

TeraFlow
SDN
by ETSI

Int5Gent+ALLEGRO+SEASON Multi-layer slice controller architecture

Pablo Armingol Robles

Telefónica I+D

18/10/2023

Agenda

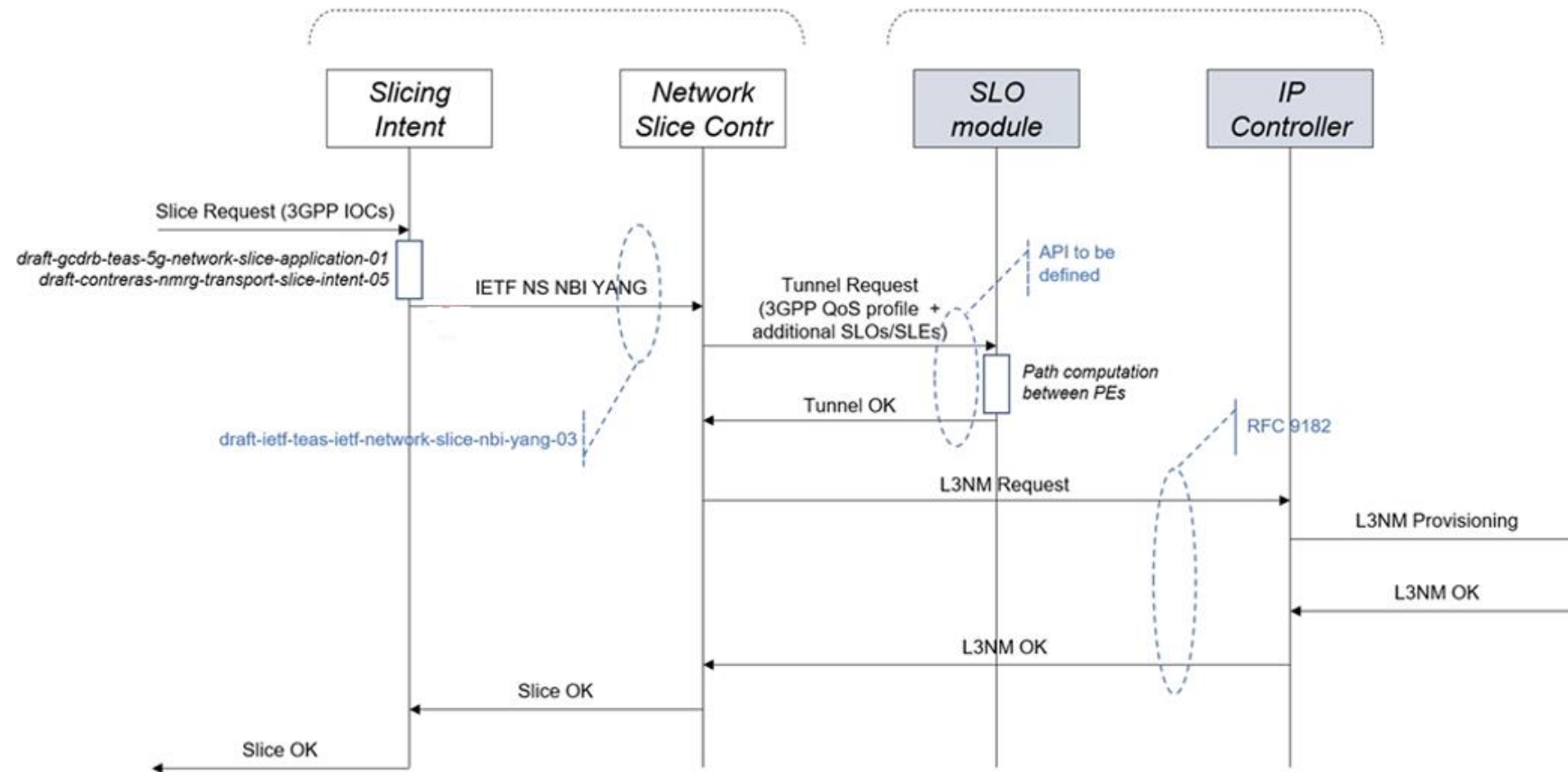
- General Architecture
- Main modules
- Request process
- Functional Architecture

General Architecture

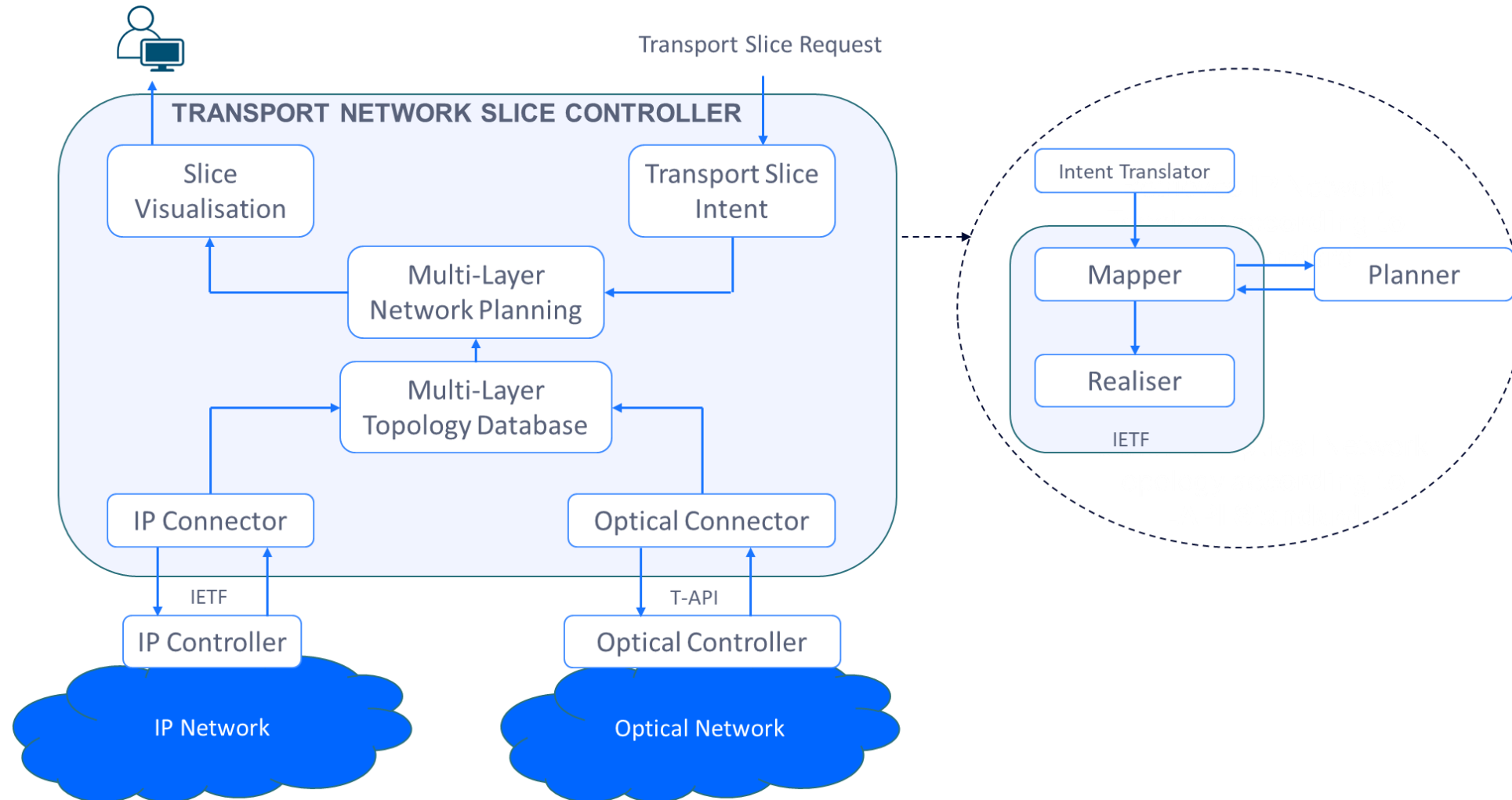
Introduction

- 3GPP does not specify slicing capabilities in the Transport Network.
- IETF is working on the definition of what is called IETF Network Slices, that essentially describes how to request and realized network slices requested by 3GPP systems with IETF-based technologies.
- What do you need to fulfil the gap in transport networks?
 - Connecting 3GPP slices through IETF Network Slice services.
 - Mapping process between 3GPP and IETF network slices.
 - Multilayer Path Computation enabling transport resources allocation for each transport slice
 - Vendor agnostic Multilayer Topology and Inventory database
- Telefonica is prototyping and demonstrating this architecture in the ETSI TeraFlow SDN framework

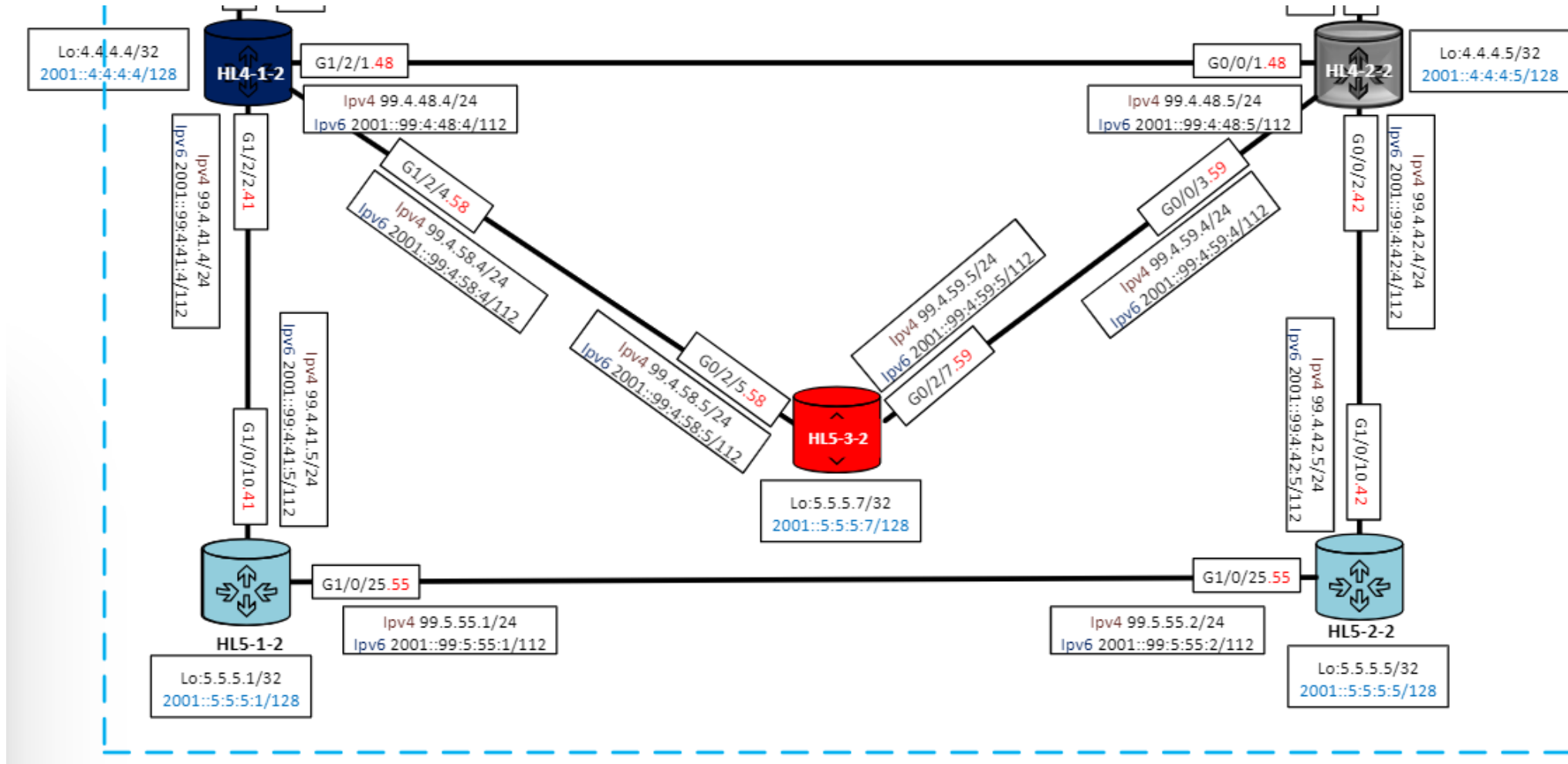
Transport Slice Workflow



General Architecture



Test bed



Main modules

Transport Slice Intents

Mapper:

1. Receives the IETF Network Slice request
2. Does it fit?
3. Performance Notification (if needed)

Realizer:

1. Receives provisioning requests.
2. Determines how to realize it.
3. Requests it to the required Controller.

Needs a Translation
3GPP → IETF

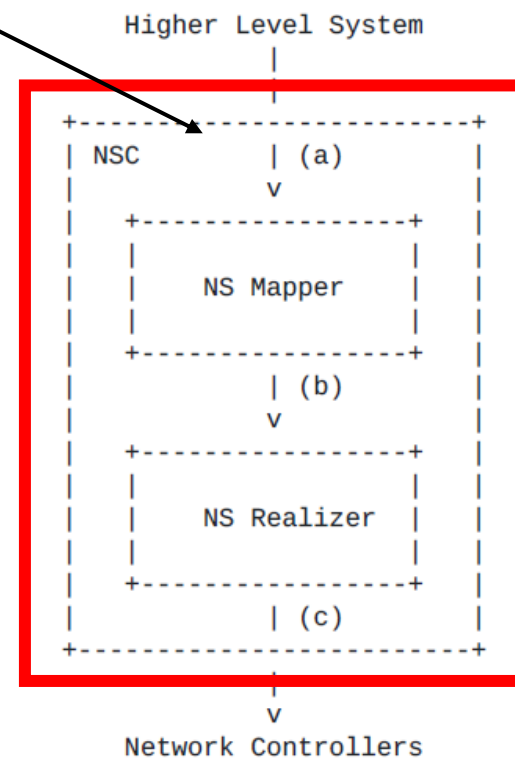


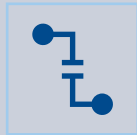
Figure 2: IETF Network Slice Controller structure and associated data models

draft-contreras-teas-slice-controller-models-05

Multilayer Network Planning (Planner)



Focus on multi-layer path computation using Segment Routing (SR) technology.



SR enables efficient Traffic Engineering (TE) and path calculation through predefined segments or labels.



Exploration of AI techniques in segment routing for optimized path computation.

Teraflow SDN Controller

Multilayer topology DB

- Stores Topology information and shares it with the Planner.
- Updated with network modifications.

Slice Creation

- Zero-touch automation paradigm.
- Transforms SLA intents into L3VPN connections.
- Interacts with both IP and Optical controllers to enable the multilayer capability.

Request process

3GPP request

```
1 {
2   "Nsi":"6075934279a17d12456d19ac",
3   "ServiceProfile":{
4     "SNSSAIList":null,
5     "PLMNIdList":null,
6     "MaxNumberOfUEs":0,
7     "CoverageArea": "",
8     "Latency":0,
9     "DLThptPerSlice":{
10      "GuaThpt":0,
11      "MaxThpt":0
12    },
13    "DLThptPerUE":{
14      "GuaThpt":0,
15      "MaxThpt":0
16    },
17    "ULThptPerSlice":{
18      "GuaThpt":0,
19      "MaxThpt":0
20    },
21    "ULThptPerUE":{
22      "GuaThpt":0,
23      "MaxThpt":0
24    }
25  },
26  "NetworkSliceSubnet":{
27    "SliceProfile":{
28      "SNSSAIList":null,
29      "PLMNIdList":null,
30      "PerfReq":{
31        "PerfReqEmbbList":{
32          "ExpDataRateDL":0,
33          "ExpDataRateUL":0,
34          "AreaTrafficCapDL":0,
35          "AreaTrafficCapUL":0,
36          "UserDensity":0,
37          "ActivityFactor":0
38        },
```

```
39      "PerfReqUrllcList":{
40        "CSAvailabilityTarget":0,
41        "CSReliabilityMeanTime":0,
42        "ExpDataRateDL":0,
43        "MsgSizeByte":""
44      }
45    },
46    "CNSliceSubnetProfile":{
47      "MaxNumberOfUEs":0,
48      "Latency":0,
49      "DLThptPerSliceSubnet":{
50        "GuaThpt":0,
51        "MaxThpt":0
52      },
53      "DLThptPerUEPerSubnet":{
54        "GuaThpt":0,
55        "MaxThpt":0
56      },
57      "ULThptPerSlicePerSubnet":{
58        "GuaThpt":0,
59        "MaxThpt":0
60      },
61      "ULThptPerUEPerSubnet":{
62        "GuaThpt":0,
63        "MaxThpt":0
64      },
65      "MaxNumberOfPDUSessions":0,
66      "CoverageAreaTAList":{
67        "TAC":null
68      }
69    },
70    "RANSliceSubnetProfile":{
71      "CoverageAreaTAList":{
72        "TAC":null
73      },
74      "UEMobilityLevel": "",
75      "ResourceSharingLevel":0,
76      "MaxNumberOfUEs":0,
77      "ActivityFactor":0,
78      "DLThptPerUEPerSubnet":{
79        "GuaThpt":0,
```

```
80      "MaxThpt":0
81    },
82    "ULThptPerUEPerSubnet":{
83      "GuaThpt":0,
84      "MaxThpt":0
85    },
86    "UESpeed":0
87  },
88  "TopSliceSubnetProfile":""
89  },
90  "ManagedFunction":""
91  "EpTransport":[
92    {
93      "IpAddress":"1.1.1.1/24",
94      "LogicInterfaceType": "tcp-ip",
95      "LogicInterfaceId": "GigabitEthernet0/0/1.56",
96      "NextHopInfo": "98.4.56.1",
97      "QosProfile": "6000",
98      "EpApplicationRef": "EP_F1U CU-UP1"
99    },
100   {
101     "IpAddress":"1.1.1.1/24",
102     "LogicInterfaceType": "tcp-ip",
103     "LogicInterfaceId": "GigabitEthernet2/2/3.43",
104     "NextHopInfo": "98.5.54.2",
105     "QosProfile": "",
106     "EpApplicationRef": "EP_F2U CU-UP2"
107   }
108 ]
109 }
110 }
```

Transport metrics

Planner

NSC to planner:

```
ubuntu@tfcontroller:~/nsc$ python3 network_slice_controller.py
{'source': 'HL5-1-2', 'destination': 'HL5-3-2', 'bandwidth': 150}
{'source': 'HL5-1-2', 'destination': 'HL4-1-2', 'bandwidth': 150}
{'source': 'HL5-3-2', 'destination': 'HL5-2-2', 'bandwidth': 100}
{'source': 'HL5-2-2', 'destination': 'HL4-2-2', 'bandwidth': 50}
{'source': 'HL5-3-2', 'destination': 'HL5-1-2', 'bandwidth': 50}
```

Planner's response:

```
1  [
2      {
3          "success": "True",
4          "description": "Traffic intent established successfully!",
5          "SR Policy TeraFlow": {
6              "0": [
7                  ["HL5-2-2", "eth-1/0/10.42"],
8                  ["HL4-2-2", "to_HL5-2-2"]
9              ],
10             "1": [
11                 ["HL4-2-2", "to_HL4-1-2"],
12                 ["HL4-1-2", "to_HL4-2-2"]
13             ]
14         },
15         "SR Policy (SR-LEA)": [24006, 24004]
16     }
17 ]
```

This requests needs
two slices

Slice Request

```

1 {
2   "slices": [
3     {
4       "slice_id": {
5         "context_id": {
6           "context_uuid": {"uuid": "admin"}
7         },
8         "slice_uuid": {
9           "uuid": "test-iron-16968341404863460"
10        }
11      },
12      "name": "test-iron-",
13      "slice_config": {
14        "config_rules": [
15          {
16            "action": 1,
17            "custom": {
18              "resource_key": "/settings",
19              "resource_value": {
20                "address_families": ["IPv4"],
21                "bgp_as": "",
22                "bgp_route_target": "",
23                "mtu": ""
24              }
25            }
26          }
27        ],
28        "action": 1,
29        "custom": {
30          "resource_key": "/device[HL5-2-2]/endpoint[eth-1/0/25]/settings",
31          "resource_value": {"router_id": "HL5-2-2"}
32        },
33        {
34          "action": 1,
35          "custom": {
36            "resource_key": "/device[HL5-1-2]/endpoint[eth-1/0/25]/settings",
37            "resource_value": {"router_id": "HL5-1-2"}
38          }
39        }
40      },
41    },
42  ],
43  "slice_constraints": [
44    {
45      "endpoint_location": {
46        "endpoint_id": {
47          "device_id": {
48            "device_uuid": {"uuid": "R1"}
49          },
50          "endpoint_uuid": {"uuid": "1/2"}
51        },
52        "location": {"region": ""}
53      },
54    }
55  ]
56 }
  
```

Identifiers

Endpoints

```

55 {
56   "endpoint_location": {
57     "endpoint_id": {
58       "device_id": {
59         "device_uuid": {"uuid": "R4"}
60       },
61       "endpoint_uuid": {"uuid": "1/3"}
62     },
63     "location": {"region": ""}
64   },
65   "sla_capacity": {
66     "capacity_gbps": 6000
67   },
68   "sla_availability": {
69     "availability": 0,
70     "num_disjoint_paths": 2,
71     "all_active": "True"
72   },
73   "sla_isolation": {"isolation_level": [0]}
74 }
75 ],
76 "slice_endpoint_ids": [
77 {
78 }
79 ]
80 ]
81 }
82 }
83 }
  
```

Slice requirements

```

84 {
85   "device_id": {
86     "device_uuid": {"uuid": "HL5-2-2"}
87   },
88   "endpoint_uuid": {"uuid": "eth-1/0/25"},
89   "topology_id": {
90     "context_id": {
91       "context_uuid": {"uuid": "admin"}
92     },
93     "topology_uuid": {"uuid": "admin"}
94   },
95   "device_id": {
96     "device_uuid": {"uuid": "HL5-1-2"}
97   },
98   "endpoint_uuid": {"uuid": "eth-1/0/25"},
99   "topology_id": {
100    "context_id": {
101      "context_uuid": {"uuid": "admin"}
102    },
103    "topology_uuid": {"uuid": "admin"}
104  },
105  "slice_status": {"slice_status": 1}
106 }
107 ],
108 }
109 ]
110 }
111 }
  
```

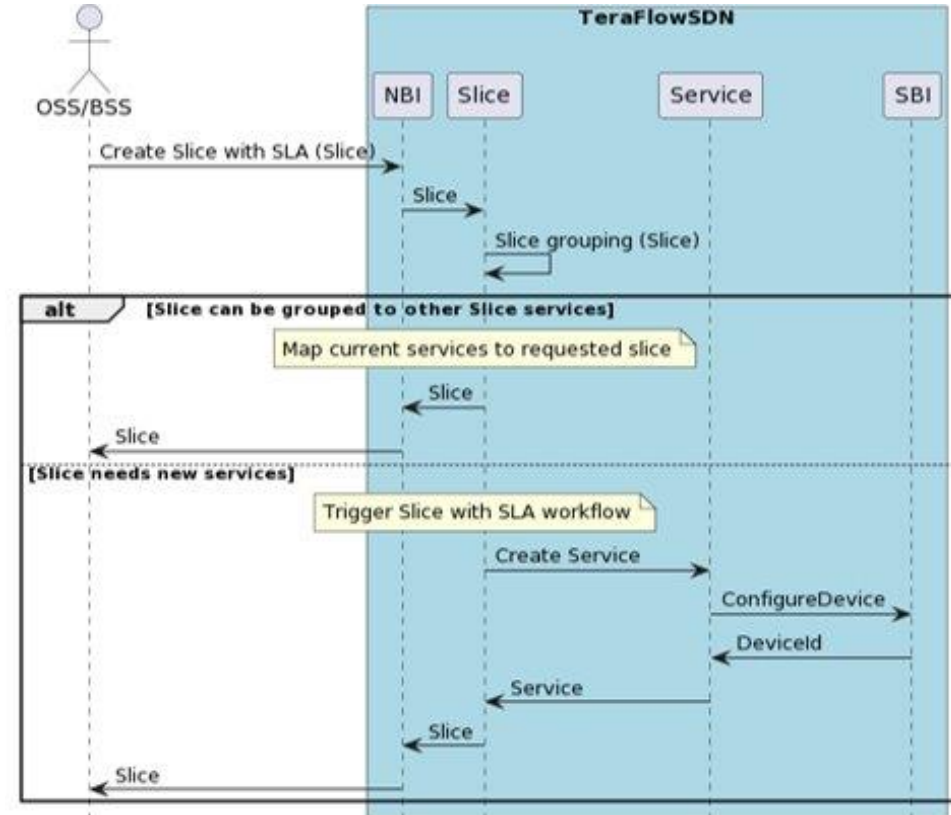
Slice management in TFS

Slice

0 slices found in context *admin*

UUID	Name	End points	Status
820a4b45-d8f9-48cf-b781-957854d5ab91	820a4b45-d8f9-48cf-b781-957854d5ab91	<ul style="list-style-type: none"> eth-1/0/22 / Device: R149 eth-1/0/21 / Device: R199 	ACTIVE

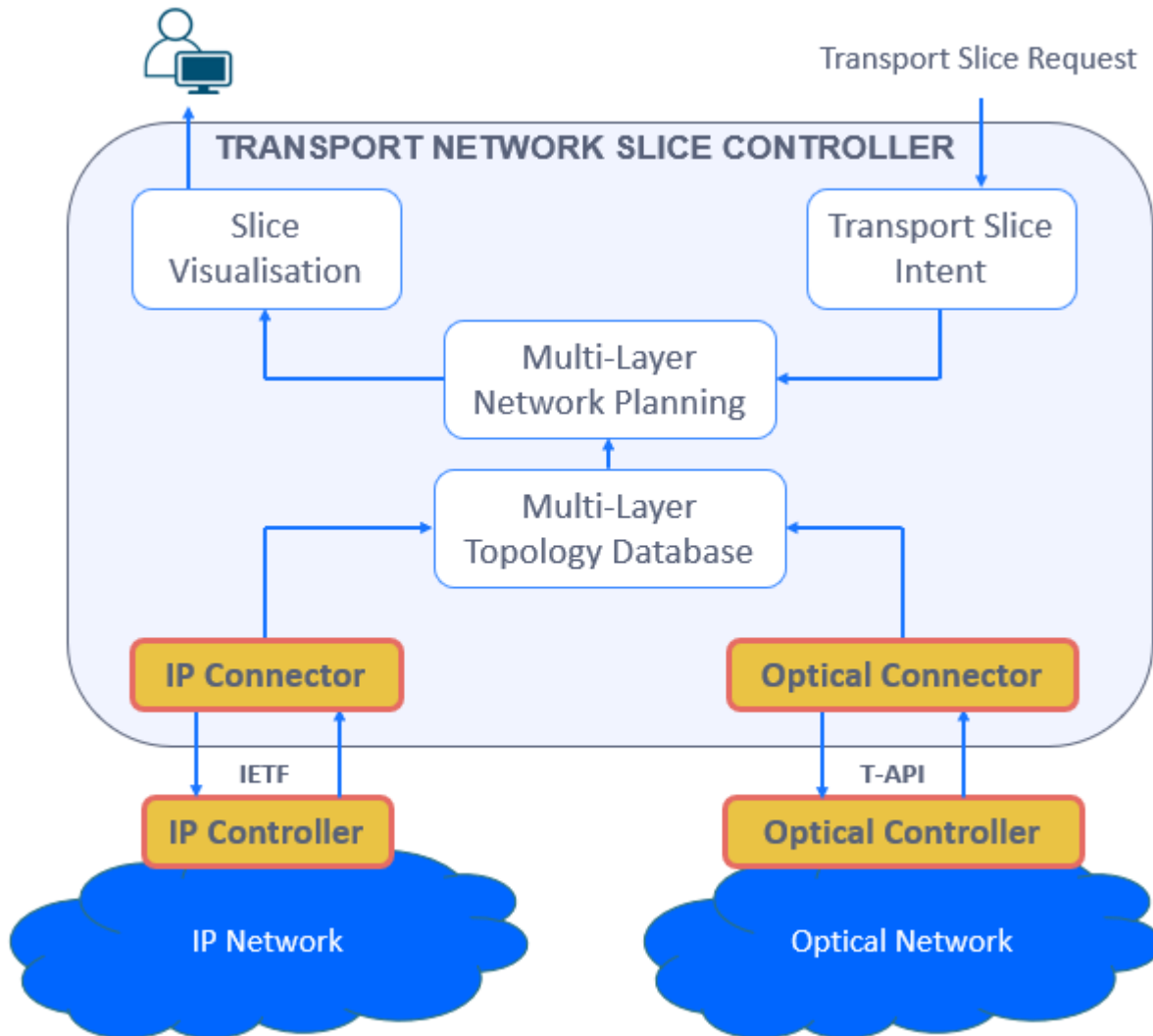
Constraints:		
Kind	Key/Type	Value
Endpoint Location	R149 / eth-1/0/22	Region: 1
Endpoint Location	R199 / eth-1/0/21	Region: 2
Configurations:		
Key	Value	
/settings	<ul style="list-style-type: none"> address_families: [IPv4] bgp_as: 65200 bgp_routes_target: 65000:333 mtu: 1512 	
/device(R149)/endpoint(eth-1/0/22)/settings	<ul style="list-style-type: none"> circuit_id: 100 remote_router: S.S.S.1 router_id: S.S.S.5 sub_interface_index: 0 vlan_id: 111 	
/device(R199)/endpoint(eth-1/0/21)/settings	<ul style="list-style-type: none"> circuit_id: 100 remote_router: S.S.S.5 router_id: S.S.S.6 sub_interface_index: 0 vlan_id: 111 	



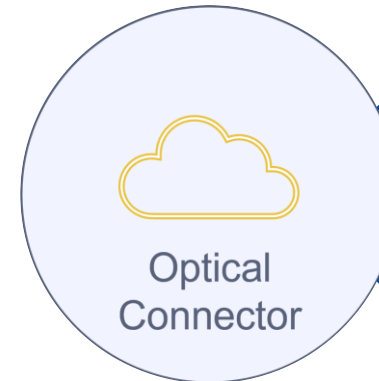
Functional Architecture

Slice Visualization

Opening up Disaggregated IP & Optical Networks



Retrieves IP Network Topology according to IETF Standard



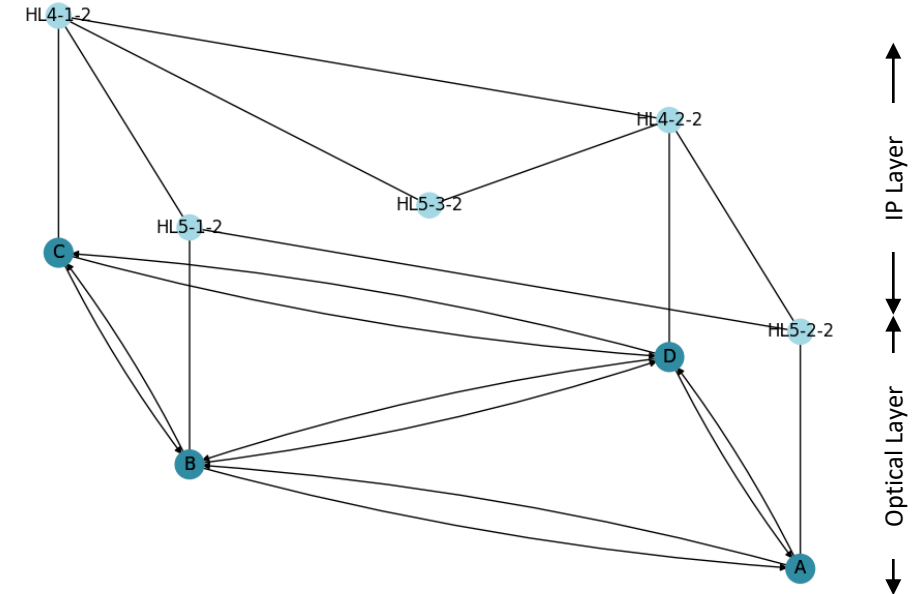
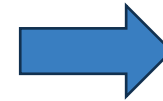
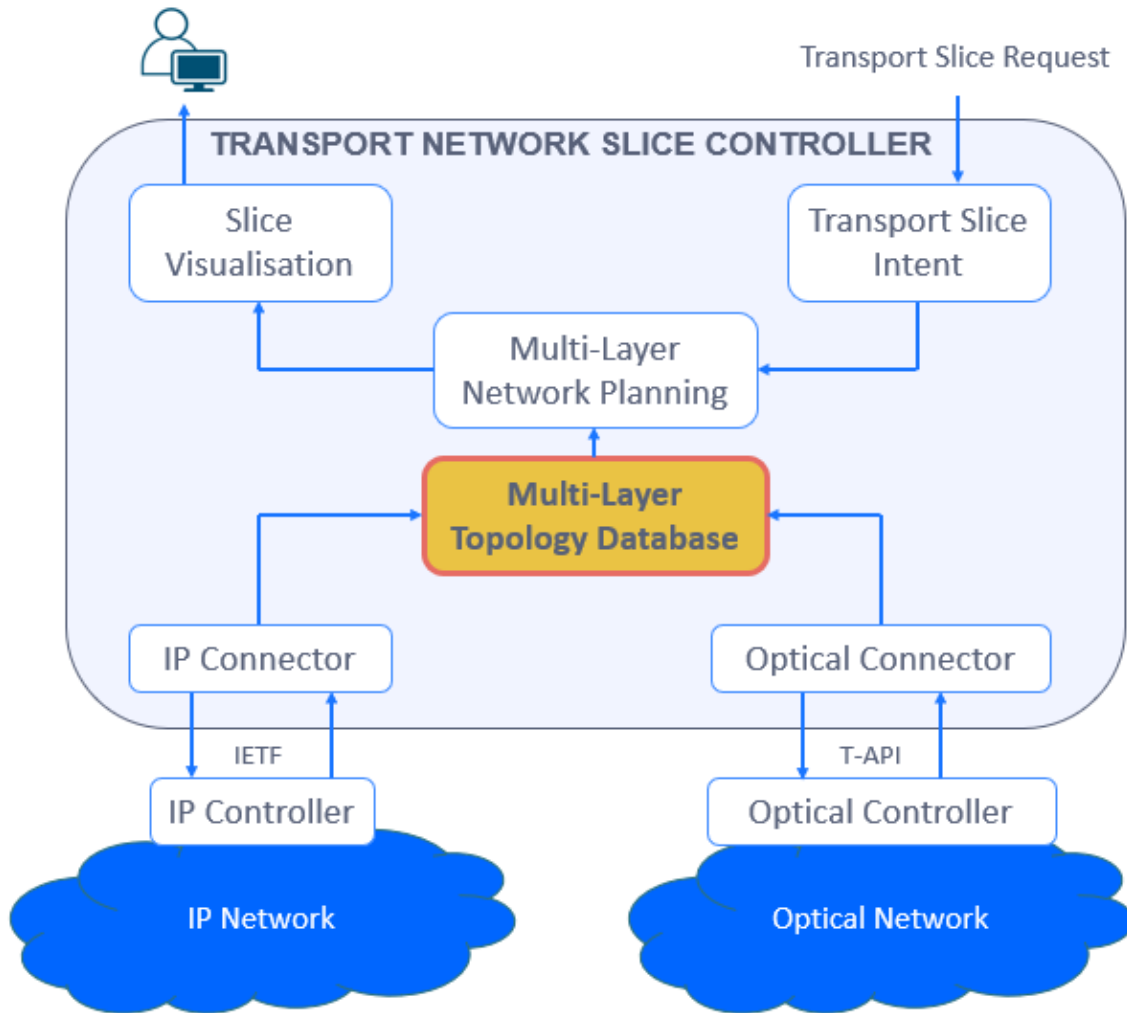
Retrieves Optical Network Topology according to T-API Standard

In Progress: **Real-Time** Extraction of Topological Data

Now: Static topology info stored under ./json

Slice Visualization

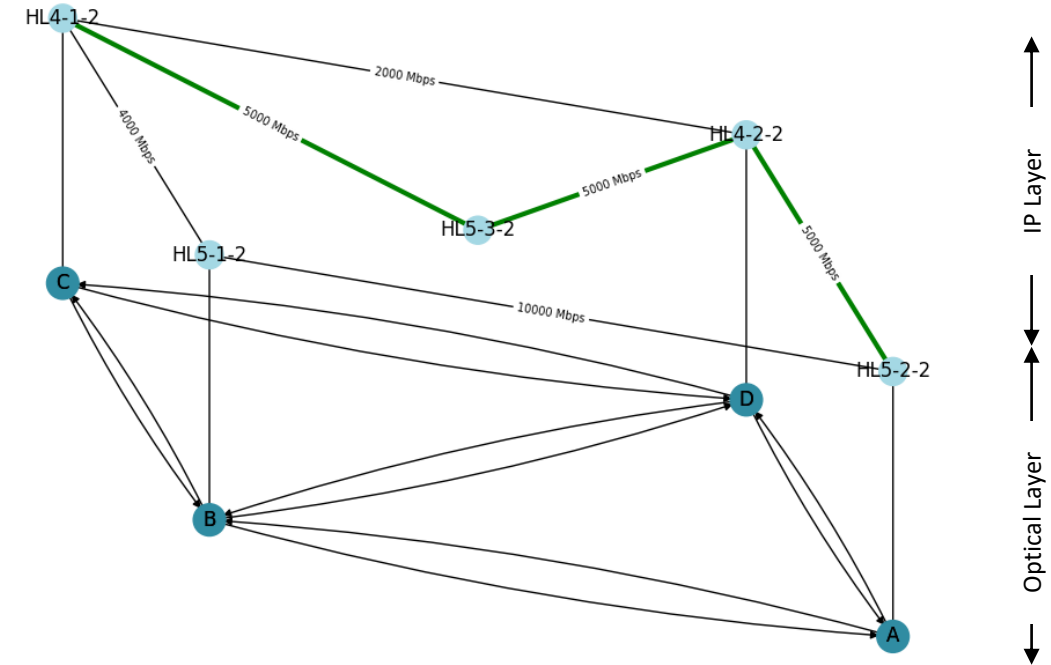
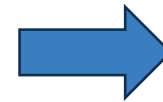
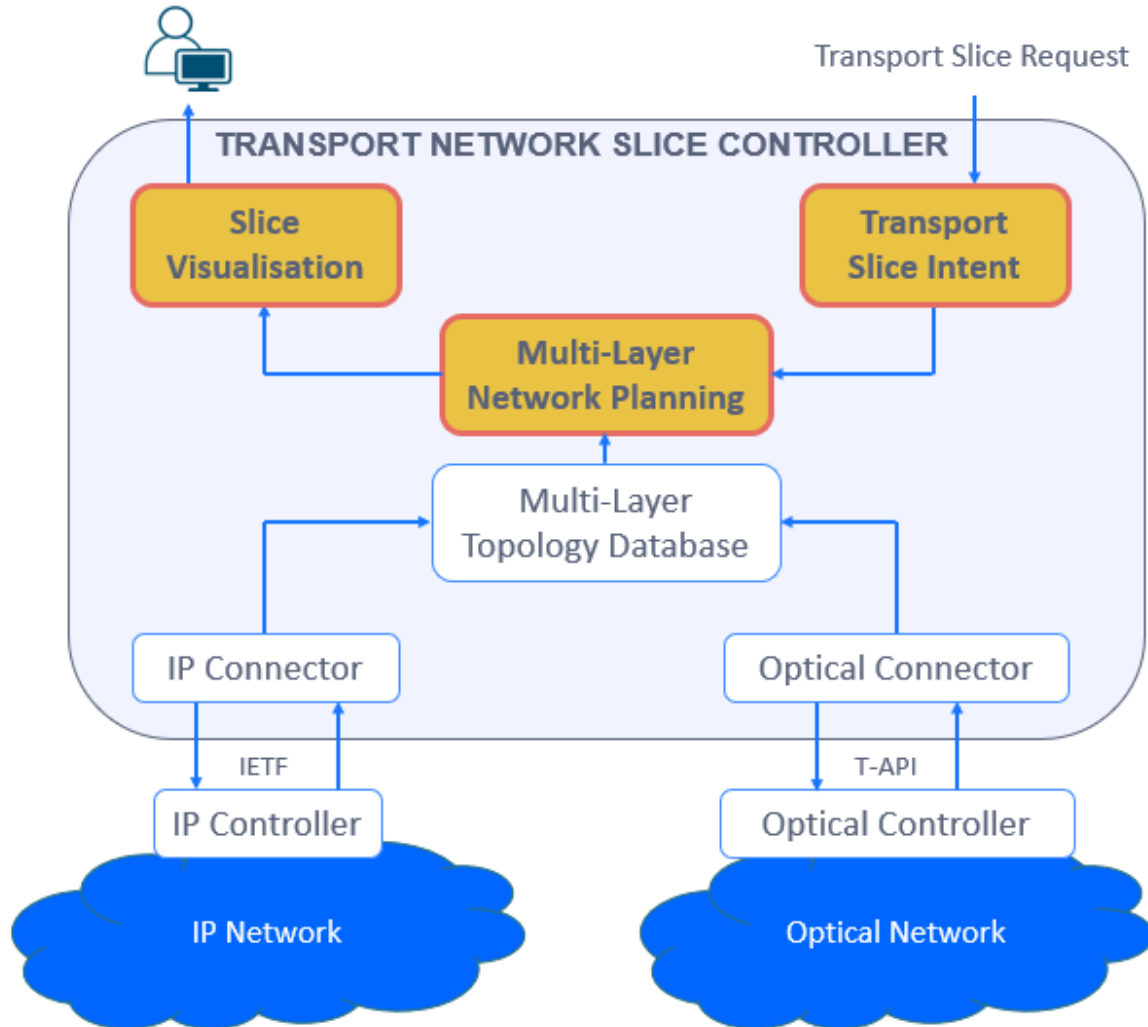
Generating and Visualizing / Optical-to-IP Mapping



- Example presents a simplified and hypothetical optical-to-IP mapping
- Specific **mapping** is based on real **network designs** and **objectives**
 - Maximizing resource utilization
 - Minimizing latency
 - Ensuring resilience

Slice Visualization

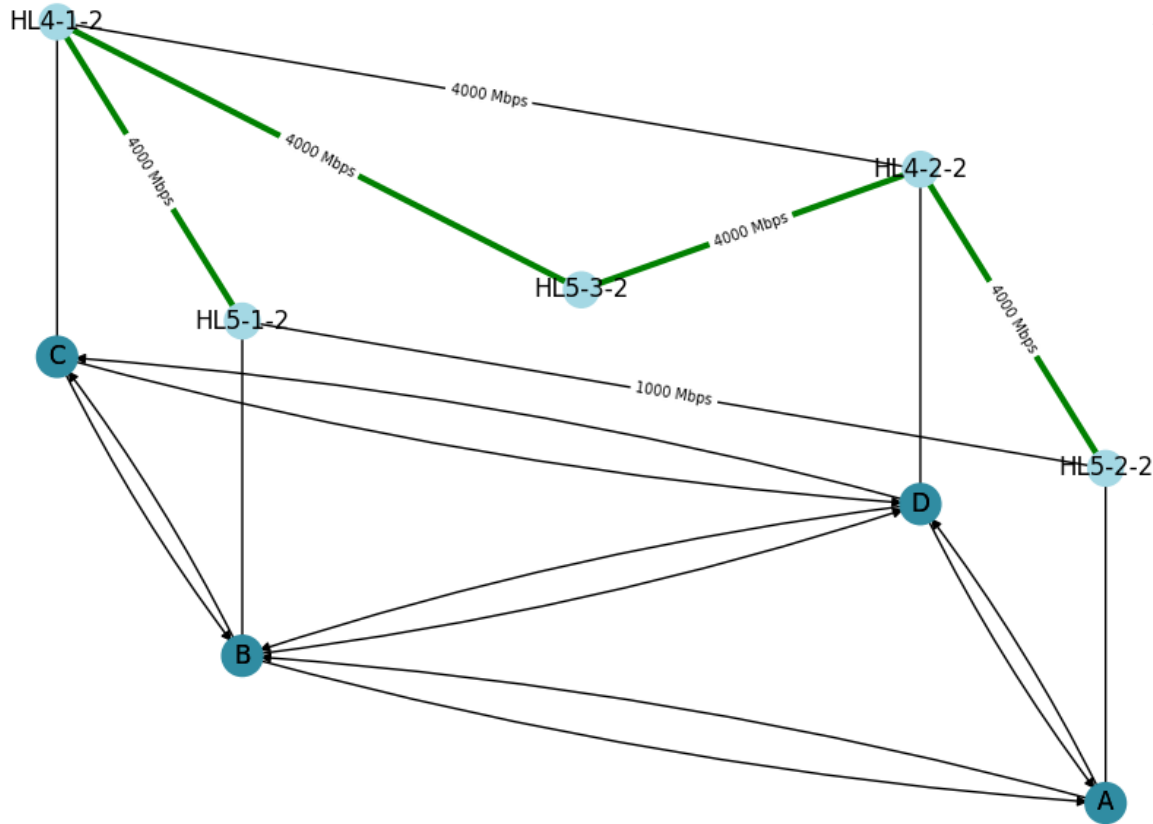
Allocating and Visualizing / Transport Slice Requests



- Transport slice request: *src, dst, BW*
- Compute **optimal path**
- Define segment routing policy
- **Resource check:** If insufficient IP resources
→ Additional connectivity request in optical layer using T-API required (**in progress**)

Path Planning - Example

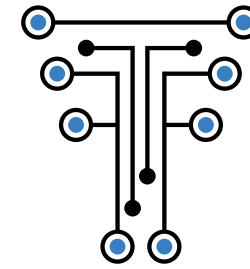
Best available path 6000 Mbps



Segment routing policies provided to TeraFlow.
Note: now printed to terminal i.o. saved, can be modified in topologyviewer.py (lines 316-317).

Theoretical segment routing policies.
Note: can probably be ignored

```
{
  "success": true,
  "description": "Traffic intent established successfully!",
  "path": [
    "HL5-1-2",
    "HL4-1-2",
    "HL5-3-2",
    "HL4-2-2",
    "HL5-2-2"
  ],
},
"SR Policy TeraFlow": {
  "0": [
    [
      "HL5-1-2",
      "eth-1/0/10.41"
    ],
    [
      "HL4-1-2",
      "to_HL5-1-2"
    ]
  ],
  "1": [
    [
      "HL4-1-2",
      "to_HL5-3-2"
    ],
    [
      "HL5-3-2",
      "GigabitEthernet0/2/5"
    ]
  ],
  "2": [
    [
      "HL5-3-2",
      "GigabitEthernet0/2/7"
    ],
    [
      "HL4-2-2",
      "to_HL5-3-2"
    ]
  ],
  "3": [
    [
      "HL4-2-2",
      "to_HL5-2-2"
    ],
    [
      "HL5-2-2",
      "eth-1/0/10.42"
    ]
  ]
},
"SR Policy (SR-LEA)": [
  16003,
  16001
]
}
```



TeraFlow
SDN
by ETSI

Thank You!

This work has been partially funded by the 2020 Horizon Europe project Int5Gent (grant agreement No. 957403), ALLEGRO (grant agreement No. 101092766) and SEASON (grant agreement No. 101096120)