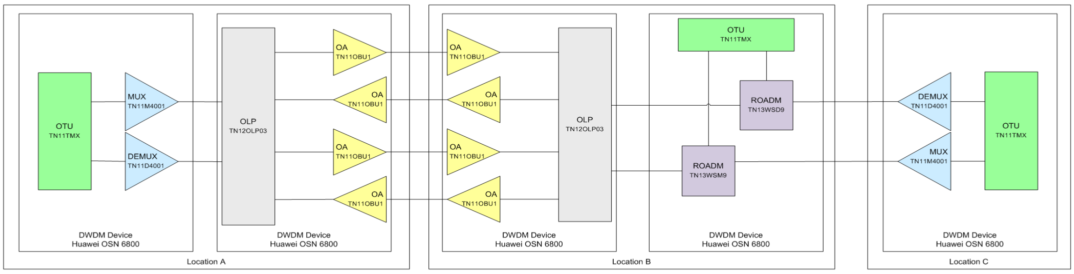


Figure 6-26 UC-5c OMS OLP protection schemas.

OMS/OTS OLP protection is not intended to be configured by the user, but to be represented by the TAPI server as part of the PHOTONIC_MEDIA layer topology.

In this case, the OMS protection is not provisioned by a connectivity-service send by the TAPI client. Instead, the TAPI server is responsible of the automatic discovery of the OMS protection scheme and its representation through the TAPI interface.

An OMS protection MUST be represented as described in [TAPI-TOP-MODEL-REQ-23]. The Link object representing the OMS protected resource (Link) MUST contain the ***/tapi-topology:link/tapi-topology:resilience-type/protection-type*** attribute to specify which type of protection service is provided. Depending on the type of protection this attribute MAY be set with the following values:

- **ONE_PLUS_ONE_PROTECTION:** Double selective receiving, dual transmitting and selective receiving. The optical power from Tx are split with a ratio of 50:50 on the main route and the standby route, which means both the main and standby routes are in use no matter whether there is a fault in the main route.
- **ONE_FOR_ONE_PROTECTION:** Selective receiving election, selective transmitting and selective receiving mode. There are a main route and a standby route between the two sites.

The OMS protected link resource MUST be realized by a OMS OLP connection which MUST be reported within the **/tapi-connectivity:connectivity-context/connection** list, and which is not related to any user-defined connectivity-service.

The TAPI server is responsible of maintaining the SLA condition and triggers the automatic protection/switching actions when the OMS OLP connection nominal path is being affected by any outage condition which provoke a degradation or interruption of the service.

The protection process MUST be triggered automatically by the TAPI server, however, the TAPI client MUST be notified about the service condition changes through the tapi-notification service (as defined in UCs 15a and 15b).

The SDN-C must provide asynchronous notifications of the events generated by the network failure which triggers the restoration including, but not limited to, topology changes, service status changes, triggered alarms…

Layers involved	PHOTONIC_MEDIA
Type	Resilience
Description & Workflow	This type of OLP protection (OMS/OTS OLP) is realized by the tapi-server. Thus, there is not a client-to-server message exchange to be reported in this section.

6.5.1.1 Expected result

The expected representation of the OMS OLP protection schema is represented over the TAPI topology scenario included in Section 4.3.

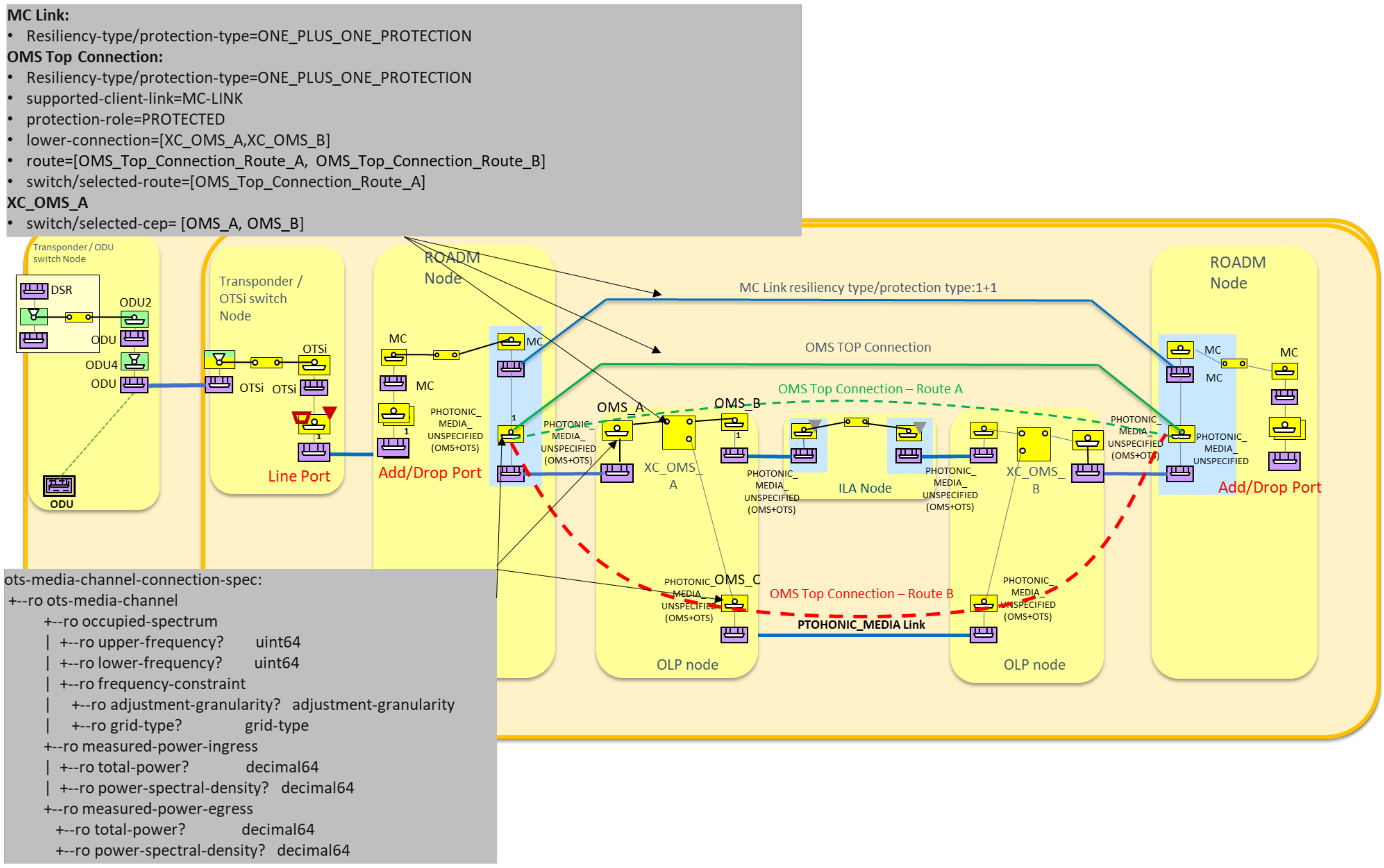
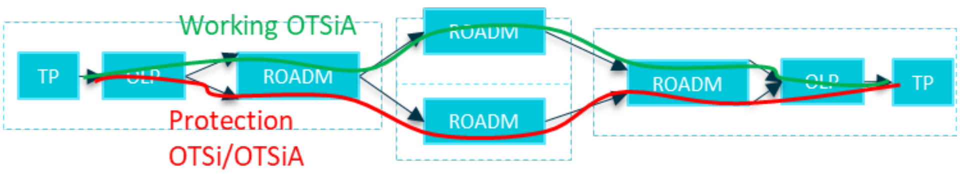


Figure 6-27 UC-5a OLP protection TAPI representation.

<u>Note</u>: Transponder representation in the Z side has been omitted due to the symmetry of the scenario.

<u>Note 2</u>**:** The Top-Connection switch control is under discussion thus the selected-route attribute shall be considered optional when implementing OLP protection UCs. In contrast, the switch-control at the XC level MUST be present, representing/reflecting the actual switch state.

6.5.2 Use case 5b: 1+1 OLP Line Protection with Diverse Service Provisioning

Number	**UC5b**
Name	**1+1 OLP Protection with Diverse Service Provisioning**
Technologies involved	Optical
Process/Areas Involved	Planning and Operations
Brief description	This use case covers the use of OLPs elements for 1+1 and 1:1 protection service at different layers OTSi/OTSiA. This use case does not allow intermediate regeneration in any of the MC trails, this capability is left for future use case specification.  **Figure 6-28 UC-5b Line OLP protection schema.** **Line Side OLP** protection represents a protection schema where an OLP is placed between the Transponder line ports and two add/drop ports of a ROADM. The OLP components duplicates the incoming signal and implements an Automatic Protection Switch (APS) which switch the traffic between the two Add/Drop ports upon loss of signal conditions (power measurement decrease under a target threshold) in less than 50ms. The TAPI server is responsible of maintaining the SLA condition and triggers the automatic protection/switching actions when the connectivity-service nominal path is being affected by any outage condition which provoke a degradation or interruption of the service. The protection process MUST be triggered automatically by the TAPI server, however, the TAPI client MUST be notified about the service condition changes through the tapi-notification service (as defined in UCs 15a and 15b). The SDN-C must provide asynchronous notifications of the events generated by the network failure which triggers the restoration including, but not limited to, topology changes, service status changes, triggered alarms.
Layers involved	PHOTONIC_MEDIA
Type	Resilience
Description & Workflow	This type of OLP protection (**Line Side OLP**) requires the reservation of two disjoint routes along the PHOTONIC_MEDIA layer for the provisioning of connections to support the 1+1 or 1:1 Connectivity Services. The TAPI Client SHOULD select two valid SIPs, to be referenced by the CSEPs, which reference two PHOTONIC_LAYER_QUALIFER_OTSI NEPs directly connected to the Line-Side OLP elements, as described in the Connectivity Service example (Figure 6-30). The TAPI Client MAY include the three SIPs mapped to the MC/Photonic_Media NEPs representing the switching ports of the OLP components (MC1, MC2, MC3 as represented in the Figure), to associated to them different Protection Roles within the CSEPs definition (please note that zEnd peer SIPs, NEPs have been omitted in this explanation for simplicity). These protection roles MAY be in the example PROTECTED (71.D1 - MC1), WORK (72.D1 - MC2), PROTECT (73.D1 - MC3). Alternatively, the TAPI Client MAY delegate the protection role selection to the TAPI Server during the CS provisioning process and thus, only the two OTSi/Photonic_Media SIPs MUST be referenced within the CS request. In case, the protection role is delegated, the TAPI Client MUST include the **preferred-restoration-layer** attribute in the CS's request.

The Connectivity Service object sent to the TAPI Server MUST include the ***tapi-connectivity:connectivity-service/tapi-topology:resilience-type/protection-type*** attribute to specify which type of protection service is requested. Depending on the type of protection this attribute MAY be set with the following values:

- **ONE_PLUS_ONE_PROTECTION:** Double selective receiving, dual transmitting and selective receiving. The optical power from Tx are split with a ratio of 50:50 on the main route and the standby route, which means both the main and standby routes are in use no matter whether there is a fault in the main route.
- **ONE_FOR_ONE_PROTECTION:** Selective receiving election, selective transmitting and selective receiving mode. There are a main route and a standby route between the two sites.

This UC is implemented following the same workflow described in "Description & Workflow" of UC1a described in section 6.2.1.

Resiliency workflow:

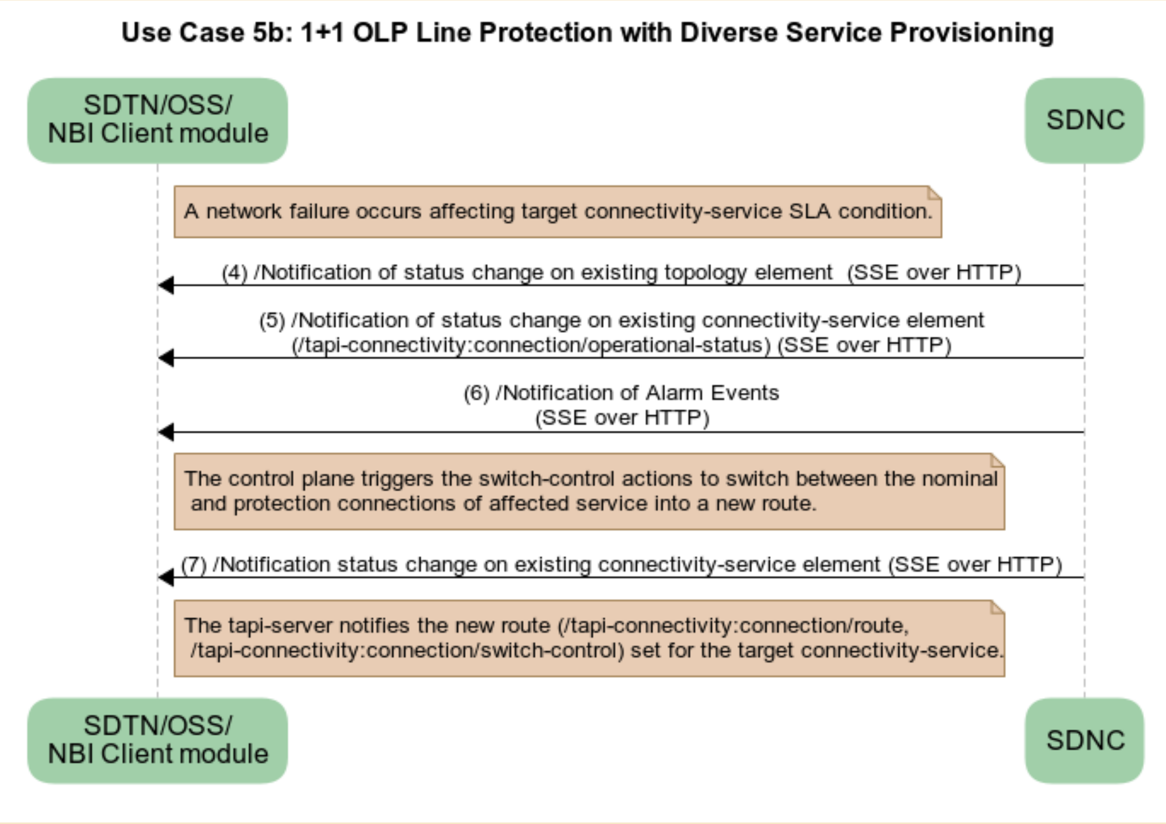


Figure 6-29 UC-5b: Resiliency workflow.

6.5.2.1 Expected results

As described in UC1f (Section 6.2.6) the OLS PHOTONIC_MEDIA layer may be modelled unidirectionally or bidirectionally. In UC1f three possible models were presented:

1. Full-bidirectional - UNI and PHOTONIC_MEDIA model.
2. ~~Mixed-scenario - UNI bidirectional and topology unidirectional.~~
3. Full-unidirectional OLS scenario - UNI and PHOTONIC_MEDIA unidirectional

For this use case, the mixed approach is not allowed. The other two models are described in the following subsections.

6.5.2.1.1 UC5b Example 1: Bidirectional OLP Line protection modelling

The expected result after the creation of the OLP protected PHOTONIC_LAYER_QUALIFER_OTSI connectivity service is represented over the TAPI topology scenario included in Figure 6-30.

Please note, OTSiMC layer is intentionally left out from the diagram for simplicity but it MAY be included on top of every MC CEP described in the example.

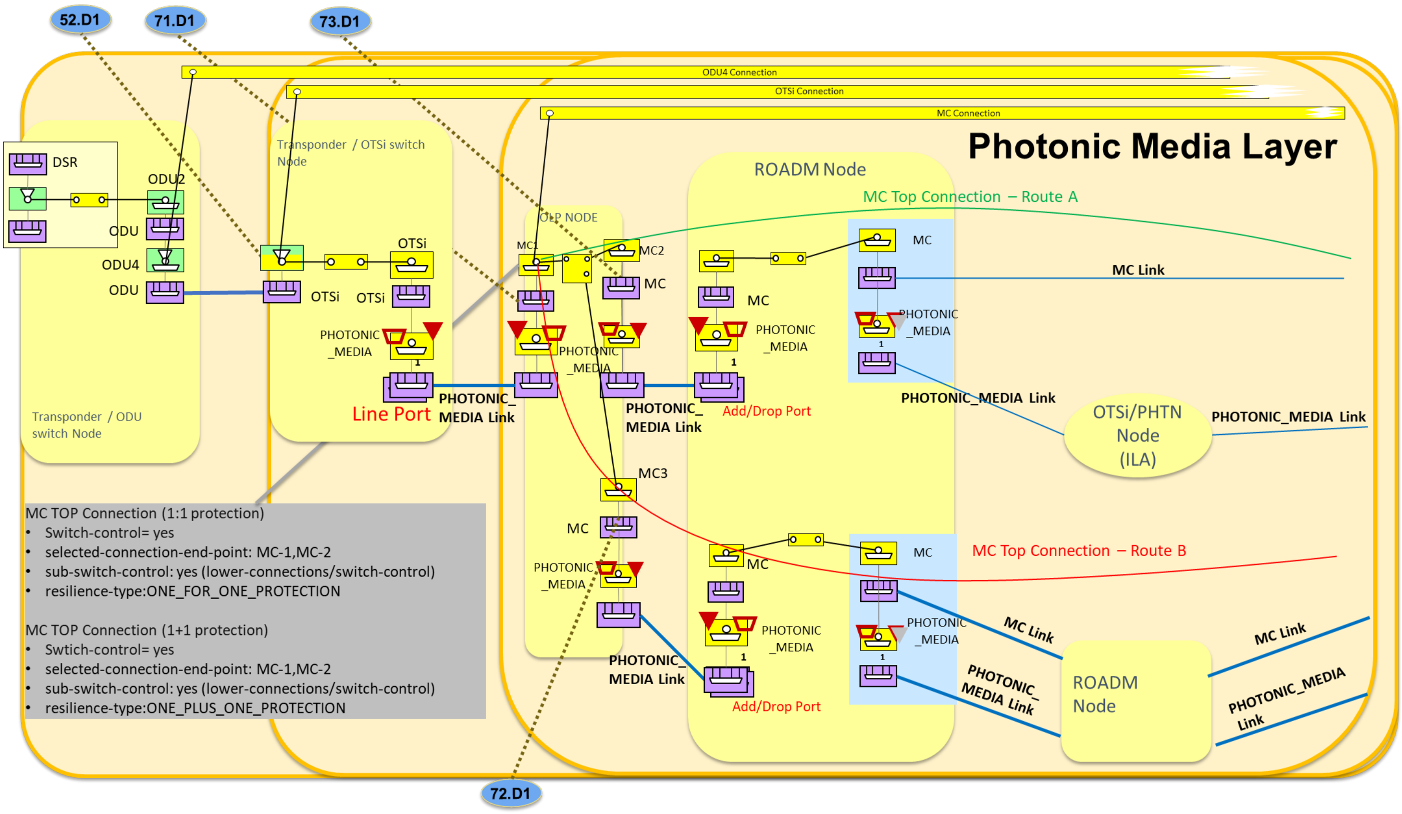


Figure 6-30 UC-5b OLP TAPI Connectivity-Service low-level description.

Once the CS is created, the TAPI Server is responsible of implementing the Switch control among the connections generated to support the different protection schemas (1+1, 1:1), which MUST provide automatic switch control between the working and protection connections in less than 50ms.

The requested PHOTONIC_LAYER_QUALIFER_OTSI CS triggers the creation of:

- **Protected OTSi Top Connection**.
- **Protected MC Top Connection**: which has two routes and implements switch-control and sub-switch-control based on its aEnd and zEnd lower-connections switch-control.
    - **aEnd OLP site MC lower connection**: implementing the switch-control between aEnd working and protection CEPs (MC-2, MC-3).

- **N intermediate MC lower connections**: without switch control (connections along the ROADM nodes)
- **zEnd OLP site MC lower connection**: implementing the switch-control between aEnd working and protection CEPs (respective CEPs to MC-2, MC-3 in the other zEnd OLP).

In case the service is 1:1 PROTECTION:

- **selected-connection-end-points:** MC-1, MC-2 (and the respective CEPs in zEnd) switch between MC-2 and MC-3 when the conditions changes.
- **selected-route:** Route-A, and switch to Route-B, when the conditions changes. **The Top-Connection switch control is under discussion thus the selected-route attribute shall be considered optional when implementing OLP protection UCs.**

In case the service is 1+1 PROTECTION:

- **selected-connection-end-points:** MC-1, MC-2 (and the respective CEPs in zEnd) initially when both routes are active, and switch between MC-2 and MC-3 when the conditions changes.
- **selected-route:** Route-A, Route-B initially when both routes are active, and switch to one of them, when the conditions changes. **The Top-Connection switch control is under discussion thus the selected-route attribute shall be considered optional when implementing OLP protection UCs.**

For both cases:

- **sub-switch-control:** Referencing aEnd, zEnd OLP site MC lower connection switch-control objects.

6.5.2.1.2 UC5b Example 2: Unidirectional OLP Line protection modelling

In this case the connectivity model is split in the two directions (Add/Drop). Each direction includes a MC Top Connections which MUST be included within the related CS connection list.

The CS request in this case MUST include four CSEPs referencing the Add/Drop interfaces SIPs at aEnd and zEnd points.

The actual guidelines for the switch control described for the bidirectional model applies to each unidirectional Top Connection. To all extend, the two unidirectional Top Connections MUST expose the management of its switch-control the same one as it was described in the bidirectional model. Thus, the switch control at each direction is independent and but the bidirectional connectivity-service operational-state is impacted by the status of both.

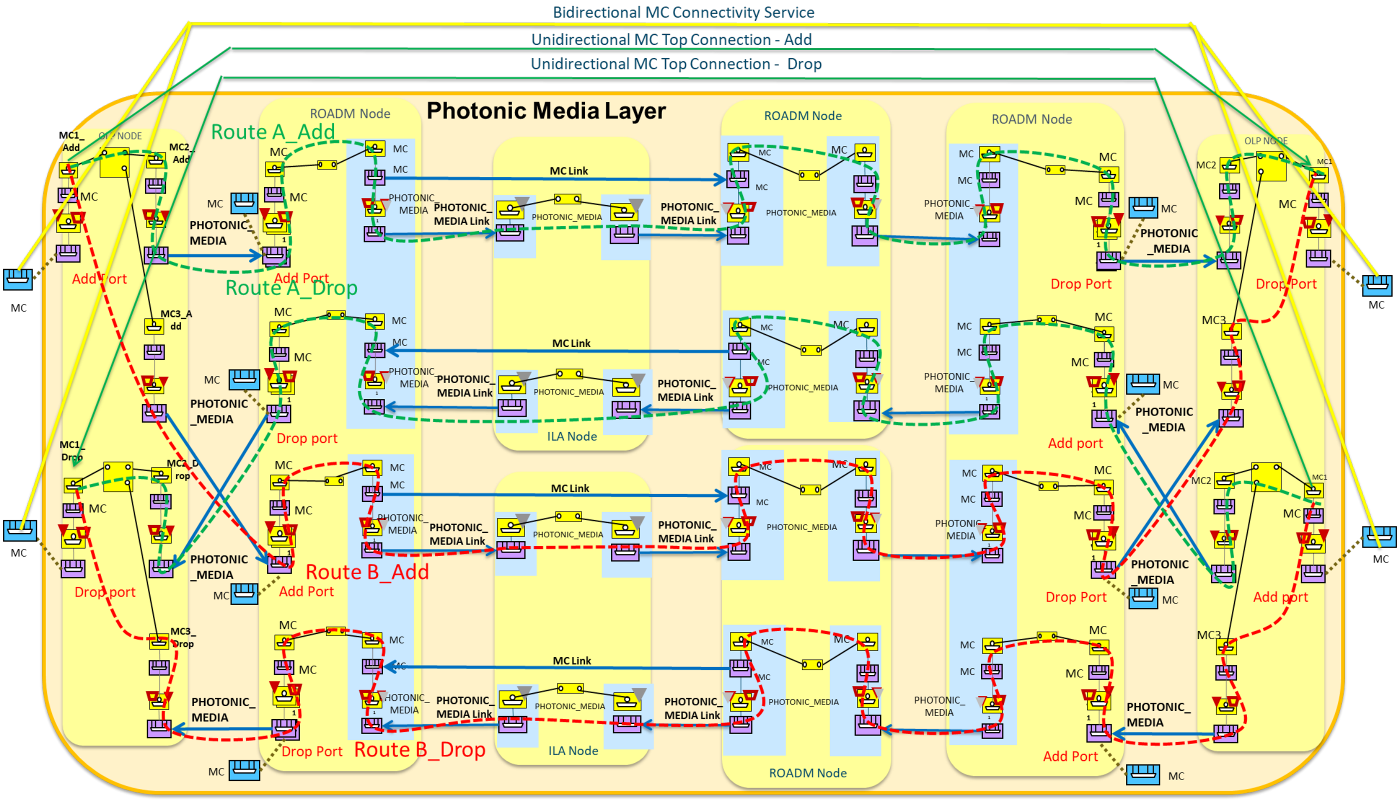


Figure 6-31 UC-5b OLP TAPI Connectivity-Service low-level description. Unidirectional OLS modelling.

6.5.2.2 Required parameters

Table 34, Table 35, Table 36, Table 37 and Table 38 complements the information included in the Use Case 1a definition, with the Connectivity-Service, Connectivity-Service-End-Points, Connections and Switch-control, attributes required to implement this use case, this implies all attributes and objects included in Section 6.2.1.1 tables is required for this use case too.

Table 34: Connectivity-service attributes for 1+1 UC5b.

connectivity-service	**/tapi-common:context/tapi-connectivity:connectivity-context/connectivity-service**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
resilience-type	{"**protection-type**": [ONE_FOR_ONE_PROTECTION, ONE_PLUS_ONE_PROTECTION] }	RW	M	• Provided by *tapi-client*
preferred-restoration-layer	[PHOTONIC_MEDIA]	RW	M	• Provided by *tapi-client*
reversion-mode	["REVERTIVE", "NON-REVERTIVE"]	RW	M	• Provided by tapi-client
restore-priority	"[0-9]+"	RW	O	• Provided by tapi-client

hold-off-time	"[0-9]{4}"	RW	O	• Provided by tapi-client
wait-to-revert-time	"[0-9]{4}"	RW	O	• Provided by tapi-client
max-switch-times	"[0-9]{2}"	RW	O	• Provided by tapi-client
is-coordinated-switching-both-ends	[true, false]	RW	O	• Provided by tapi-client
is-lock-out	[true, false]	RW	O	• Provided by tapi-client
is-frozen	[true, false]	RW	O	• Provided by tapi-client

Table 35: Connectivity-service-End-Points attributes for UC5b.

connectivity-service-end-point	**/tapi-common:context/tapi-connectivity:connectivity-context/connectivity-service/end-point**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
protection-role	["WORK", "PROTECT", "PROTECTED"]	RW	M	• Provided by *tapi-client* • Support only P2P

Table 36: Connection attributes for UC5b.

connection	**/tapi-common:context/tapi-connectivity:connectivity-context/connection**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
switch-control	List of { **switch-control** }	RO	M	• Provided by *tapi-server*

Table 37: Switch-control attributes for UC5b.

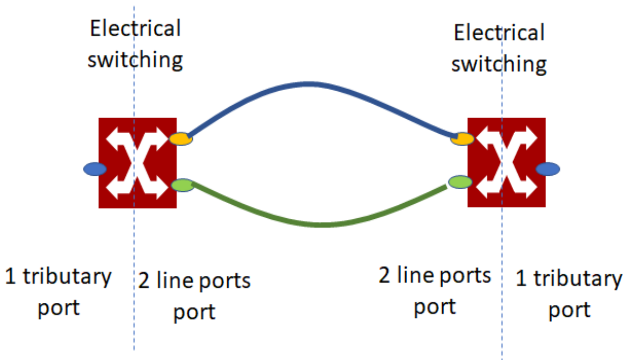
switch-control	**/tapi-common:context/tapi-connectivity:connectivity-context/connection/switch-control**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
uuid	"[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}"	RO	M	• As per RFC 4122 • Provided by ***tapi-server***
name	List of {value-name: value} • "value-name": "SWC_NAME" "value": " [0-9a-zA-Z]{64}"	RO	M	• Provided by ***tapi-server***
resilience-type	{"**protection-type**": [ONE_PLUS_ONE_PROTECTION" , "ONE_FOR_ONE_PROTECTION"] }	RO	O	• Provided by ***tapi- server***
reversion-mode	["REVERTIVE", "NON-REVERTIVE"]	RO	O	• Provided by ***tapi- server***
restore-priority	"[0-9]+"	RO	O	• Provided by ***tapi- server***
hold-off-time	"[0-9]{4}"	RO	O	• Provided by ***tapi- server***

switch-control	**/tapi-common:context/tapi-connectivity:connectivity-context/connection/switch-control**			
wait-to-revert-time	"[0-9]{4}"	RO	O	• Provided by ***tapi- server***
max-switch-times	"[0-9]{2}"	RO	O	• Provided by ***tapi- server***
is-coordinated-switching-both-ends	[true, false]	RO	O	• Provided by ***tapi- server***
is-lock-out	[true, false]	RO	O	• Provided by ***tapi- server***
is-frozen	[true, false]	RO	O	• Provided by ***tapi- server***
preferred-restoration-layer	List of [“PHOTONIC_MEDIA”]	RO	M	• Provided by **tapi- server**
sub-switch-control	List of {"*/config/context/connection/{uuid}/switch-control/{switch_control_uuid}/*"}	RO	M	• Provided by ***tapi-server***
switch	List of { **switch** }	RO	M	• Provided by ***tapi-server***

Table 38: Switch attributes for UC5b.

switch	**/tapi-common:context/tapi-connectivity:connectivity-context/connection/switch-control/switch**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
local-id	"[0-9a-zA-Z_]{32}"	RO	M	• Provided by ***tapi-server***
name	List of {value-name: value} • "value-name": "SW_NAME" "value": " [0-9a-zA-Z_]{64}"	RO	M	• Provided by ***tapi-server***
switch-direction	["BIDIRECTIONAL", "INPUT", "OUTPUT"]	RO	M	• Provided by ***tapi-server***
selection-control	["LOCK_OUT", "NORMAL", "MANUAL", "FORCED"]	RO	M	• Provided by ***tapi-server***
selection-reason	["LOCKOUT", "NORMAL", "MANUAL", "FORCED", "WAIT_TO_REVERT", "SIGNAL_DEGRADE", "SIGNAL_FAIL"]	RO	M	• Provided by ***tapi-server***
selected-connection-end-point	List of {"**connection-end-point-ref** - */tapi-common:context/tapi-topology:topology-context/topology/node/owned-node-edge-point/tapi-connectivity:cep-list/connection-end-point/uuid* "}	RO	M	• Provided by ***tapi-server***
selected-route	List of {"*/tapi-common:context/tapi-connectivity:connectivity-context/connection/{uuid}/route/{local_id}/* "}	RO	O	• Provided by ***tapi-server***

6.5.3 Use case 5c: 1+1 protection with Diverse Service Provisioning (eSNCP)

Number	**UC5c**
Name	1+1 protection with Diverse Service Provisioning (eSNCP)
Technologies involved	Optical
Process/Areas Involved	Planning and Operations
Brief description	This use case covers the provisioning of 1+1 protected services implemented through the electrical subnetwork connection protection (eSNCP) schema (Figure 6-32) also known as ODUk SNCP. ODUk SNCP protection protects services on line boards and on the units physically located behind the line boards. Cross-connections on the electrical layer are used to implement dual feeding and selective receiving, and protection switching is triggered by OTN overhead alarms. The cross-connection granularity is ODUk (k = 0, 1, 2, 3, 4, flex).  **Figure 6-32 UC-5c eSNCP protection schema.** The actual implementation of the ODUk SNCP monitoring mode is out of the scope of this document but it is assumed that at least the Path Monitoring (PM) OTN mechanism MUST be supported to provide end-to-end ODU protection switching for DSR client services (e.g., multi-domain ODU services are out of the scope of this use case definition). The TAPI server is responsible of maintaining the SLA condition and triggers the automatic protection/switching actions when the connectivity-service nominal path is being affected by any outage condition which provoke a degradation or interruption of the service. The protection process MUST be triggered automatically by the TAPI server, however, the TAPI client MUST be notified about the service condition changes through the **tapi-notification** service (as defined in UCs 15a and 15b). The SDN-C must provide asynchronous notifications of the events generated by the network failure which triggers the restoration including, but not limited to, topology changes, service status changes, triggered alarms…
Layers involved	ODU
Type	Resilience
Description & Workflow	The connectivity-service is requested between two DSR/ODU connectivity service endpoints and requires the reservation of two disjoint routes at the ODU layer between transponder's line interfaces. The TAPI Client SHOULD select two valid SIPs representing the client layer interfaces (which can be potentially at the DSR layer or at any LO-ODUj layer rate), to be referenced by the

ConnectivityServiceEndPoints, which reference two NEPs which are intended to be connected by the 1+1 connectivity-service at the ODU layer (Figure 6-33).

The TAPI Client MAY explicitly state the optical line interfaces to use to implement the protection at the HO-ODUk layer by including the two pairs of SIPs (67.D1, 68.D1) mapped to the OTSi NEPs representing the line ports of the transponders selected to provide the two disjoint routes for the HO-ODUk protected Top Connection. The TAPI Client MAY associate to them different Protection Roles within the CSEPs definition.

Alternatively, the TAPI Client MAY delegate the protection role selection to the TAPI Server during the CS provisioning process and thus, only specify the two client SIPs (potentially at the DSR or any LO-ODUj layer rate) MUST be referenced within the CS request. In case, the protection role is delegated, the TAPI Client MUST include the **preferred-restoration-layer** attribute in the CS's request.

The Connectivity Service object sent to the TAPI Server MUST include the ***tapi-connectivity:connectivity-service/tapi-topology:resilience-type/protection-type*** attribute with **ONE_PLUS_ONE_PROTECTION** attribute value.

This UC is implemented following the same workflow described in "Description & Workflow" of UC1a described in section 6.2.1

Resiliency workflow:

This UC is implemented following the same workflow described in "Description & Workflow" of UC5b described in section 6.5.2

6.5.3.1 Expected result

The expected result after the creation of the eSNCP ODU Connectivity Service is represented over the TAPI topology scenario included in Figure 6-33.

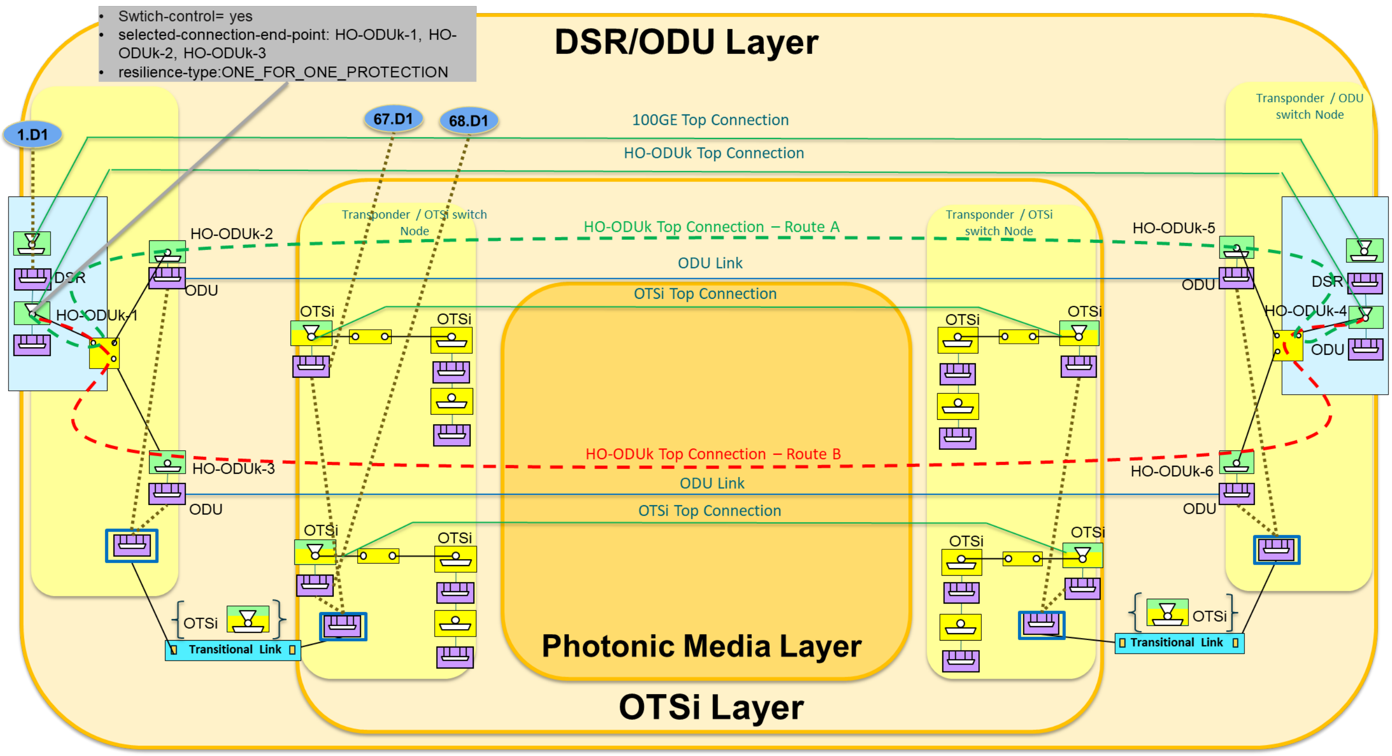


Figure 6-33 UC5c: eSNCP protection schema for HO-ODUk Top Connection.

Once the CS is created, the TAPI Server is responsible of implementing the Switch control among the connections generated to support the ODU eSNCP protection schema, which MUST provide automatic switch control between the working and protection connections in less than 50ms. Please refer to the graphical illustration of this use case included Figure 6-33, which includes the Connection generated after the CS is created.

The requested ODU CS triggers the creation of:

- **Protected DSR Top Connection.**
- **Protected HO-ODUk Top Connection:** which has two routes and implements switch-control and sub-switch-control based on its aEnd and zEnd lower-connections switch-control.
    - **aEnd HO-ODU lower connection**: implementing the switch-control between aEnd protected CEP (HO-ODUk-1) and the working and protection CEPs (HO-ODUk-2, LO-ODU-3).
    - **zEnd HO-ODU lower connection**: implementing the switch-control between zEnd protected CEP (HO-ODUk-4) and the working and protection CEPs (HO-ODUk-5, HO-ODUk-6).
    - **selected-connection-end-points:** HO-ODUk-1 and HO-ODUk-2 or HO-ODUk-3 depending on the actual switch-control. The selected-CEPs switch between HO-ODUk-2 and HO-ODUk-3 when the network conditions changes in order to maintain the service SLA.
    - **selected-route:** Route-A (HO-ODU-Top-Connection-A) or Route-B (HO-ODU-Top-Connection-B) according to the switching state of the lower XCs. **The Top-Connection switch control is under discussion thus the selected-route attribute shall be considered optional when implementing OLP protection UCs.**
    - **sub-switch-control:** Referencing aEnd, zEnd lower-connection switch-control objects.

Additionally, the service might generate the PHOTONIC_MEDIA Layer connections if ODU layer does not has enough resources:

- **One working OTSi-Top Connection-A.** Without switch control and its associated lower-connections.
- **One protection OTSi -Top Connection-B.** Without switch control and its associated lower-connections.
- **One working MC Top Connection-A.** Without switch control and its associated lower-connections.
- **One protection MC Top Connection-B.** Without switch control and its associated lower-connections.

6.5.3.2 Required parameters

Table 39 complements the information included in the Use Case 1a and Use Case 5b definitions, with the Connectivity-Service attributes required implementing this use case, this implies all attributes and objects included in Section 6.2.1.1 and 6.5.2.2 tables are required for this use case too.

Table 39: Connectivity-service attributes for UC5c.

connectivity-service	**/tapi-common:context/tapi-connectivity:connectivity-context/connectivity-service**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
resilience-type	{"**protection-type**": ONE_PLUS_ONE_PROTECTION] }	RW	M	• Provided by *tapi-client*

preferred-restoration-layer	[ODU]	RW	M	• Provided by *tapi-client*
reversion-mode	["REVERTIVE", "NON-REVERTIVE"]	RW	M	• Provided by tapi-client
restore-priority	"[0-9]+"	RW	O	• Provided by tapi-client
hold-off-time	"[0-9]{4}"	RW	O	• Provided by tapi-client
wait-to-revert-time	"[0-9]{4}"	RW	O	• Provided by tapi-client
max-switch-times	"[0-9]{2}"	RW	O	• Provided by tapi-client
is-coordinated-switching-both-ends	[true, false]	RW	O	• Provided by tapi-client
is-lock-out	[true, false]	RW	O	• Provided by tapi-client
is-frozen	[true, false]	RW	O	• Provided by tapi-client

6.5.4 Use case 6a: Dynamic restoration policy for unconstrained and constrained connectivity services.

Number	**UC6a**
Name	**Dynamic restoration policy for DSR/ODU/OTSI unconstrained connectivity services**
Technologies involved	Optical
Process/Areas Involved	Planning and Operations
Brief description	This use case covers the provisioning of connectivity-services with restoration capabilities. The dynamic restoration capability can be requested at different layers i.e., DSR, ODU, OTSI. The TAPI client just need to specify two available Connectivity Service End Points and introduce the service characteristics (including the restoration-type and protection-type attributes). The TAPI server is responsible of maintaining the SLA condition and triggers the dynamic restoration process when the connectivity-service nominal path is being affected by any outage condition which provoke a degradation or interruption of the service. The TAPI server is expected to calculate the new restoration path according to the current network conditions when the SLA failure occurs. The dynamic restoration process MUST be triggered automatically by the TAPI server, however, the TAPI client MUST be notified about the service condition changes through the tapi-notification service (as defined in UCs 15a and 15b). The SDN-C must provide asynchronous notifications of the events generated by the network failure which triggers the restoration including, but not limited to, topology changes, service status changes, triggered alarms. Additional constrains, such coroute-inclusion or diversity-exclusion SHALL be account as **loose constrains** at the time of the restoration occurs, i.e., applicable if possible.
Layers involved	ODU, PHOTONIC_MEDIA
Type	Resilience
Description & Workflow	The connectivity service is requested between two DSR/ODU/OTSI CSEPs. The TAPI Client SHOULD select two valid SIPs, to be referenced by the CSEPs, which reference two DSR/LO-ODUj/OTSI NEPs which are intended to be connected by the dynamic-restorable service. The TAPI Client MAY include the ***tapi-connectivity:preferred-restoration-layer*** object to specify the preferred restoration layer, but the final decision on how to implement the restoration is responsibility of the TAPI server based on the current network conditions when the dynamic restoration process occurs. The Connectivity Service object sent to the TAPI Server MUST include the ***tapi-connectivity:connectivity-service/tapi-topology:resilience-type/protection-type*** attribute with **DYNAMIC_RESTORATION** attribute value. This UC is implemented following the same workflow described in “Description & Workflow” of UC1a described in section 6.2.1. Resiliency workflow:

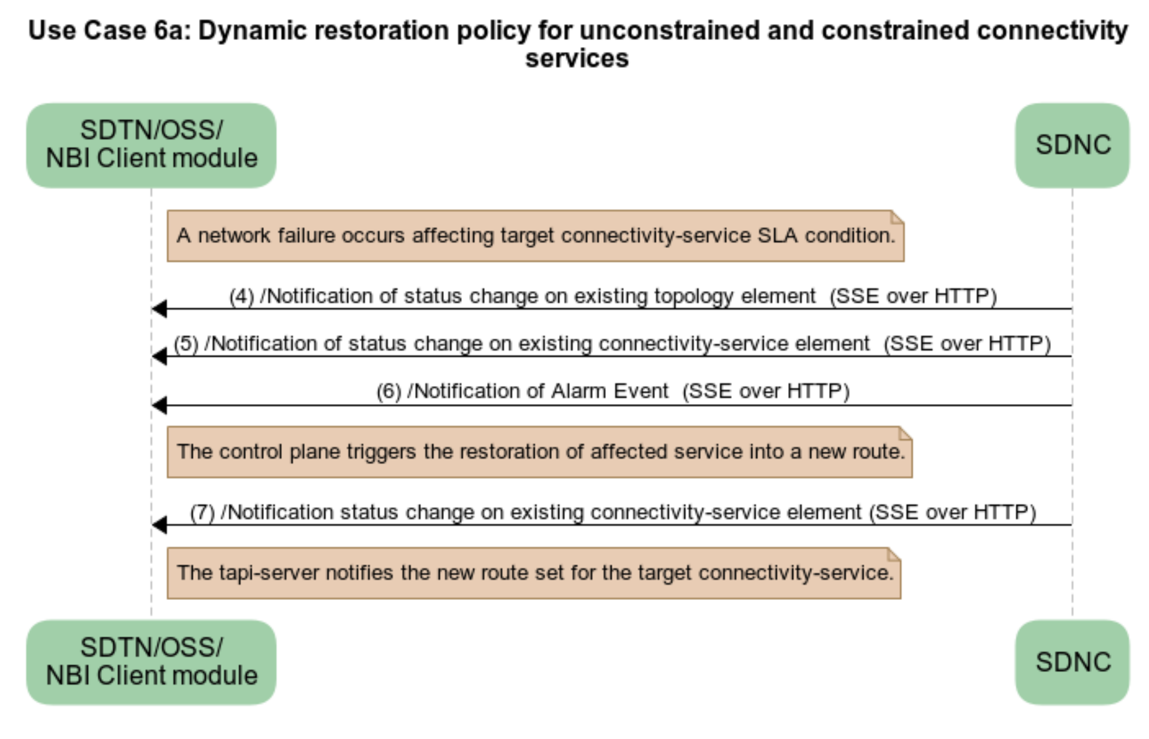


Figure 6-34 UC-6a: Resiliency workflow.

6.5.4.1 Required parameters

Table 40 complements the information included in the Use Case 1a and Use Case 5b definitions, with the Connectivity-Service attributes required to implement this use case, this implies all attributes and objects included in Section 6.2.1.1 tables are required for this use case too.

Table 40: Connectivity-service attributes for UC6a.

connectivity-service	**/tapi-common:context/tapi-connectivity:connectivity-context/connectivity-service**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
resilience-type	{" protection-type": [DYNAMIC_RESTORATION]}	RW	M	• Provided by *tapi-client*
preferred-restoration-layer	[ODU, PHOTONIC_MEDIA]	RW	M	• Provided by *tapi-client*
reversion-mode	["REVERTIVE", "NON-REVERTIVE"]	RW	M	• Provided by tapi-client
restore-priority	"[0-9]+"	RW	O	• Provided by tapi-client

6.5.5 Use case 6b: Pre-Computed restoration policy for unconstrained and constrained connectivity services.

Number	**UC 6b**
Name	Pre-Computed restoration policy for unconstrained and constrained connectivity services.
Technologies involved	Optical
Process/Areas Involved	Planning and Operations
Brief description	This use case covers the provisioning of connectivity-services with restoration capabilities and the effective restoration process when a failure occurs in the nominal path. This use case assumes the same definition, workflow and specification defined in UC6a. Additionally, the SDN-C MUST accept a pre-computed (preset) restoration path as part of the connectivity-service provisioning request. Please note this use case depends on the use case 12b. The path computation operation might include routing policies (i.e., min. hops, min. latency) and/or constrains (the same applicable to the creation of services I .e., use cases 3). The underlying connection provisioning and management (including lower OCh/OTSi layer connections) is performed by the SDN-C. Also, the new path calculation in case of failure of the working path. The dynamic restoration process MUST be triggered automatically by the TAPI server, however, the TAPI client MUST be notified about the service condition changes through the **tapi-notification** service (as defined in UCs 15a and 15b). The SDN-C must provide asynchronous notifications of the events generated by the network failure which triggers the restoration including, but not limited to, topology changes, service status changes, triggered alarms.
Layers involved	ODU, PHOTONIC_MEDIA
Type	Resilience
Description & Workflow	The initial path computation procedure is the same defined in UC12b. In particular, the two disjoined paths between the service-interface-points included in the Connectivity-Service request are previously computed by the SDN-C. This UC's connectivity-service provisioning and protection procedures follows the same workflow defined in the "Procedure" section of UC6a plus the following sub-workflow: The nominal and the pre-computed restoration paths MUST be included within the **tapi-connectivity:connectivity-service/tapi-topology:include-path** attribute. Due to model limitations, the specification of the nominal and restoration route SHOULD be done sequentially. 1) First the tapi-client includes the nominal path in the Connectivity-Service initial POST request. 2) Once the connectivity-service creation is completed, the tapi-client has to modify the existing Connectivity-Service object by adding the pre-computed restoration path, into the **tapi-connectivity:connectivity-service/tapi-topology:include-path** leaf-list attribute. The TAPI user MUST use the PUT operation to modify the existing object. The Connectivity Service object sent to the SDN-C MUST include the tapi-**connectivity:connectivity-service/tapi-topology:resilience-type/protection-type** attribute with PRE_COMPUTED_RESTORATION attribute value. <u>Resiliency workflow</u>:

	The UC's protection workflow, of this UC, follows the same workflow defined in the "Procedure" section of UC6a.

6.5.5.1 Required parameters

Table 41 complements the information included in the Use Case 1a and Use Case 5b definitions, with the Connectivity-Service attributes required to implement this use case, this implies all attributes and objects included in Section 6.2.1.1 tables are required for this use case too.

Table 41: Connectivity-service attributes for UC6a.

connectivity-service	**/tapi-common:context/tapi-connectivity:connectivity-context/connectivity-service**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
resilience-type	{" protection-type": [DYNAMIC_RESTORATION]}	RW	M	• Provided by *tapi-client*
preferred-restoration-layer	[ODU, PHOTONIC_MEDIA]	RW	M	• Provided by *tapi-client*
reversion-mode	["REVERTIVE", "NON-REVERTIVE"]	RW	M	• Provided by tapi-client
restore-priority	"[0-9]+"	RW	O	• Provided by tapi-client

6.5.6 Use case 7a: Dynamic restoration and 1+1 protection of DSR/ODU unconstrained service provisioning.

Number	**UC7a**
Name	**Restoration and protection 1+1**
Technologies involved	Optical
Process/Areas Involved	Planning and Operations
Brief description	This use case covers the provisioning of connectivity-services with restoration capabilities and 1+1 protection capabilities. This first sub use case of this block proposes a 1+1 protection schema, which can be implemented either: • Over the PHOTONIC_MEDIA layer as the OLP Protection schema defined in section 6.5.2 • Over the ODU layer as the eSNCP protection schema defined in section 6.5.3 This protection schema MUST provide automatic switch control between the working and protection connections in less than 50ms. Moreover, this use case introduces a second level of protection which is implemented through dynamic restoration of the first connection affected by a failure. The dynamic restoration policy MUST be implemented just one time. The Connectivity-Service can be requested at different layers i.e., DSR, ODU. The TAPI client just need to specify two available Connectivity Service End Points and introduce the service characteristics (including the restoration-type and protection-type attributes). The TAPI server is responsible of maintaining the SLA condition and triggers the protection and dynamic restoration processes when the connectivity-service working path is being affected by any outage condition which provoke a degradation or interruption of the service. The TAPI server is expected to calculate the new restoration path according to the current network conditions when the SLA failure occurs. The protection and dynamic restoration processes MUST be triggered automatically by the TAPI server, however, the TAPI client MUST be notified about the service condition changes through the tapi-notification service. The description of the notification process and its detail Low-Level-Design is out of the scope of this use case definition. Additional constrains, such coroute-inclusion or diversity-exclusion SHALL be account as **loose constrains** at the time of the restoration occurs, i.e., applicable if possible
Layers involved	DSR, ODU, PHOTONIC_MEDIA
Type	Resilience
Description & Workflow	The connectivity service is requested between two DSR/ODU CSEPs. The TAPI Client SHOULD select two valid SIPs, to be referenced by the CSEPs, which reference two DSR/LO-ODUj NEPs which are intended to be connected by the dynamic-restorable service. The TAPI Client MAY include the ***tapi-connectivity:preferred-restoration-layer*** object to specify the preferred restoration layer, but the final decision on how to implement the restoration is responsibility of the TAPI server based on the current network conditions when the dynamic restoration process occurs. The Connectivity Service object sent to the TAPI Server MUST include the ***tapi-connectivity:connectivity-service/tapi-topology:resilience-type/protection-type*** attribute with **ONE_PLUS_ONE_PROTECTION_WITH_DYNAMIC_RESTORATION** attribute value. This UC is implemented following the same workflow described in “Description & Workflow” of UC1a described in section 6.2.1 Resiliency workflow:

	The service with this SLA MUST be able to support a first failure within the nominal or protection paths and switched within 50ms. After the first failure, the affected path is dynamically restored by the control plane, by calculating a new path which may imply a wavelength change. 1) The first failure affecting the nominal path MUST trigger the protection workflow described in “Description & Workflow” of UC5b described in section 6.5.2 2) The restoration of the affected route by the first failure MUST trigger the protection workflow described in “Description & Workflow” of UC6a described in section 6.5.4

6.5.6.1 Required parameters

Table 42 complements the information included in the Use Case 1 and Use Case 5b definitions, with the Connectivity-Service, Connectivity-Service-End-Points, Connections and Switch-control, attributes required to implement this use case, this implies all attributes and objects included in Section 6.2.1.1 and 6.5.2.2 tables are required for this use case too.

Table 42: Connectivity-service attributes for UC7a.

connectivity-service	**/tapi-common:context/tapi-connectivity:connectivity-context/connectivity-service**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
resilience-type	{"**protection-type**": [ONE_PLUS_ONE_PROTECTION_WITH_DYNAMIC_RESTORATION]}	RW	M	• Provided by *tapi-client*
preferred-restoration-layer	[ODU, PHOTONIC_MEDIA]	RW	M	• Provided by *tapi-client*
reversion-mode	["REVERTIVE", "NON-REVERTIVE"]	RW	M	• Provided by tapi-client
restore-priority	"[0-9]+"	RW	O	• Provided by tapi-client
hold-off-time	"[0-9]{4}"	RW	O	• Provided by tapi-client
wait-to-revert-time	"[0-9]{4}"	RW	O	• Provided by tapi-client
max-switch-times	"[0-9]{2}"	RW	O	• Provided by tapi-client
is-coordinated-switching-both-ends	[true, false]	RW	O	• Provided by tapi-client
is-lock-out	[true, false]	RW	O	• Provided by tapi-client
is-frozen	[true, false]	RW	O	• Provided by tapi-client

6.5.7 Use case 7b: Pre-Computed restoration policy and 1+1 protection of DSR/ODU unconstrained service provisioning.

Number	**UC7b**
Name	**Pre-Computed restoration policy and 1+1 protection of DSR/ODU unconstrained service provisioning.**
Technologies involved	Optical
Process/Areas Involved	Planning and Operations
Brief description	This use case covers the provisioning of connectivity-services with restoration capabilities and 1+1 protection capabilities and the demonstration of the effective protection and restoration process when a failure occurs in the nominal path. This use case of this block proposes a 1+1 protection schema, which can be implemented either: ▪ At the OTSi_PHOTONIC_MEDIA layer, as the OLP Protection schema defined in UC5b. ▪ Or at the ODU layer, as the eSNCP protection schema defined in UC5c. Additionally, the SDN-C MUST accept a pre-computed (preset) restoration path as part of the connectivity-service provisioning request. Please note this use case depends on the use case 12a. The path computation operation might include routing policies (i.e., min. hops, min. latency) and/or constrains (the same applicable to the creation of services i.e., use cases 3). The underlying connection provisioning and management (including lower OCh/OTSi layer connections) is performed by the SDN-C. Also the new path calculation in case of failure of the working path The protection and dynamic restoration processes MUST be triggered automatically by the TAPI server, however, the TAPI client MUST be notified about the service condition changes through the tapi-notification service (as defined in UCs 15a and 15b). The SDN-C must provide asynchronous notifications of the events generated by the network failure which triggers the restoration including, but not limited to, topology changes, service status changes, triggered alarms… Additional constrains, such **coroute-inclusion** or **diversity-exclusion** SHALL be account as loose constrains at the time of the restoration occurs, i.e., applicable if possible.
Layers involved	DSR, ODU, PHOTONIC_MEDIA
Type	Resilience
Description & Workflow	The initial path computation procedure is identical to the one included in UC12b. The UC's service provisioning and protection procedures, of this UC, follows the same workflow defined in the "Procedure" section of UC7a. <u>Resiliency workflow</u>: The service with this SLA MUST be able to support a first failure within the nominal or protection paths and switched within 50ms. After the first failure, the affected path is restored through the pre-computed path defined in the service. 1) The first failure affecting the nominal path MUST trigger the protection workflow described in "Description & Workflow" of UC5b described in section 6.5.1. 2) The restoration of the affected route by the first failure MUST trigger the protection workflow described in "Description & Workflow" of UC6a described in section 6.5.4

6.5.7.1 Required parameters

Table 43 complements the information included in the Use Case 1a and Use Case 5b definitions, with the Connectivity-Service, Connectivity-Service-End-Points, Connections and Switch-control, attributes required to implement this use case, this implies all attributes and objects included in Section 6.2.1.1 and 6.5.2.2 tables are required for this use case too.

Table 43: Connectivity-service attributes for UC7b.

connectivity-service	**/tapi-common:context/tapi-connectivity:connectivity-context/connectivity-service**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
resilience-type	{"**protection-type**": [**ONE_PLUS_ONE_PROTECTION_WITH_PRE_COMPUTED_RESTORATION**]}	RW	M	• Provided by *tapi-client*
preferred-restoration-layer	[ODU, PHOTONIC_MEDIA]	RW	M	• Provided by *tapi-client*
reversion-mode	["REVERTIVE", "NON-REVERTIVE"]	RW	M	• Provided by tapi-client
restore-priority	"[0-9]+"	RW	O	• Provided by tapi-client
hold-off-time	"[0-9]{4}"	RW	O	• Provided by tapi-client
wait-to-revert-time	"[0-9]{4}"	RW	O	• Provided by tapi-client
max-switch-times	"[0-9]{2}"	RW	O	• Provided by tapi-client
is-coordinated-switching-both-ends	[true, false]	RW	O	• Provided by tapi-client
is-lock-out	[true, false]	RW	O	• Provided by tapi-client
is-frozen	[true, false]	RW	O	• Provided by tapi-client

6.5.8 Use case 8: Permanent protection 1+1 for use cases

Number	**UC8**
Name	Permanent protection 1+1 for use cases
Technologies involved	Optical
Process/Areas Involved	Planning and Operations
Brief description	This use case covers the provisioning of connectivity-services with permanent 1+1 protection capabilities and the demonstration of the effective protection and restoration process when a failure occurs in the nominal or backup paths. This use case of this block proposes a 1+1 protection schema, which can be implemented either: ▪ At the OTSi_PHOTONIC_MEDIA layer, as the OLP Protection schema defined in UC5b ▪ At the ODU layer, as the eSNCP protection schema defined in UC5c The underlying connection provisioning and management (including lower OCh/OTSi layer connections) is performed by the SDN-C. Also, the new path calculation in case of failure of the working path MUST be computed and provisioned by the SDN-C according to the control-plane policies installed in the network. The TAPI server is responsible of maintaining the SLA condition and triggers the automatic protection/switching actions when the connectivity-service nominal path is being affected by any outage condition which provoke a degradation or interruption of the service. The protection process MUST be triggered automatically by the TAPI server, however, the TAPI client MUST be notified about the service condition changes through the tapi-notification service (as defined in UCs 15a and 15b). The SDN-C must provide asynchronous notifications of the events generated by the network failure which triggers the restoration including, but not limited to, topology changes, service status changes, triggered alarms…
Layers involved	DSR, ODU, PHOTONIC_MEDIA
Type	Resilience
Description & Workflow	The UC's service provisioning procedure of this UC follows the same workflow defined in the "Procedure" section of UC5c. Additional considerations: The Connectivity Service object sent to the SDN-C MUST include the tapi-connectivity:connectivity-service/tapi-topology:resilience-type/protection-type attribute with PERMANENT_ONE_PLUS_ONE_PROTECTION attribute value. Resiliency workflow: The service implementing **PERMANENT_ONE_PLUS_ONE_PROTECTION** protection schema MUST be able to switch from the nominal path to the backup path against any failure which degrades the service under the defined SLA conditions within 50ms. There are 2 possible fails: • If the working path fails, the control plane will commute to the protection path. The role of this path changes from protected to working and the control plane will compute and provision a new protection path. This procedure will occur permanently. The protection workflow described in “Description & Workflow” of UC5b described in section 6.5.1. • If the protection path fails, the control plane will compute and restore it through a new protection path. This procedure will be permanent as the previous one and it MUST trigger the same workflow described in “Description & Workflow” of UC6a described in section 6.5.4.

6.5.8.1 Required parameters

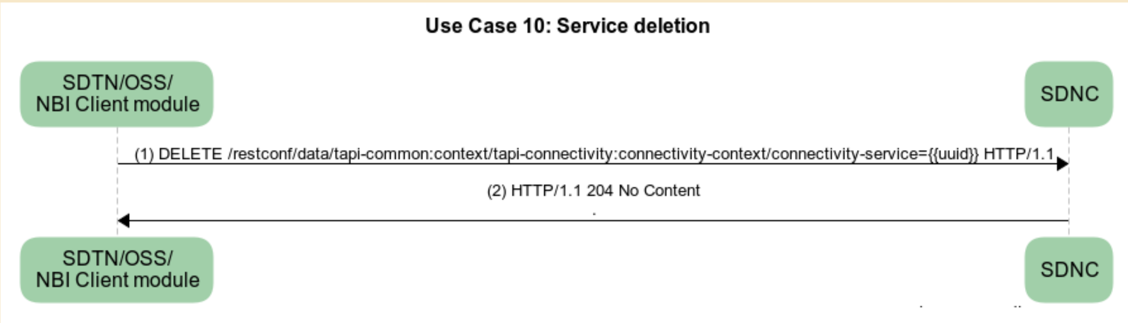
The following table complements the information included in the Use Case 1a and Use Case 5b definitions, with the Connectivity-Service, Connectivity-Service-End-Points, Connections and Switch-control, attributes required to implement this use case, this implies all attributes and objects included in Section 6.2.1.1 and 6.5.2.2 tables are required for this use case too.

Table 44: Connectivity-service attributes for UC8.

connectivity-service	**/tapi-common:context/tapi-connectivity:connectivity-context/connectivity-service**			
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
resilience-type	{"protection-type": [PERMANENT_ONE_PLUS_ONE_PROTECTION]}	RW	M	• Provided by *tapi-client*
preferred-restoration-layer	[ODU, PHOTONIC_MEDIA]	RW	M	• Provided by *tapi-client*
reversion-mode	["REVERTIVE", "NON-REVERTIVE"]	RW	M	• Provided by tapi-client
restore-priority	"[0-9]+"	RW	O	• Provided by tapi-client
hold-off-time	"[0-9]{4}"	RW	O	• Provided by tapi-client
wait-to-revert-time	"[0-9]{4}"	RW	O	• Provided by tapi-client
max-switch-times	"[0-9]{2}"	RW	O	• Provided by tapi-client
is-coordinated-switching-both-ends	[true, false]	RW	O	• Provided by tapi-client
is-lock-out	[true, false]	RW	O	• Provided by tapi-client
is-frozen	[true, false]	RW	O	• Provided by tapi-client

6.6 Maintenance

6.6.1 Use Case 10: Service deletion (applicable to all previous use cases)

Number	**UC10**
Name	**Service deletion (applicable to all previous use cases)**
Technologies involved	Optical
Process/Areas Involved	Planning and Operations
Brief description	This use case covers the deletion of a target connectivity-service and all the related connections created by this service (i.e., all connections included as part of the CS connection list). If the user is not authorized to delete the target resource, then an error response containing a "403 Forbidden" status-line SHOULD be returned. The error-tag value "access-denied" is returned in this case. A server MAY return a "404 Not Found" status-line, as described in Section 6.5.4 in [RFC7231]. The error-tag value "invalid-value" is returned in this case.
Layers involved	DSR, ODU, PHOTONIC_MEDIA
Type	Resilience
Description & Workflow	The TAPI client MUST specify the ***tapi-connectivity:connectivity-service/uuid*** attribute in the RESTCONF DELETE request to identify the service to be removed.  **Figure 6-35 UC-10: Service Deletion workflow.**

6.7 Planning

6.7.1 Use case 12a: Pre-calculation of the optimum path (applicable to all previous use cases)

<table>
<tr><td>Number</td><td>UC12a</td></tr>
<tr><td>Name</td><td>Pre-calculation of the optimum path (applicable to all previous use cases)</td></tr>
<tr><td>Technologies involved</td><td>Optical</td></tr>
<tr><td>Process/Areas Involved</td><td>Planning and Operations</td></tr>
<tr><td>Brief description</td><td>[Disclaimer]: This use case is not completely discussed, nor agreed by the TAPI team. This use case remains documented for future discussion and the content of it may vary in future versions. Thus, the use of the documentation for this use case should be considered only for Proof of Concept purposes.

The SDN-C must support the pre-calculation of a service path between two endpoints (site/interface) of the network including the network layer (i.e., DSR, ODU, OTSi/OTSiA, MC).

The underlying calculation of the new path is made by the SDN-C, and it might include regeneration (3R).

The path computation operation might include routing policies (i.e., min. hops, min. latency) and/or constrains (the same applicable to the creation of services i.e., use cases 3).

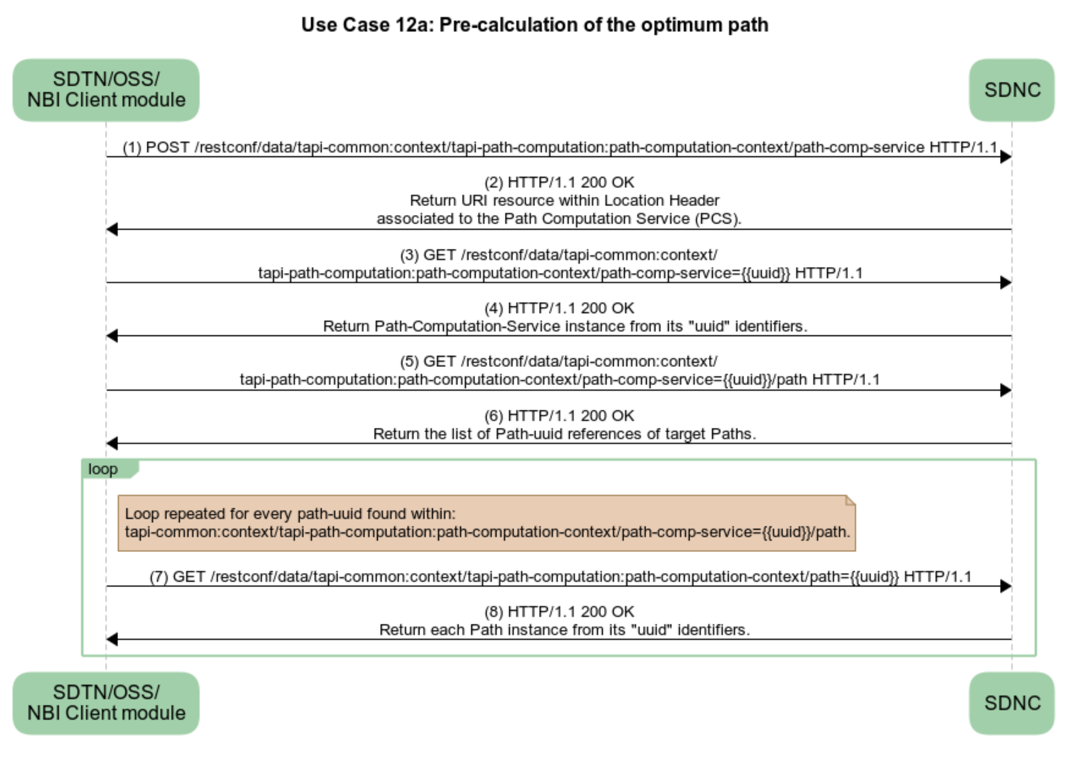
If not specific layer is included within the request, the calculated path MUST include all routed links even if they belong to multiple layers.</td></tr>
<tr><td>Layers involved</td><td>DSR, ODU, PHOTONIC_MEDIA</td></tr>
<tr><td>Type</td><td>Planning</td></tr>
<tr><td>Description & Workflow</td><td>


Figure 6-36 UC-12a: Pre-calculation of the optimum path workflow.</td></tr>
</table>

6.7.1.1 Required parameters

Table 45: Path-computation-context attributes.

path-computation-context				
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
path-comp-service	List of {**path-comp-service**}	RW	M	• Provided by *tapi-client*
path	List of {**path**}	RO	M	• Provided by *tapi-server*

Table 46: Path-comp-serv object's attributes.

path-comp-serv				
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
end-point	List of {**path-service-end-point**}	RW	M	• Provided by *tapi-client*
routing-constraint	{ **routing-constraint** }	RW	M	• Provided by *tapi-client*
topology-constraint	{ **topology-constraint** }	RW	M	• Provided by *tapi-client*
objective-function	{**objective-function**}	RW	M	• Provided by *tapi-client* •
optimization-constraint	{**optimization-constraint**}			•
uuid	"[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}"	RW	M	• As per RFC 4122 • Provided by ***tapi-client***
name	List of {value-name: value} • "value-name": "PATH_COMP_REQ_NAME" "value": " [0-9a-zA-Z_]{64}"	RW	M	• Provided by *tapi-client*

Table 47: Path-service endpoint object's attributes.

path-service-end-point				
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
local-id	"[0-9a-zA-Z_]{32}"	RW	M	• Provided by *tapi-client*
layer-protocol-name	List of ["DSR", "ODU", "PHOTONIC_MEDIA"]	RW	M	• Provided by *tapi-client*
layer-protocol-qualifier	List of ["DIGITAL_SIGNAL_TYPE", "ODU_TYPE", "PHOTONIC_LAYER_QUALIFIER”]	RW	M	• Provided by tapi-server • All children identities defined for ["DIGITAL_SIGNAL_TYPE", "ODU_TYPE", "PHOTONIC_LAYER_QUALIFIER”] MUST be supported.

administrative-state	["UNLOCKED", "LOCKED"]	RW	O	• Provided by *tapi-client*
operational-state	["ENABLED", "DISABLED"]	RO	O	• Provided by *tapi-server*
lifecycle-state	["PLANNED", "POTENTIAL_AVAILABLE", "POTENTIAL_BUSY", "INSTALLED", "PENDING_REMOVAL"]	RO	O	• Provided by *tapi-server*
direction	["BIDIRECTIONAL", "INPUT", "OUTPUT"]	RW	O	• Provided by *tapi-client*
role	["SYMMETRIC", "TRUNK"]	RW	O	• Provided by *tapi-client* • Support only P2P
capacity	"total-size": {value: unit} • "value": "[0-9]{8}", • "unit": ["TB", "TBPS", "GB", "GBPS", "MB", "MBPS", "KB", "KBPS"]	RW	O	• Provided by *tapi-client*
service-interface-point	*"/tapi-common:context/service-interface-point/uuid*"	RW	M	• Provided by *tapi-client*

Table 48: Topology constraint object's attributes.

topology-constraint				
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
include-topology	LeafList of { "[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}" }	RW	O	• This is a loose constraint - that is it is unordered and could be a partial list
avoid-topology	LeafList of { "[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}" }	RW	O	• This is a loose constraint - that is it is unordered and could be a partial list
include-path	LeafList of { "[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}" }	RW	M	• Provided by *tapi-client* • This is a loose constraint - that is it is unordered and could be a partial list. • The uuid MUST refer to a valid { **tapi-path-computation:path}** object present within the tapi-server datastore.
exclude-path	LeafList of {	RW	M	• Provided by *tapi-client* • This is a loose constraint - that is it is unordered and could be a partial list.

	"[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}" }			• The uuid MUST refer to a valid {**tapi-path-computation:path**} object present within the tapi-server datastore
include-node	LeafList of { "[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}" }	RW	M	• Provided by *tapi-client* • This is a loose constraint - that is it is unordered and could be a partial list. • The uuid MUST refer to a valid {**tapi-topology:node**} object present within the tapi-server datastore
exclude-node	LeafList of { "[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}" }	RW	M	• Provided by *tapi-client* • Reference to an existing node-id already present in the TAPI server context MUST be valid. • The uuid MUST refer to a valid {**tapi-topology:node**} object present within the tapi-server datastore
include-link	LeafList of { "[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}" }	RW	M	• Provided by *tapi-client* • This is a loose constraint - that is it is unordered and could be a partial list • The uuid MUST refer to a valid {**tapi-topology:link**} object present within the tapi-server datastore
exclude-link	LeafList of { "[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}" }	RW	M	• Provided by *tapi-client* • This is a loose constraint - that is it is unordered and could be a partial list • The uuid MUST refer to a valid {**tapi-topology:link**} object present within the tapi-server datastore
preferred-transport-layer	[ODU, PHOTONIC_MEDIA]	RW	M	• Provided by *tapi-client*

Table 49: Routing constraint object's attributes.

routing-constraint				
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
cost-characteristic	{ cost-name, cost-value, cost-algorithm} • "cost-name": "string", • "cost-value": "string", • "cost-algorithm": "string",	RW	M	• Provided by *tapi-client*
latency-characteristic	{ traffic-property-name, fixed-latency-characteristic, queing-latency-characteristic, jitter-characteristic, wander-characteristic } • "traffic-property-name": "string", • "fixed-latency-characteristic": "string", • "queing-latency-characteristic": "string", • "jitter-characteristic":"string" • "wander-characteristic":"string"	RW	M	• Provided by *tapi-client*

risk-diversity-characteristic	{ risk-characteristic-name, risk-identifier-list} • risk-characteristic-name • risk-identifier-list	RW	M	• Provided by *tapi-client*
diversity-policy	{SRLG, SRNG, SNG,NODE, LINK}	RW	M	• Provided by *tapi-client*
route-objective-function	["MIN_WORK_ROUTE_HOP", "MIN_WORK_ROUTE_COST", "MIN_WORK_ROUTE_LATENCY", "MIN_SUM_OF_WORK_AND_PROTECTION_ROUTE_HOP", "MIN_SUM_OF_WORK_AND_PROTECTION_ROUTE_COST" , "MIN_SUM_OF_WORK_A ND_PROTECTION_ROUTE_LATENCY", "LOAD_BALANCE_MAX_UNUSED_CAPACITY"	RW	M	• Provided by tapi-client
route-direction	["BIDIRECTIONAL", "INPUT", "OUTPUT"]	RW	M	• Provided by *tapi-client*
is-exclusive	Boolean	RW	M	• Provided by *tapi-client*

Table 50: Objective function object's attributes.

objective-function				
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
bandwidth-optimization	{"MINIMIZE", "MAXIMIZE", "ALLOW", "DISALLOW", "DONT_CARE"]	RW	M	• Provided by *tapi-client*
concurrent-paths	{"MINIMIZE", "MAXIMIZE", "ALLOW", "DISALLOW", "DONT_CARE"]	RW	M	• Provided by *tapi-client*
cost-optimization	{"MINIMIZE", "MAXIMIZE", "ALLOW", "DISALLOW", "DONT_CARE"]	RW	M	• Provided by *tapi-client*
link-utilization	{"MINIMIZE", "MAXIMIZE", "ALLOW", "DISALLOW", "DONT_CARE"]	RW	M	• Provided by *tapi-client*
resource-sharing	{"MINIMIZE", "MAXIMIZE", "ALLOW", "DISALLOW", "DONT_CARE"]	RW	M	• Provided by tapi-client
local-id	"[0-9a-zA-Z_]{32}"	RW	M	• Provided by *tapi-client*
name	List of {value-name: value} • "value-name": "OBJ_FUNCTION" "value": " [0-9a-zA-Z_]{64}"	RW	O	• Provided by *tapi-client*

Table 51: Optimization-constraint object's attributes.

optimization-constraint				
Attribute	**Allowed Values/Format**	**Mod**	**Sup**	**Notes**
traffic-interruption	{ "MINIMIZE", "MAXIMIZE", "ALLOW", "DISALLOW", "DONT_CARE"]	RW	M	• Provided by *tapi-client*
local-id	"[0-9a-zA-Z_]{32}"	RW	M	• Provided by *tapi-client*
name	List of {value-name: value} • "value-name": "OPT_CONSTRAINT_NAME" • "value": " [0-9a-zA-Z_]{64}"	RW	O	• Provided by *tapi-client*

6.7.2 Use case 12b: Simultaneous pre-calculation of two disjoint paths

Number	**UC12b**
Name	**Simultaneous pre-calculation of two disjoint paths**
Technologies involved	Optical
Process/Areas Involved	Planning and Operations
Brief description	**[Disclaimer]: This use case is not completely discussed, nor agreed by the TAPI team. This use case remains documented for future discussion and the content of it may vary in future versions. Thus, the use of the documentation for this use case should be considered only for Proof of Concept purposes.** The SDN-C must support the pre-calculation of a service path between two endpoints (site/interface) of the network including the network layer (i.e., DSR, ODU, OCh/OTSi, MC). The underlying calculation of the new path is made by the SDN-C, and it might include regeneration (3R). The path computation operation might include routing policies (i.e., min. hops, min. latency) and/or constrains (the same applicable to the creation of services i.e., use cases 3). If not specific layer is included within the request, the calculated path MUST include all routed links even if they belong to multiple layers.
Layers involved	DSR, ODU, PHOTONIC_MEDIA
Type	Planning
Description & Workflow	The simulateneous calculation is not yet ready in the standard NBI at the moment, thus a secuential approach will be followed. The approach is to request two paths sequentially and impose a "exclude-path" constrain to the second path-request by including a reference to the previously calculated, thus assuring the second path is disjoint from the previous one.

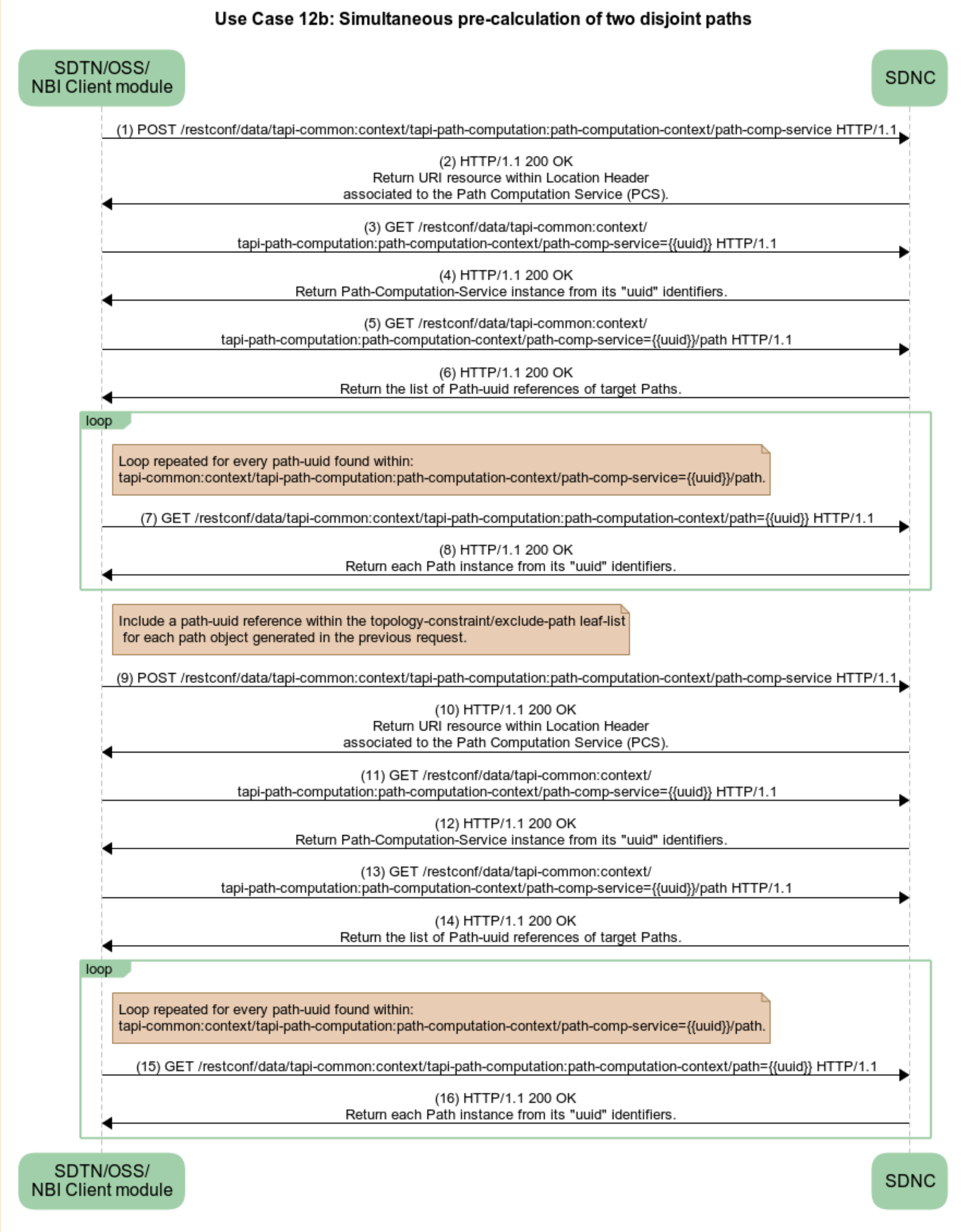


Figure 6-37 UC-12b: Simultaneous pre-calculation of two disjoint paths

6.8 Notifications and alarms.

6.8.1 Use case 13a: Subscription to notification service.

Number	**UC 13a**
Name	**Subscription to notification service.**
Technologies involved	All
Process/Areas Involved	Planning and Operations
Brief description	The SDN-C MUST offer a notification subscription service which allows several client applications to subscribe to asynchronous notifications about the changes occurred in the network. The notification stream service SHALL allow notification filtering to select the relevant information upon the use case to be implemented. The filtering system MUST allow filtering by object-type (i.e., Connectivity-Service, Connection…), by networking layer, by notification-type (OBJECT_CREATION, ATTRIBUTE_VALUE_CHANGE, OBJECT_DELETION, ALARM_EVENT, THRESHOLD_CROSSING_ALERT). The same client application MAY request more than one notification stream with different filtering characteristics. All NOTIFICATIONs emitted by the SDN-C through a dedicated subscribtion channel must be correlated linearly by: `/tapi-common:context/tapi-notification:notification-context/tapi-notification:notification:` `+--ro sequence-number?            uint64` And timely by: `/tapi-common:context/tapi-notification:notification-context/tapi-notification:notification:` `+--ro event-time-stamp?          tapi-common:date-and-time` The TAPI server MUST support the RESTCONF Notifications subscription mechanism is defined in Section 6.3 of [RFC 8040]. In addition the TAPI server must support the *"filter"* Query Parameter, as defined in Section 4.8.4 of [RFC 8040], to indicate the target subset of the possible events being advertised by the RESTCONF server stream. The TAPI server MUST implement the defined filtering mechanism following the [XPath] format. Here we define some possible filters that can be designed by the *tapi-client* and which MUST be accepted by the server. **Please note the scope of the filtering mechanism is not restricted to the examples proposed.** Thus, please considers the following examples as further documentation to help the development teams to understand the defined service. Without loss of generality, for the examples please assume all notifications are defined within the custom "*tapi-notification*" stream. `Example 1:` `----------------------------------------------------------` `// filter = /tapi-notification:notification/notification-type='OBJECT_CREATION'` `GET /streams/tapi-notification?filter=%2Ftapi-notification%3Anotification%2Fnotification-type%3D'OBJECT_CREATION'` `Example 2:` `----------------------------------------------------------`

```
// filter = (/tapi-notification:notification/notification-
type='ATTRIBUTE_VALUE_CHANGE' and /tapi-notification:notification/target-object-
type='NODE')

      GET /streams/tapi-notification?filter=%2Ftapi-
notification%3Anotification%2Fnotification-
type%3D'ATTRIBUTE_VALUE_CHANGE'%20and%20%2Ftarget-object-type%3D'NODE'

Example 3:
----------------------------------------------------------
// filter = /tapi-notification:notification/target-object-
name['INVENTORY_ID']/value[contains(.,'/ne=MadridNorte')]

      GET /streams/tapi-notification?filter=%2Ftapi-
notification%3Anotification%2Ftarget-object-
name['INVENTORY_ID']%2Fvalue[contains(.%2C'/ne=MadridNorte')]
```

Layers involved	DSR, ODU, PHOTONIC_MEDIA
Type	Notifications and Alarms

Description & Workflow

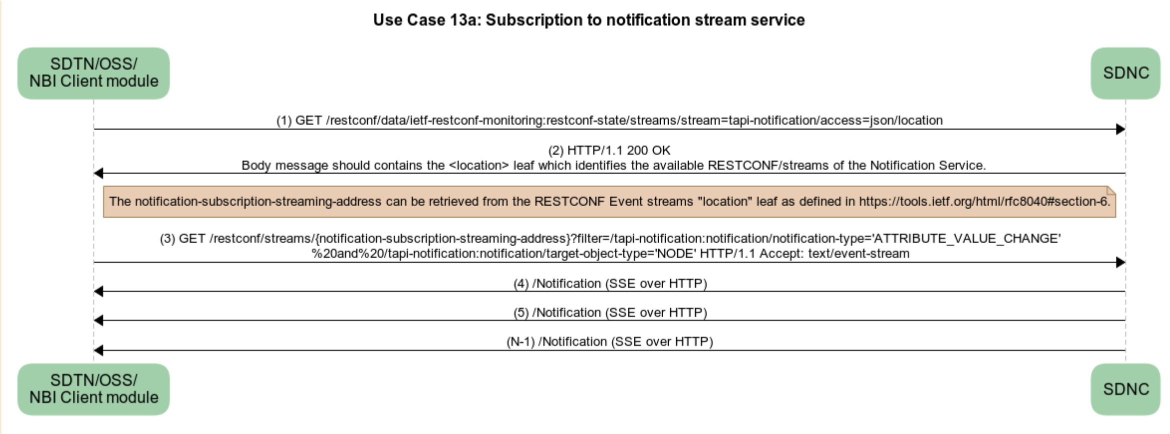


Figure 6-38 UC-13a: Subscription to notification stream service

Please assume RESTCONF Event Notifications MUST be supported by the SDN-C, while the TAPI Notification subscription mechanism is optional. The asyncronous notification transport protocol MUST be SSE (http://www.w3.org/TR/2015/REC-eventsource-20150203/).

6.8.2 Use case 14a: Notification of new topology element (topology, link, node, node-edge-point) inserted/removed in/from the network

Number	**UC 14a**
Name	**Notification of new topology element (topology, link, node, node-edge-point) inserted/removed in/from the network**
Technologies involved	All
Process/Areas Involved	Planning and Operations
Brief description	The SDN-C Notification system MUST emit events exposing the creation/deletion of Topology object-types such topology, link, node and node-edge-points. The SDN-C MUST report a TOPOLOGY object creation/deletion notification when a network element is introduced or removed from the network or a logical construct is generated as part of a connection or service is provisioned in the network. A SDN-C NBI client which is subscribed to the notification service with the filter which includes the following parameters: • notification-type= OBJECT_CREATION, OBJECT_DELETION • object-type= TOPOLOGY, NODE, LINK or NODE_EDGE_POINT , it must receive all notification which are address by the filter with the corresponding object and notification types and identifiers: `/tapi-common:context/tapi-notification:notification-context/tapi-notification:notification:` `+--ro notification-type?          notification-type` `+--ro target-object-type?         tapi-common:object-type` `+--ro target-object-identifier?   tapi-common:uuid` `+--ro target-object-name* [value-name]` `|  +--ro value-name    string` `|  +--ro value?        string` Please note the INVENTORY_ID name SHOULD be included as "target-object-name" in the notification as additional information.
Layers involved	DSR, ODU, PHOTONIC_MEDIA
Type	Notifications and Alarms
Description & Workflow	

6.8.3 Use case 14b: Notification of new connectivity-service element inserted/removed in/from the network.

Number	**UC 14b**
Name	**Notification of new connectivity-service element inserted/removed in/from the network.**
Technologies involved	All
Process/Areas Involved	Planning and Operations
Brief description	The SDN-C Notification system MUST emit events exposing the creation/deletion of Connectivity object-types such connectivity-services, connections, connection-end-points and service-interface-points. The SDN-C MUST report a CONNECTIVITY object creation/deletion notification when a network element is introduced or removed from the network or a logical construct is generated as part of a connection or service is provisioned in the network. A SDN-C NBI client which is subscribed to the notification service with the filter which includes the following parameters: • notification-type= OBJECT_CREATION, OBJECT_DELETION • object-type= CONNECTIVITY_SERVICE, CONNECTION, CONNECTION-END-POINT, SERVICE-INTERFACE-POINTS , it must receive all notification which are address by the filter with the corresponding object and notification types and identifiers: `/tapi-common:context/tapi-notification:notification-context/tapi-notification:notification:` `+--ro notification-type?           notification-type` `+--ro target-object-type?          tapi-common:object-type` `+--ro target-object-identifier?    tapi-common:uuid` `+--ro target-object-name* [value-name]` `|  +--ro value-name     string` `|  +--ro value?         string` Please note the INVENTORY_ID name SHOULD be included as "target-object-name" in the notification as additional information in case of a SERVICE-INTERFACE-POINT object is related.
Layers involved	DSR, ODU, PHOTONIC_MEDIA
Type	Notifications and Alarms
Description & Workflow	

6.8.4 Use case 14c: Notification of new path-computation element inserted/removed in/from the network

Number	**UC 14c**
Name	**Notification of new path-computation element inserted/removed in/from the network.**
Technologies involved	All
Process/Areas Involved	Planning and Operations
Brief description	The Notification system MUST emit events exposing the creation/deletion of Path-Computation object-types such path-comp-service, or paths.

	The SDN-C MUST report a Path Computation object creation/deletion notification when a logical construct is generated as part of a connection or service is provisioned in the network. A SDN-C NBI client which is subscribed to the notification service with the filter which includes the following parameters: • notification-type= OBJECT_CREATION, OBJECT_DELETION • object-type=PATH_COMP_SERVICE, PATH , it must receive all notification which are address by the filter with the corresponding object and notification types and identifiers:

```
          /tapi-common:context/tapi-notification:notification-context/tapi-notification:notification:

          +--ro notification-type?          notification-type

          +--ro target-object-type?         tapi-common:object-type

          +--ro target-object-identifier?   tapi-common:uuid

          +--ro target-object-name* [value-name]

          |  +--ro value-name    string

          |  +--ro value?        string
```

Layers involved	DSR, ODU, PHOTONIC_MEDIA
Type	Notifications and Alarms
Description & Workflow	

6.8.5 Use case 15a: Notification of status change on existing topology element (topology, link, node, node-edge-point) in the network.

Number	**UC 15a**
Name	**Notification of status change on existing topology element (topology, link, node, node-edge-point) in the network.**
Technologies involved	All
Process/Areas Involved	Planning and Operations
Brief description	The Notification system MUST emit events exposing the attribute changes of Topology object-types such topology, link, node and node-edge-points. The SDN-C MUST report a TOPOLOGY object change notification when a network element the state of a target resource is modified due to a network condition or user modification. The reason must be notified by: `/tapi-common:context/tapi-notification:notification-context/tapi-notification:notification:` `+--ro source-indicator?           source-indicator` A SDN-C NBI client which is subscribed to the notification service with the filter which includes the following parameters: • notification-type= ATTRIBUTE_VALUE_CHANGE • object-type= TOPOLOGY, NODE, LINK or NODE_EDGE_POINT , it must receive all notification which are address by the filter with the corresponding object and notification types and identifiers: `/tapi-common:context/tapi-notification:notification-context/tapi-notification:notification:` `+--ro notification-type?          notification-type` `+--ro target-object-type?         tapi-common:object-type` `+--ro target-object-identifier?   tapi-common:uuid` `+--ro target-object-name* [value-name]` `|  +--ro value-name    string` `|  +--ro value?        string` Please note the INVENTORY_ID name SHOULD be included as "target-object-name" in the notification as additional information. And it also MUST include the values which has been modified such: `/tapi-common:context/tapi-notification:notification-context/tapi-notification:notification:` `+--ro changed-attributes* [value-name]` `|  +--ro value-name    string` `|  +--ro old-value?    string`

```
          |  +--ro new-value?    string
```

Layers involved	DSR, ODU, PHOTONIC_MEDIA
Type	Notifications and Alarms
Description & Workflow	

6.8.6 Use case 15b: Notification of status change on existing connectivity-service element in the network.

Number	**UC 15b**
Name	**Notification of status change on existing connectivity-service element in the network.**
Technologies involved	All
Process/Areas Involved	Planning and Operations
Brief description	The Notification system MUST emit events exposing the attribute changes of Connectivity object-types such connectivity-services, connections and connection-end-points and service-interface-points. The SDN-C MUST report a Connectivity object change notification when a network element the state of a target resource is modified due to a network condition or user modification. The reason must be notified by: `/tapi-common:context/tapi-notification:notification-context/tapi-notification:notification:` `+--ro source-indicator?           source-indicator` A SDN-C NBI client which is subscribed to the notification service with the filter which includes the following parameters: • notification-type= ATTRIBUTE_VALUE_CHANGE • object-type= CONNECTIVITY_SERVICE, CONNECTION, CONNECTION-END-POINT, SERVICE-INTERFACE-POINTS , it must receive all notification which are address by the filter with the corresponding object and notification types and identifiers: `/tapi-common:context/tapi-notification:notification-context/tapi-notification:notification:` `+--ro notification-type?          notification-type` `+--ro target-object-type?         tapi-common:object-type` `+--ro target-object-identifier?   tapi-common:uuid` `+--ro target-object-name* [value-name]` `|  +--ro value-name    string` `|  +--ro value?        string` Please note the INVENTORY_ID name SHOULD be included as "target-object-name" in the notification as additional information in case of a SERVICE-INTERFACE-POINT object is related. And it also MUST include the values which has been modified such: `/tapi-common:context/tapi-notification:notification-context/tapi-notification:notification:` `+--ro changed-attributes* [value-name]` `|  +--ro value-name    string` `|  +--ro old-value?    string` `|  +--ro new-value?    string`

Layers involved	DSR, ODU, PHOTONIC_MEDIA
Type	Notifications and Alarms
Description & Workflow	

6.8.7 Use case 15c: Notification of status change on the switching conditions of an existing connection element in the network.

Number	**UC 15c**
Name	**Notification of status change on the switching conditions of an existing connection element in the network.**
Technologies involved	All
Process/Areas Involved	Planning and Operations
Brief description	The Notification system MUST emit events exposing the attribute changes of Connection sub-object-types such ROUTE and SWITCH. The SDN-C MUST report a TOPOLOGY object change notification when a network element the state of a target resource is modified due to a network condition or user modification. The reason must be notified by:

```
   /tapi-common:context/tapi-notification:notification-context/tapi-notification:notification:

              +--ro source-indicator?           source-indicator
```

A SDN-C NBI client which is subscribed to the notification service with the filter which includes the following parameters:

- notification-type= ATTRIBUTE_VALUE_CHANGE
- object-type= ROUTE, SWITCH, NODE-RULE-GROUPS

, it must receive all notification which are address by the filter with the corresponding object and notification types and identifiers:

```
          /tapi-common:context/tapi-notification:notification-context/tapi-notification:notification:

          +--ro notification-type?          notification-type

          +--ro target-object-type?         tapi-common:object-type

          +--ro target-object-identifier?   tapi-common:uuid

          +--ro target-object-name* [value-name]

          |  +--ro value-name    string

          |  +--ro value?        string
```

And it also MUST include the values which has been modified such:

	`` /tapi-common:context/tapi-notification:notification-context/tapi-notification:notification: +--ro changed-attributes* [value-name] | +--ro value-name string | +--ro old-value? string | +--ro new-value? string ```
Layers involved	DSR, ODU, PHOTONIC_MEDIA
Type	Notifications and Alarms
Description & Workflow	

7 References

[ONF Core Model] https://www.opennetworking.org/wp-content/uploads/2018/12/TR-512_v1.4_OnfCoreIm-info.zip

[RFC 8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <https://www.rfc-editor.org/info/rfc8040>.

[RFC 6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <https://www.rfc-editor.org/info/rfc6241>.

[RFC 7950] Bjorklund, M., Ed., "The YANG 1.1 Data Modeling Language", RFC 7950, DOI 10.17487/RFC7950, August 2016, <https://www.rfc-editor.org/info/rfc7950>.

[RFC 7895] Bierman, A., Bjorklund, M., and K. Watsen, "YANG Module Library", RFC 7895, DOI 10.17487/RFC7895, June 2016, <https://www.rfc-editor.org/info/rfc7895>.

[OpenAPI] OpenAPI Specification Version 3.0.2, <https://swagger.io/specification/>

[RFC 6455] Fette, I. and A. Melnikov, "The WebSocket Protocol", RFC 6455, DOI 10.17487/RFC6455, December 2011, <https://www.rfc-editor.org/info/rfc6455>.

[W3C.REC-eventsource-20150203] Hickson, I., "Server-Sent Events", World Wide Web Consortium Recommendation REC-eventsource-20150203, February 2015Considerations <http://www.w3.org/TR/2015/REC-eventsource-20150203>.

[ONF TR-527] Functional Requirements for Transport API, June 10, 2016, ONF TR-527, <https://wiki.opennetworking.org/display/OTCC/TAPI+Documentation>

[ONF TR-512] https://www.opennetworking.org/wp-content/uploads/2018/12/TR-512_v1.4_OnfCoreIm-info.zip

[ITU-T G.709] ITU-T G.709: Interfaces for the optical transport network, G.709/Y.1331 (06/16), https://www.itu.int/rec/T-REC-G.709

[ITU-T G.872] ITU-T G.872: Architecture of optical transport networks, ITU-T G.872 2017, https://standards.globalspec.com/std/10165255/ITU-T G.872

[ITU-T G.872] ITU-T G.805: Generic functional architecture of transport networks, (03/2000), https://www.itu.int/rec/dologin_pub.asp?lang=e&id=T-REC-G.805-200003-I!!PDF-E&type=items

[RFC 7951] Lhotka, L., "JSON Encoding of Data Modeled with YANG", RFC 7951, DOI 10.17487/RFC7951, August 2016, <http://www.rfc-editor.org/info/rfc7951>.

[TMF 814] https://www.tmforum.org/resources/reference/mtnm-r4-5-supporting-documents/

8 Definitions

8.1 Terms defined elsewhere

Forwarding Construct [ONF TR-512]

The ForwardingConstruct (FC) represents enabled constrained potential for forwarding between two or more FcPorts (representing the association of the FC to LTPs) at a particular specific Layer Protocol.

Forwarding Domain [ONF TR-512]

The ForwardingDomain (FD) class models the topological component that represents a forwarding capability that provides the opportunity to enable forwarding (of specific transport characteristic information at one or more protocol layers) between points. The FD object provides the context for and constrains the formation, adjustment and removal of FCs and hence offers the potential to enable forwarding.

Logical Termination Point [ONF TR-512]

The LogicalTerminationPoint (LTP) class encapsulates the termination and adaptation functions of one or more transport layers represented by instances of LayerProtocol. The encapsulated transport layers have a simple fixed 1:1 client-server relationship defined by association end ordering. The structure of LTP supports all transport protocols including analogue, circuit and packet forms.

8.2 Terms defined in this TR

Connection

A Connection represents an enabled (provisioned) potential for forwarding (incl. all circuit/packet forms) between two or more Node Edge Points of a Node. The bounding Node of a Connection may be explicit or be conceptually implicit.

Connection is a container for provisioned connectivity that tracks the state of the allocated resources and is distinct from the Connectivity Service request.

Connection End Point

The Connection End Point encapsulates information related to a Connection at the ingress/egress points of every Node that the Connection traverses in a Topology. Thus they represent the ingress/egress port functions (including termination, encapsulation, processing, mapping, etc) of the Connection.

Connectivity Service

A Connectivity Service represents an “intent-like” request for connectivity between two or more Service Interface Points. Connectivity Service is a container for connectivity request details and is distinct from Connection that realizes the request.

Connectivity Service End Point

The Connectivity Service End Point encapsulates information related to a Connectivity Service at the ingress/egress points of that Connectivity Service.

Context

The Context provides a scope of control, naming and information exchange between particular instances of controllers.

Link

A Link is an abstract representation of the effective adjacency between two or more Nodes (specifically Node Edge Points) in a Topology.

Node

The Node is an abstract representation of the forwarding capabilities of a particular set of Network Resources. It is described in terms of the aggregation of set of ports (Node Edge Point) belonging to those Network Resources and the potential to enable forwarding of information between those edge ports.

Node Edge Point

The Node Edge Point represents the ingress-egress edge-port functions that access the forwarding capabilities provided by the Node. Hence it provides an encapsulation of addressing, mapping, termination, adaptation and OAM functions of one or more transport layers (including circuit and packet forms) performed at the entry and exit points of the Node.

Path

The Path is described by an ordered list of Links.

Route

The Route of a Connection is modeled as a collection of Connection End Points.

Service Interface Point

A Service Interface Point represents the network-interface-facing aspects of the edge-port functions that access the forwarding capabilities provided by the Node. Hence it provides a limited, simplified view of interest to external clients (e.g. shared addressing, capacity, resource availability, etc.), that enable the clients to request connectivity without the need to understand the provider network internals.

Topology

The Topology is an abstract representation of the topological aspects of a particular set of Network Resources. It is described in terms of the underlying topological network of Nodes and Links that enable the forwarding capabilities of that particular set of Network Resources.

Transitional Link

Link that is formed by abstracting one or more LTPs (in a stack) to focus on the flow and deemphasize the protocol transformation. This abstraction is relevant when considering multi-layer routing.

8.3 Abbreviations and acronyms

CEP	Connection End Point
CRUD	Create, Read/Retrieve, Update, Delete
CS	Connectivity Service
CSEP	Connectivity Service End Point
DSR	Digital Signal Rate
EMS	Element Management System
FC	Fibre Channel
FC	Forwarding Construct
FD	Forwarding Domain
ILA	InLine Amplifier
INNI	Internal Network-to-Network Interface
JSON	JavaScript Object Notation
LTP	Logical Termination Point
MC	Media Channel
MCA	Media Channel Assembly
MEG	Maintenance Entity Group
MEP	Maintenance Entity Group End Point
NBI	Northbound Interface
NEP	Node Edge Point
NMS	Network Management System
OADM	Optical Add-Drop Multiplexer
OAM	Operations, Administration, and Maintenance
OCH	Optical Channel
ODU	Optical Data Unit
OLP	Optical Line Protection
OLS	Optical Line System
OMS	Optical Multiplex Section
OSS	Operations Support Systems
OTN	Optical Transport Network
OTS	Optical Transmission Section
OTSi	Optical Tributary Signal
OTSiA	Optical Tributary Signal Assembly
OTSiG	Optical Tributary Signal Group
OTSiMC	Optical Tributary Signal Media Channel
OTSiMCA	Optical Tributary Signal Media Channel Assembly

ROADM	Reconfigurable Optical Add-Drop Multiplexer
SDK	Software Development Kit
SDN	Software Defined Networking
STM	Synchronous Transport Module
SIP	Service Interface Point
TAPI or T-API	Transport API Information Model
UML	Unified Modeling Language
UNI	User-Network Interface
URI	Uniform Resource Identifier
UUID	Universally Unique Identifier
WDM	Wavelength Division Multiplexing
XC	Cross-Connection

9 Individuals engaged

9.1 Editors

Arturo Mayoral López de Lerma	Telefónica
Nigel Davis	Ciena
Andrea Mazzini	Nokia

9.2 Contributors

Pedro Amaral	Infinera
Karthik Sethuraman	NEC
Malcolm Betts	ZTE
Jonathan Sadler	Infinera
Kam Lam	FiberHome
Jia Qian	ZTE

End of Document