TELECOM INFRA PROJECT®

TIP MUST Use Cases Definition Document

Optical planning

05/08/2024

Document version: v1.1

Authors

Stefan Melin
Network Architect, Telia Company
stefan.melin@teliacompany.com

Renzo Diaz
Network Architect, Telia Company
renzo.z.diaz@teliacompany.com

Anders Lindgren
Network Architect, Telia Company
anders.x.lindgren@teliacompany.com

Esther Le Rouzic
Research Engineer, Orange
esther.lerouzic@orange.com

Valéry Augais
Network Architect, Colt
valery.augais@colt.net

Dirk Breuer
Network Architect, Technology Architecture & Innovation, Deutsche Telekom AG
d.breuer@telekom.de

Steven Hill
Transport Manager: IP Access, Optical, and Fiber, MTN
steven.hill@mtn.com

José Antonio Gómez
Optical and SDN Network Architect, Vodafone
jose-a.gomez@vodafone.com

Arturo Mayoral López de Lerma
Technical Program Manager, Telecom Infra Project
amayoral@telecominfraproject.com

Contributors

Andrea D'Amico

Researcher at Nec Laboratories America, Inc.

adamico@nec-labs.com

Renato Ambrosone

PhD Student, University di Torino

renato.ambrosone@polito.it

Giacomo Borrachini

Postdoctoral Scientist, NEC Laboratories America, Inc.

gborraccini@nec-labs.com

Roberto Manzotti

Technical Leader, Cisco Systems

rmanzott@cisco.com

Gert Grammel

Solutions and Network Architect, Juniper Networks

ggrammel@juniper.net

Vittorio Curri

Professor, Politecnico di Torino

vittorio.curri@polito.it

Jan Kundrát

SDN Developer, CESNET

jan.kundrat@cesnet.cz

TIP Document License

Disclaimers

Change Tracking

Date	Revision	Author(s)	Comment
15/07/2024	v1.0	Arturo Mayoral	Final version
5/8/2024	V1.1	Noah Lavallee	Design review

Table of Contents

List of Figures

List of Tables

Introduction

Planning of optical networks is an important process for operators to be in control of cost and network quality. TIP OOPT MUST subgroup has defined the target Open Transport SDN Architecture with standard and open APIs and the TIP OOPT PSE subgroup is developing the open-source software for optical planning, GNPy, Gaussian Noise model in Python

1. Introduction

Planning of optical networks is an important process for operators to be in control of cost and network quality. TIP OOPT MUST subgroup has defined the target Open Transport SDN Architecture with standard and open APIs and the TIP OOPT PSE subgroup is developing the open-source software for optical planning, GNPy, Gaussian Noise model in Python. How online optical planning should be implemented into the MUST Open Transport SDN Architecture has not yet been defined and both OOPT subgroups collaborate with the intention to produce a common use case definition document which would shed light on how to address the optical planning problem.

The "TIP OOPT MUST Optical Whitepaper Target Architecture: Disaggregated Open Optical Networks", [2], sets the high-level description of the MUST optical architecture. The optical planning is a part of the architecture and by formulating this document we want to take the next step in the development of the Optical planning architecture.

In this document the MUST subgroup gives the background to the definition of optical planning (section 2) and describes use cases for optical planning (section 3). Due to the importance to define a solution that is aligned with the operators' expectations, including the alignment to the already defined architectures ([1],[2]) by TIP OOPT MUST optical workstream, section 4 is provided on prerequisites and considerations for the implementation of Optical planning target architecture.

Following the description of use case #1 this document may be extended with additional use cases.

1.1 Motivation and objectives

The purpose of describing the optical planning use cases is to provide a base for reviewing how the function of online optical planning should be incorporated into the TIP MUST Open Transport SDN Architecture. The scope of the optical planning is to plan the properties of Open-Optical Networks O-ON with its O-OLS (Open-Optical Line System) and connected O-OLI (Open Optical Line Interfaces). To prepare a proposal for implementation of the use case functionality, all necessary topics of relevance should be reviewed. In section 4, input is given for consideration on a solution, however this is not a complete description but the first set of important topics shared by the operators. Any other topics of relevance have to be identified during the continuing work.

A major objective is to define how the open-source library GNPy should be implemented into the SDN architecture to provide feasibility evaluation for connections. The required data exchange to/from the GNPy [3] in it's current release should be assessed to identify APIs and any required improvements of APIs to fulfill the planning function. The solution shall support planning of connections over single and multiple O-OLS domains as shown in Figure 1.

Planning of optical networks is carried out to secure network cost and performance/quality of customer connections. It is therefore of high importance for operators to be able to perform accurate optical planning and be in control of the planning process. The target is to simplify the planning process, improve accuracy and reduce time for the optical planning process.

The target architecture for online optical planning shall support that, as much as possible, data is retrieved from the network itself and data storage without having to enter data manually.

The first use case is about connection feasibility evaluation which relies on the GNPy functionality to carry out the needed simulations. The most common connections to be evaluated are over a single O-OLS. However, support for evaluation of connections passing over multiple O-OLS supplier topologies is of importance for the target solution, Figure 1.

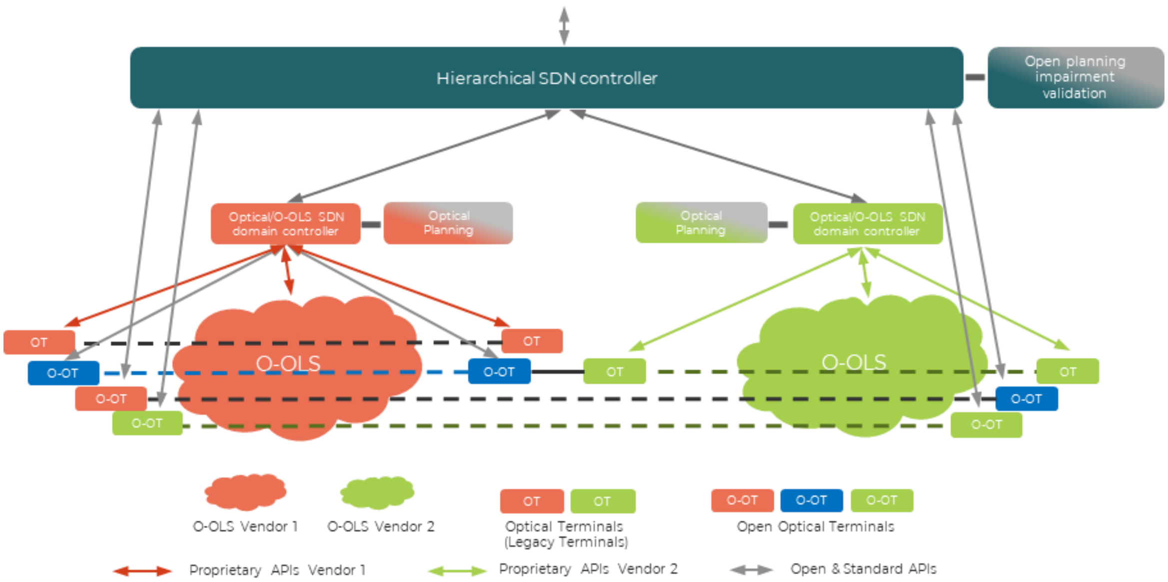

Figure 1: Example of connection passing several of O-OLS supplier domains

For terminology used in this use cases document please consider as a reference the one included in: "TIP OOPT MUST Optical Whitepaper Target Architecture: Disaggregated Open Optical Networks" [2], section 1.1.

2

Background on optical planning

In [2], the TIP OOPT MUST Target Architecture for Disaggregated Open Optical Networks is described, and its major building blocks with their function, disaggregation and openness

2. Background on optical planning

In [2], the TIP OOPT MUST Target Architecture for Disaggregated Open Optical Networks is described, and its major building blocks with their function, disaggregation and openness. The Open Planning and impairment validation, O-PaIV, is one of the building blocks being described in the SDN architecture, Figure 2.

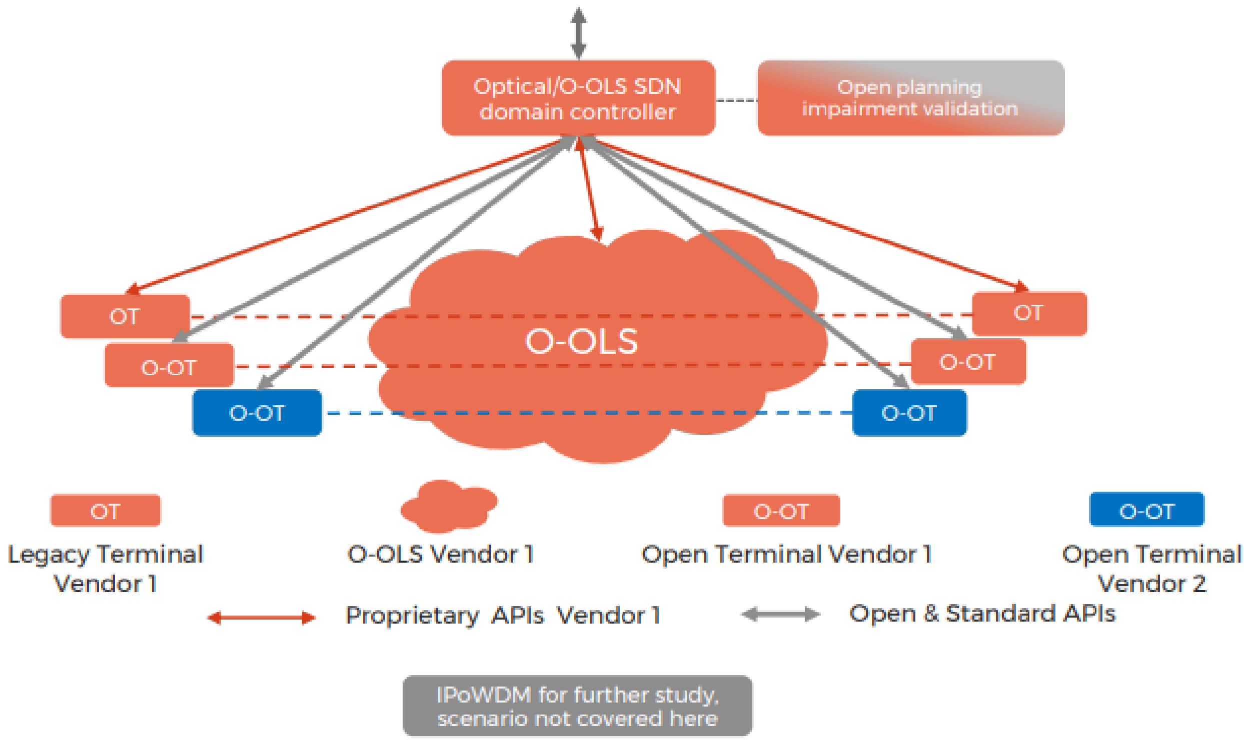

Figure 2: TIP MUST Target Open Optical Network - single domain scenario [2].

The optical planning can be divided into greenfield and brownfield planning. The greenfield planning is the design and planning of the optical line systems, OLS. The brownfield planning is the dimensioning of the connections/wavelengths and their properties over a deployed O-OLS.

2.1 Brownfield Planning

Before setting up a new optical connection we evaluate its performance based on the existing O-OLS design and connection requirements, for example bitrate, end points and diversity. The connections originate in Open Optical Line Interfaces (O-OLIs) in Optical Terminals (OTs) or pluggable optics in routers for example. An important objective of the connection planning is to understand the cost for connections. Cost for O-OTs , regeneration, spectral efficiency, space on sites and power consumption. Leaving out the planning related to sites, the optical brown field planning has following steps:

- Routing (R)
- Modulation format assignment (MA).
- Spectrum assignment (SA).
- Impairment validation (IV).

For a given route (R) the bitrate to be transported shall be matched with a modulation format assignment (MA) and baud rate to provide the desired spectral efficiency and reach versus cost. This can be achieved by evaluating different O-OLI modes that represent different bitrates, modulation formats and baud rates. Once the spectrum of the modulated signal is assigned (Spectral Assignment, SA) to a feasible consecutive spectrum slot across all hopes of the given route, the final step is to understand the performance of a selected modulation format and baud rate over a selected route we perform impairment validation (IV).

The use case is to ensure that a connection is performing without bit errors at End-of-Life or to understand the expected performance at the beginning of life at deployment time. To secure error free transmission by the End-of-Life (EoL), the EoL condition is defined by setting margins. Along the lifecycle of a system and connections, noise will accumulate and margins must be added to the Beginning-of-Life (BoL) performance for connections to secure a working connection by EoL. System margin [3] account for time-varying network operating conditions:

- Channel load, number of expected channels by the EoL per link.
- Increasing fibre losses due to splices to repair fibre cuts.

- Degrading amplifier noise factor.
- Detuning of the lasers leading to misalignment with optical filters in the intermediate nodes.
- Additional operator margin.

2.2 Greenfield planning

The greenfield planning is about designing an O-OLS based on the input of fibre data and O-OLS equipment properties. The infrastructure of sites and fibres are often a starting point for elaborating an O-OLS design with different types of ROADMs and inline amplifiers to achieve a specific performance. The performance of the O-OLS design can be measured by the GSNR [3] that can be used for benchmarking against O-OTs specification to meet required targets. It has been shown that transmission impairments in coherent systems can be approximated as Gaussian noise. The main impairments; amplified spontaneous emission (ASE) and non-linear impairments (NLI) can therefore be expressed as GSNR (Gaussian or Generalized Signal to Noise Ratio).

$$\frac{1}{GSNR} = \frac{1}{SNR_{ASE}} + \frac{1}{SNR_{NLI}}$$

Once the green field design has been determined the brown field planning can be carried out to verify connections performance to be deployed on the design.

3

Use Cases

This section will contain the set of use cases definitions proposed for open optical planning. The methodology to describe each use case may vary and different use cases may be added in different releases of the current document.

3. Use Cases

This section will contain the set of use cases definitions proposed for open optical planning. The methodology to describe each use case may vary and different use cases may be added in different releases of the current document.

3.1 Use case # 1: EoL planning of connections

3.1.1 Function

Estimation of End-of-Life performance of a candidate transceiver (not installed) on an existing network, a case of brown field planning according to section 2.1. The use case environment is partially disaggregated open optical networks described in [2]. It is of importance to have high accuracy EoL impairment validations to have good alignment between reality and planned connections and thereby secure the connections quality over time and to have accurate cost estimations.

3.1.2 Purpose

The purpose of the use case is to determine the required equipment (mainly the O-OTs) needed to realize connections. To determine which equipment delivers the expected performance, brown field planning is carried out. The planning supports different activities for different purposes such as Capacity planning, Deployment planning and Connection analysis. Each activity relies on understanding the End-of-Life performance of connections for different purposes that can be converted into a cost. See section 3.1.3 for a summary of the activities.

3.1.2.1 Capacity planning

Capacity planning is prepared as input to a global budget and strategy for the coming periods of deployment accounting for traffic volume forecasts. It is based on the current state of the network and already programmed deployment for which the planning team elaborates different scenarios for the next months or year. The forecasted connections are evaluated by performing optical brown field planning to understand the feasibility and number of terminals, regenerations or interfaces required. The EoL impairment validation is performed to get as accurate estimation of costs as possible. The expected result is a volume of EoL feasible connections on the optical layer (IP and OTN is of interest as well but out of scope in this use case) that will serve as input to a global budget and strategy.

3.1.2.2 Deployment planning

When the network planning team receives a request for one or more new connections, deployment planning is initiated. Deployment planning is prepared for a for one or several candidate connections. The scope of planning might be only the optical part or IP and OTN part also. In this use case we focus on optical planning. The planning teams elaborate a deployment project to increase capacity on targeted links/nodes, to replace equipment or to add protection for example. Optical brown field planning is used as a tool to verify feasibility/performance of possible terminals including regenerations to verify deployment proposals. The accuracy of the EoL impairment validation is of high importance to have cost optimized and reliable connections that will provide error free transmission over its life cycle. The expected result is a design or a simulation propagation for one or several candidate scenarios before discussion with the supplier for a quotation.

3.1.2.3 Connection analysis

Connection analysis is prepared by the planning team to review proposals (NE data in GNPy format). It can be prepared by suppliers for a given deployment received in an RFI for example. The expected result is an analysis (compliancy to constraints, design and feasibility estimation) of the supplier’s scenario. The scope of the planning might be only the optical connection or IP and OTN part also. In this use case the focus is on the optical connection.

Activity	Capacity planning	Deployment planning	Connection analysis
Purpose	Prepare input to budget	Prepare connection design before deployment. Basis for quotation.	Investigation of existing or planned connections, RFI evaluation.
Input information	Forecasts on traffic growth, network data	Capacity demands	Connection characteristics
Input data	Network data (not on the day collected data needed). Ageing margins. Projections of O-OLS changes over time.	Actual network data and possibly off-line data on new NE. Ageing margins.	Actual network data and off-line data on new NE. Ageing margins.
Accuracy	Medium accuracy for budgeting.	High accuracy for reliable connections and cost calculation.	High accuracy for reliable connections and cost calculation.
Result	A volume of EoL feasible connections for cost estimation.	Design or a simulation propagation for one or several candidate scenarios before discussion with supplier for quotation.	Feasibility evaluation with details on performance and design.

3.1.3 Summary of activities of the use case.

Table 1. Summary of use case #1 activities.

3.1.4 Dependencies and limitations to the scope

The actual deployment, commissioning, of a new connection is not considered in this use case. When a connection planning is ready for deployment another procedure/architecture will manage the deployment. The information and data that has been determined during the planning procedure and is available in the APM (Agnostic Planning Module defined in section 4) can be shared within the SDN architecture to reduce manual intervention for faster and less error prone deployment.

4

Prerequisites and considerations for the implementation

In this section we have summarized architectural considerations that are of importance for the target optical planning architecture. To support the presented and future planning use cases an architecture for optical planning should be defined within the MUST SDN target architecture. ...

4. Prerequisites and considerations for the implementation of Optical planning target architecture

In this section we have summarized architectural considerations that are of importance for the target optical planning architecture. Note that this section is not aimed to describe a complete target architecture but rather highlighting important concepts defined by the operators. To support the presented and future planning use cases an architecture for optical planning should be defined within the MUST SDN target architecture. The architecture shall be based on TIP OOPT MUST already agreed APIs as far as possible and the open-source planning engine GNPy developed by the TIP OOPT PSE team. To set an architecture for the planning we propose to introduce a module responsible for the planning, an Agnostic Planning Module, APM. The APM would provide a GUI for the planning teams to work in and it would be the interface to be used for coordinating:

- Routing.
- Modulation format assignment.
- Spectrum assignment.
- Impairment validation (delegated to GNPy)

See Figure 3 for proposal on architecture and how the APM connects to the GNPy and the Optical SDN Controller. The APM together with GNPy would form the O-PaIV, [2].

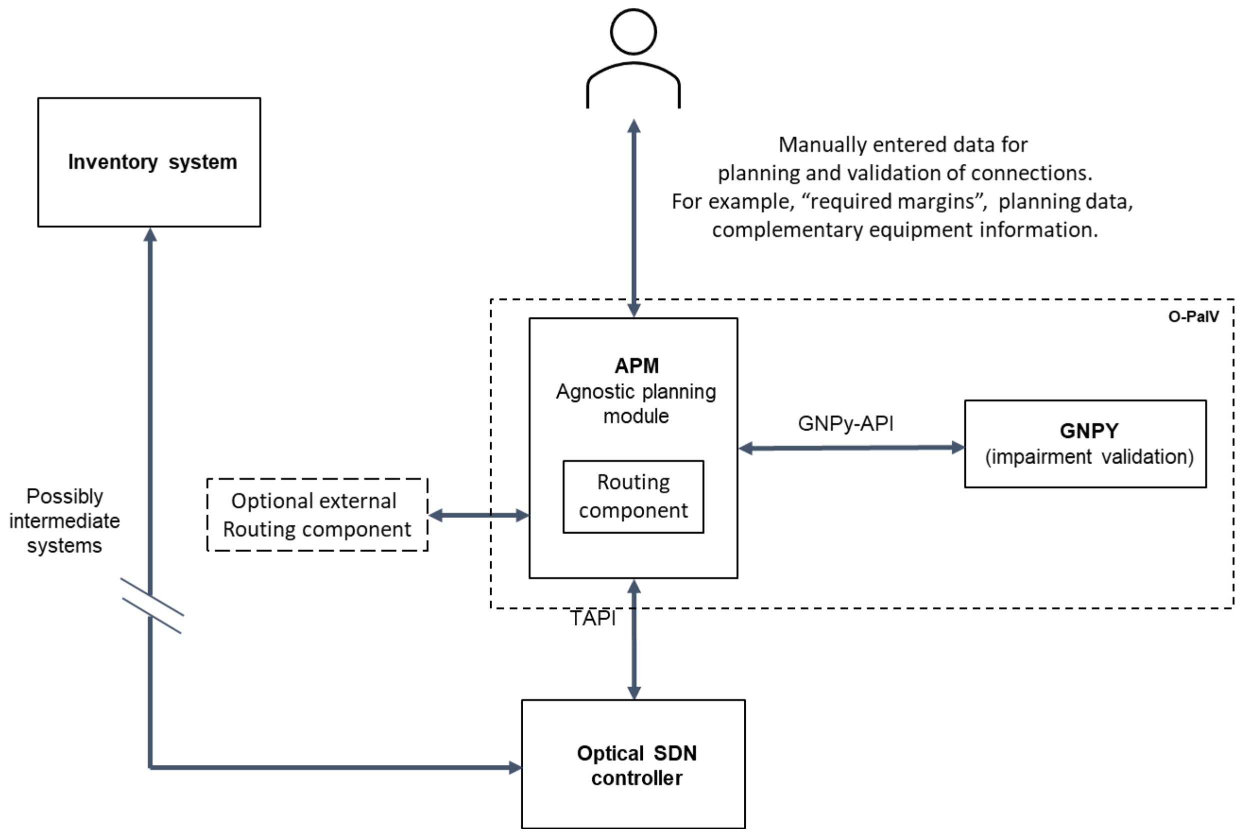

Figure 3: Architecture proposal for optical planning function

The APM would be responsible for collecting the data needed for the planning and to have a library for the topology and equipment models. The library in the APM is updated by having connection to the O-SDN-C (Optical SDN controller) via TAPI that is the target API for network management in the MUST SDN architecture. Between the APM and the GNPy the GNPy-API is proposed due to the need for support for network topology data and simulation data.

As proposed, the APM is responsible for the orchestration of the optical planning. The routing is performed by the APM by using an internal routing component that is considered to be the mandatory implementation [4]. The routing can be performed by the GNPy or even an optional external routing component. To avoid setting operator specific or complicated development requirements on routing constraints and routing functions on the GNPy, the development is preferred to be focused on the APM internal routing component. For the optional external routing function, the API would need to be defined.

The impairment validation is delegated to the GNPy. The GNPy library [4] is housed by the APM. The APM collects information according to the planning team requirements, link and O-OLI data, and sends it to the GNPy for an estimation of the performance. The GNPy responds to the APM with an OSNR/GSNR value. The GNPy module can receive multiple optical service's impairment validation requests in one single API call and each of the services can have a single explicit candidate path constraint within. The reply by GNPy on the quality per connection has to be evaluated in/by the APM module for selection of desired path for a connection.

To simplify (avoid errors that occur due to manually entering of data) and improve accuracy of impairment validation and the EoL planning it is desirable to read all possible data from the network for brown field planning. However, there is data that is not available per se in the network, for example noise figures, fibre data and new O-OLI equipment models. Fibre data (for example fibre type, PMD, PDL and effective_area) could potentially be available online from inventory system via the O-SDN-C. Equipment models for new O-OLI that are off-line, not present in the network can be stored/uploaded locally to the APM library. As a short-term solution option, the fibre data can be added manually to the APM library.

By having the possibility to populate the GNPy request from the APM with data with different origins, it would be preferable if data can be traced in the results of different simulations. To be able to validate the result of impairment validation would further improve the reliability of the planning work.

As a long term goal the TIP OOPT can define and propose to the industry how to make missing on-line data today available on-line for example noise figures.

In the text following definitions are used for on-line and off-line data:

- Online – Has connectivity and retrieves data from other units. For example, data is used by the APM that is stored on the O-SDN-C.
- Offline – No connectivity. For example, data is stored locally for computation.

In the target architecture, most data are available in the network via the O-SDN-C, static and dynamic data.

- Static data – Data that does not change.
- Dynamic data – Data that changes over time.

4.1 Margins to be considered for connection EoL planning

In the use case #1 the aim is to ensure that the computed planning remains effective throughout the lifespan of the network. To achieve this, it is necessary to identify all the factors that may impact performance and evolve over time. These variations are typically accounted for by including additional margins, as discussed in the article [5].

The planning margins encompass various factors such as equipment and fibre infrastructure ageing, industrial process accuracy, system load, and fast time-varying impairments. They also consider potential inaccuracies in data or planning tools, as well as the need for margin reservation by the operator.

Here are some examples of these margins:

- **Industrial margin**: Variations in parameters such as minimum required OSNR of transceiver or noise figure of amplifiers are taken into account. Actual performance must be better than the specified values based on measurement samples.
- **Ageing margin**: The ageing margin can be divided into equipment ageing margin and fibre ageing margin. Expected degradation of parameters like minimum Q factor for transceivers or saturation power of amplifiers over time is considered. Fibre infrastructure may also experience increased total loss and changes in loss distribution due to ageing caused by cuts and repairs.
- **System load:** The addition of multiple channels in the network affects both new and existing channel performances. Full load assumption is commonly used to cover worst-case scenarios, especially for non-linear impairments and crosstalk effects when crossing ROADMs.
- **Fast time varying effects:** Impairments caused by time-varying factors like polarization mode dispersion or polarization-dependent loss or gain are taken into account.

GNPy can incorporate most of these margins, including equipment performance definitions.

- Non-linear effects are simulated under full load conditions. In cases where channels with different baud rates are mixed, defining full load may be challenging, but the impact of the channel distribution can be considered insignificant if their power is close to optimal, as mentioned in the article [6].
- The exact fibre layout with lumped losses can be evaluated, and an additional loss can be included to account for fibre repair loss margin.
- Polarization mode dispersion (PMD) and polarization-dependent loss (PDL) are considered as additional penalties on computed GSNR.
- Filtering and crosstalk effects are not yet included in GNPy.

There are margins that currently not are managed by the GNPy, for example the equipment ageing margins and margins for fast time varying effects. The equipment ageing margins are known by the network element suppliers however they are not based on a common definition to be used in simulations. The fast time varying effects are also missing a general definition for modelling in simulations.

The GNPy will perform simulations based on the input data that is provided, including the required margin which is assumed to be part of the data that is provided to GNPy.

4.2 Workflow and exchange of data for connection EoL planning

The task of the GNPy in use case #1 is path feasibility estimation. Following data is delegated by the agnostic planning module to GNPy:

- OLS network data.
- Connection data.
- Simulation input.

The output data from the GNPY is the result of the connection feasibility simulation in GNPy format.

Current format is a JSON file with the following structure [structure from draft-ietf-teas-yang-path-computation-01.txt]. Here are one sample for requests (showing several candidate requests with different routes for the same service) and an extract of the response output:


```
{
  "path-request": [
    {
      "request-id": "0-a",
      "source": "trx Brest_KLA",
      "destination": "trx Vannes_KBE",
      "src-tp-id": "trx Brest_KLA",
      "dst-tp-id": "trx Vannes_KBE",
      "bidirectional": false,
      "path-constraints": {
        "te-bandwidth": {
          "technology": "flexi-grid",
          "trx_type": "Voyager",
          "spacing": 50000000000.0,
          "max-nb-of-channel": 80,
          "output-power": 0.0012589254117941673,
          "path_bandwidth": 100000000000.0
        }
      },
      "explicit-route-objects": {
        "route-object-include-exclude": [
          {
            "explicit-route-usage": "route-include-ero",
            "index": 0,
            "num-unnum-hop": {
              "node-id": "west edfa in Quimper",
              "link-tp-id": "link-tp-id is not used",
              "hop-type": "LOOSE"
            }
          }
            ]
        }
    },
    {
      "request-id": "0-b",
      "source": "trx Brest_KLA",
      "destination": "trx Vannes_KBE",
      "src-tp-id": "trx Brest_KLA",
      "dst-tp-id": "trx Vannes_KBE",
      "bidirectional": false,
      "path-constraints": {
        "te-bandwidth": {
          "technology": "flexi-grid",
          "trx_type": "Voyager",
          "spacing": 50000000000.0,
          "max-nb-of-channel": 80,
          "output-power": 0.0012589254117941673,
          "path_bandwidth": 100000000000.0
        }
```



```
      },
      "explicit-route-objects": {
        "route-object-include-exclude": [
          {
            "explicit-route-usage": "route-include-ero",
            "index": 0,
            "num-unnum-hop": {
              "node-id": "roadm Rennes_STA",
              "link-tp-id": "link-tp-id is not used",
              "hop-type": "LOOSE"
            }
          }
            ]
        }
    },
    {
      "request-id": "0-c",
      "source": "trx Brest_KLA",
      "destination": "trx Vannes_KBE",
      "src-tp-id": "trx Brest_KLA",
      "dst-tp-id": "trx Vannes_KBE",
      "bidirectional": false,
      "path-constraints": {
        "te-bandwidth": {
          "technology": "flexi-grid",
          "trx_type": "Voyager",
          "trx_mode": "mode 1",
          "spacing": 50000000000.0,
          "output-power": 0.0012589254117941673,
          "path_bandwidth": 100000000000.0
        }
      },
      "explicit-route-objects": {
        "route-object-include-exclude": [
          {
            "explicit-route-usage": "route-include-ero",
            "index": 0,
            "num-unnum-hop": {
              "node-id": "roadm Brest_KLA",
              "link-tp-id": "link-tp-id is not used",
              "hop-type": "LOOSE"
            }
          },
          {
            "explicit-route-usage": "route-include-ero",
            "index": 1,
            "num-unnum-hop": {
              "node-id": "roadm Lannion_CAS",
              "link-tp-id": "link-tp-id is not used",
              "hop-type": "LOOSE"
            }
          },
          {
```

```
            "explicit-route-usage": "route-include-ero",
            "index": 2,
            "num-unnum-hop": {
              "node-id": "roadm Lorient_KMA",
              "link-tp-id": "link-tp-id is not used",
              "hop-type": "LOOSE"
            }
          },
          {
            "explicit-route-usage": "route-include-ero",
            "index": 3,
            "num-unnum-hop": {
              "node-id": "roadm Vannes_KBE",
              "link-tp-id": "link-tp-id is not used",
              "hop-type": "LOOSE"
            }
          }
        ]
      }
    }
  ]
}
```


Figure 4a: Example of GNPy input requests (JSON file).


```
{
  "response": [
    {
      "response-id": "0",     # same reference as the formulated request
      "path-properties": {
        "path-metric": [
          {
            "metric-type": "SNR-bandwidth",
            "accumulative-value": 26.75     # in dB@signal baudrate bandwidth
          },
          {
            "metric-type": "SNR-0.1nm",
            "accumulative-value": 30.84     # in dB@0.1nm
          },
          {
            "metric-type": "OSNR-bandwidth",
            "accumulative-value": 26.76    # in dB@signal baudrate bandwidth
          },
          {
            "metric-type": "OSNR-0.1nm",
            "accumulative-value": 30.84    # in dB@0.1nm
          },
          {
            "metric-type": "reference_power",
```

```
            "accumulative-value": 0.001    # reference channel power in Watt
          },
          {
            "metric-type": "path_bandwidth",
            "accumulative-value": 100000000000.0  # requested capacity in bit/s
          }
        ],
        "path-route-objects": [   # list of crossed elements in the topology,
with the occupied spectrum slot and the tuned mode on transceiver
          {
            "path-route-object": {
              "index": 0,
              "num-unnum-hop": {
                "node-id": "trx Lorient_KMA",
                "link-tp-id": "trx Lorient_KMA"
              }
            }
          },
          {
            "path-route-object": {
              "index": 1,
              "label-hop": [
                {
                  "N": -284,
                  "M": 4
                }
              ]
            }
          },
          {
            "path-route-object": {
              "index": 2,
              "transponder": {
                "transponder-type": "Voyager",
                "transponder-mode": "mode 1"
              }
            }
          },
          {
            "path-route-object": {
              "index": 3,
              "num-unnum-hop": {
                "node-id": "roadm Lorient_KMA",
                "link-tp-id": "roadm Lorient_KMA"
              }
            }
….
```


Figure 4b: Example of GNPy output (JSON file).

Per channel performance estimation and per element settings and power levels can be assessed via the command line interface and standard output (not yet in the json export).

```
There are 76 channels propagating
Power mode is set to True
=> it can be modified in eqpt_config.json - Span

There are 1 fiber spans over 80 km between Site_A and Site_B

Now propagating between Site_A and Site_B:

Propagating with input power = [1;36;40m0.00 dBm[0m:
Transceiver Site_A
  GSNR (0.1nm, dB):          40.00
  GSNR (signal bw, dB):      35.92
  OSNR ASE (0.1nm, dB):      40.00
  OSNR ASE (signal bw, dB):  35.92
  CD (ps/nm):                0.00
  PMD (ps):                  0.00
  PDL (dB):                  0.00
  Latency (ms):              0.00
Fiber          Span1
  type_variety:                SSMF
  length (km):                 80.00
  pad att_in (dB):             0.00
  total loss (dB):             17.00
  (includes conn loss (dB) in: 0.50 out: 0.50)
  (conn loss out includes EOL margin defined in eqpt_config.json)
  reference pch out (dBm):     -17.00
  actual pch out (dBm):        -17.00
Edfa Edfa1
  type_variety:           std_low_gain
  effective gain(dB):     15.00
  (before att_in and before output VOA)
  noise figure (dB):      6.62
  (including att_in)
  pad att_in (dB):        0.00
  Power In (dBm):         1.81
  Power Out (dBm):        16.82
  Delta_P (dB):           -2.00
  target pch (dBm):       -2.00
  actual pch out (dBm):   -1.99
  output VOA (dB):        0.00
Transceiver Site_B
  GSNR (0.1nm, dB):          31.18
  GSNR (signal bw, dB):      27.10
  OSNR ASE (0.1nm, dB):      33.30
  OSNR ASE (signal bw, dB):  29.21
  CD (ps/nm):                1336.00
  PMD (ps):                  0.36
  PDL (dB):                  0.00
  Latency (ms):              0.39

Transmission result for input power = 0.00 dBm:
  Final GSNR (0.1 nm): [1;36;40m31.18 dB[0m

The GSNR per channel at the end of the line is:
Ch. #   Channel frequency (THz)      Channel power (dBm)    OSNR ASE (signal bw, dB)     SNR NLI
(signal bw, dB)        GSNR (signal bw, dB)
    1             191.35000                     -2.00                       29.25
32.89                   27.69
    2             191.40000                     -2.00                       29.25
32.33                   27.51
```

```
    3                 191.45000                     -2.00                       29.25
32.08                       27.43
    4                 191.50000                     -2.00                       29.24
31.92                       27.37
    5                 191.55000                     -2.00                       29.24
31.81                       27.33
    6                 191.60000                     -2.00                       29.24
31.72                       27.30

…

   72                 194.90000                     -2.00                       29.18
31.43                       27.15
   73                 194.95000                     -2.00                       29.18
31.53                       27.19
   74                 195.00000                     -2.00                       29.18
31.68                       27.25
   75                 195.05000                     -2.00                       29.18
31.92                       27.33
   76                 195.10000                     -2.00                       29.18
32.47                       27.51

(No source node specified: picked Site_A)

(No destination node specified: picked Site_B)
```

Figure 5a: Example of GNPy output with detailed results for one candidate (CLI).

```
request 0-a
        Computing path from trx Brest_KLA to trx Vannes_KBE
        with path constraint: ['trx Brest_KLA', 'west edfa in Quimper', 'trx Vannes_KBE']
INFO:gnpy.topology.request:     Computed path (roadms):['roadm Brest_KLA', 'roadm Lorient_KMA',
'roadm Vannes_KBE']

INFO:gnpy.topology.request:
request 0-b
        Computing path from trx Brest_KLA to trx Vannes_KBE
        with path constraint: ['trx Brest_KLA', 'roadm Rennes_STA', 'trx Vannes_KBE']
INFO:gnpy.topology.request:     Computed path (roadms):['roadm Brest_KLA', 'roadm Lannion_CAS',
'roadm Rennes_STA', 'roadm Vannes_KBE']

INFO:gnpy.topology.request:
request 0-c
        Computing path from trx Brest_KLA to trx Vannes_KBE
        with path constraint: ['trx Brest_KLA', 'roadm Brest_KLA', 'roadm Lannion_CAS', 'roadm
Lorient_KMA', 'roadm Vannes_KBE', 'trx Vannes_KBE']
INFO:gnpy.topology.request:     Computed path (roadms):['roadm Brest_KLA', 'roadm Lannion_CAS',
'roadm Lorient_KMA', 'roadm Vannes_KBE']

Result summary
req id   demand                             GSNR@bandwidth A-Z (Z-A) GSNR@0.1nm A-Z (Z-A)  Receiver
minOSNR       mode          Gbit/s     nb of tsp pairsN,M or blocking reason
0-a      trx Brest_KLA to trx Vannes_KBE :       22.44         26.52            14           mode 1
100.0            1          ([-284],[4])
0-b      trx Brest_KLA to trx Vannes_KBE :       19.56         23.65            14           mode 1
100.0            1          ([-284],[4])
0-c      trx Brest_KLA to trx Vannes_KBE :       20.51         24.59            14           mode 1
100.0            1          ([-276],[4])
```

Figure 5b: Example of GNPy output with results for three candidates (CLI).

In the use case #1 planning scenarios, the impairments are considered for the End-of-Life of the system, assuming full load (currently defined equal for all links), and with the required margins according to the operator requirements. A planning activity is carried out by a team that use the APM module to model the brown field planning following the steps 1-7 as shown in Figure 6.

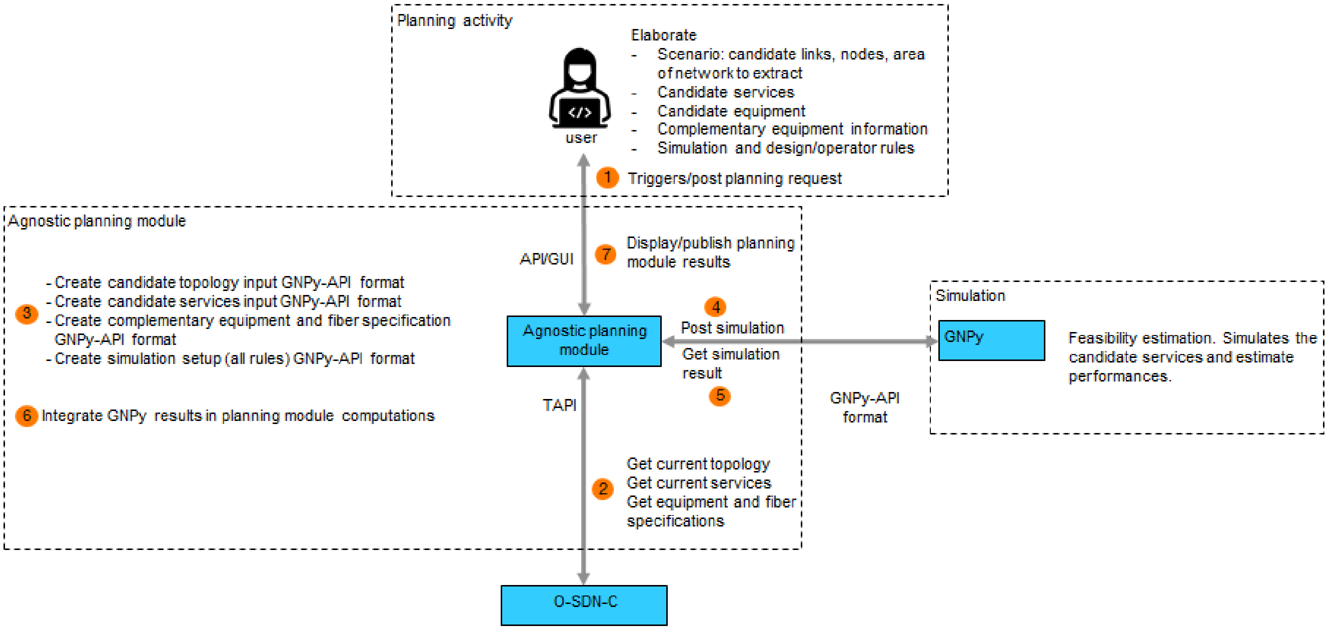

Figure 6: Online planning data exchange.

The data to be exchanged between APM and GNPy would be as follows [7]:

- **Current and candidate topology**: Detailed current topology including fibre spans type variety, length, and loss (with concentrated loss details), amplifier type variety and settings (gain, tilt, variable Optical Attenuator-VOA and nominal channel power targets[1]), ROADM equalization type and power targets; and candidate links and nodes in this topology.
- **Channel map**, target spectrum occupation for the given topology.
- **Candidate services:** Detailed services to be tested with their constraints (this can be routing, type of transceiver, mode of the transceiver, power targets, maximum number of channels expected in the network, spectrum assignment) in addition to the actual installed set of services (with their path, transceiver type, mode, and spectrum assignment)

[1] Currently performance data such as noise figure and PDL is retrieved from the library but they could also be exposed in the topology.

- **Equipment library:** Detailed specification of each equipment type variety (including Noise Figure of the amplifiers).
- **Fiber infrastructure**: Detailed specification of fibres deployed for the considered infrastructure.
- **Design and operator rules:** Detailed simulation set up used to describe both the design (engineering) rules of the set up (power optimization per span, particular usage of in/out VOA…) and the operator constraints (constraints on the amplifiers usage, the ROADM, the margins, …).
- **Numeric simulation rules**: Detailed simulation configurations, for example the type of Gaussian Noise model (GGN, analytic, etc) .

The information is provided to GNPy in the format of JSON inputs on its API and this workflow aims at identifying the different data flows and the data models for the target scenario. We expect that the required information to be retrieved and provided to **GNPy in this way:**

- **Current topology**: extracted northbound from the Optical SDN Controller via TAPI NBI via the agnostic planning module.
- **Candidate topology:** created by the agnostic planning module with the candidate link and nodes if any and merged with the current topology. We propose the format between the agnostic planning module and GNPy to be GNPy-API maintained by the TIP OOPT PSE team.
- **Candidate services** (path, transceiver, modes, power, number of channels, [6]: similarly, the current set of services is retrieved northbound from the Optical SDN Controller via TAPI NBI, and the candidate services (including all, part or none of the existing services) should be constructed by the agnostic planning module based on user scenario. We propose the format between the agnostic planning module and GNPy to be GNPy-API maintained by the TIP OOPT PSE team.
- **Equipment library**: The target is to retrieve this information northbound from the Optical SDN Controller via TAPI NBI through the "profiles". We propose the format between the agnostic planning module and GNPy to be GNPy-API maintained by the TIP OOPT PSE team. However, this use case also onboards candidate path, links or nodes, whose equipment might not yet be available from the network, since these could be candidate equipment. In this case the

target format is the GNPy equipment library format, and this should be built offline by the user.

- **Fiber infrastructure:** Fibre infrastructure is usually not an equipment managed by an SDN Controller, however TIP OOPT MUST has mandated that this data be also available in the SDN Controller. So, it is expected that deployed infrastructure information be extracted northbound from the Optical SDN Controller via TAPI NBI as part of the topology and profiles for fibre types or concentrated losses.
- **Design rules**, operator rules, and numeric simulation rules are not considered as part of the SDN Controller scope. So, this information should be elaborated by the user based on its scenario in the APM. The target format is the GNPy equipment library format, and this should be built offline by the user.

5

Summary

To plan and estimate quality of optical connections before deployment is an important procedure for operators to be in control of costs. In this document operators in the TIP OOPT MUST optical works stream are defining use cases for optical planning to be used as input to agree on how they can be supported and required in the TIP OOPT MUST Optical target architecture...

5. Summary

To plan and estimate quality of optical connections before deployment is an important procedure for operators to be in control of costs. In this document operators in the TIP OOPT MUST optical workstream are defining use cases for optical planning to be used as input to agree on how they can be supported and required in the TIP OOPT MUST Optical target architecture. The first described use case is about connection planning where the estimation of End-of-Life performance is a vital step to verify that the quality is secured during its life cycle. The use case covers several activities such as, provide basis for forecasting, planning for deployment of new connections and analysis of future or deployed connections.

The solution that is sought after is expected to be integrated into the TIP OOPT MUST Optical target architecture which is adopting open, standardized and supplier independent protocols. The operators have described concepts in section 4 that are considered of importance for the solution to be adapted to their way of working. An agnostic planning module is proposed to be in control of the optical planning and that would delegate the impairment validation to the open-source planning module GNPy.

6 References

[1]. "Open Transport SDN Architecture Whitepaper"
https://cdn.mediavalet.com/usva/telecominfraproject/03V-53HVHE2_sr3_nk47_Q/WPd6tLiuS0CDkcG5S6Etug/Original/OpenTransportArchitecture-Whitepaper_TIP_Final.pdf

[2]."TIP OOPT MUST Optical Whitepaper Target Architecture: Disaggregated Open Optical Networks"
https://cdn.mediavalet.com/usva/telecominfraproject/VdNAxFBh8UKdTtvU_OOU2A/ooMkKLHdL0i-qq-fgsIxvQ/Original/TIP_OOPT_MUST_Optical_Whitepaper_Target_Architecture_-_Disaggregated_Open_Optical_Networks__v10_-_GREENPUBLIC_ACCESS.pdf

[3].GNPy: Optical Route Planning Library
https://github.com/Telecominfraproject/oopt-gnpy/tree/master/docs

[4]. Physical Model used in GNPy
https://gnpy.readthedocs.io/en/master/model.html

[5]."Design of low-margin optical networks", Y. Pointurier, Journal of Optical Communications and Networking, vol. 9, no. 1, pp. A9-A17, Jan. 2017, doi: 10.1364/JOCN.9.0000A9.]).
https://opg.optica.org/jocn/abstract.cfm?uri=jocn-9-1-A9

[6]."Experimental validation of GNPy in a multi-vendor flex-grid flex-rate WDM optical transport scenario", A. D'Amico et al, Journal of Optical Communications and Networking, vol. 14, no. 3, pp. 79-88, March 2022, doi: 10.1364/JOCN.442208.].
https://ieeexplore.ieee.org/document/9662659

[7].GNPy: JSON Input Files
https://gnpy.readthedocs.io/en/master/json.html