

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: September 7, 2022

X. Liu
IBM Corporation
J. Tantsura
Microsoft
I. Bryskin
Individual
L. Contreras
Telefonica
Q. Wu
Huawei
S. Belotti
Nokia
R. Rokui
Ciena
March 6, 2022

IETF Network Slice YANG Data Model
draft-liu-teas-transport-network-slice-yang-05

Abstract

This document describes a YANG data model for managing and controlling IETF network slices, defined in [\[I-D.ietf-teas-ietf-network-slices\]](#).

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on September 7, 2022.

Copyright Notice

Copyright (c) 2022 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](https://trustee.ietf.org/license-info) and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Introduction	3
1.1.	Terminology	3
1.2.	Tree Diagrams	4
2.	Modeling Considerations	4
2.1.	Relationships to Related Topology Models	4
2.2.	Network Slice with TE	5
2.3.	ACTN for Network Slicing	6
3.	Model Applicability	6
3.1.	Network Slicing by Virtualization	7
3.2.	Network Slicing by TE Overlay	9
4.	Model Communication Types	11
4.1.	P2P	11
4.2.	P2MP	12
4.3.	MP2MP	13
4.4.	A2A	15
5.	Model Tree Structure	16
5.1.	Module ietf-network-slice	16
5.2.	Module ietf-network-slice-connectivity	17
6.	YANG Modules	17
6.1.	Module ietf-network-slice	17
6.2.	Module ietf-network-slice-connectivity	23
7.	IANA Considerations	26
8.	Security Considerations	27
9.	Acknowledgements	28
10.	References	29
10.1.	Normative References	29
10.2.	Informative References	30
Appendix A.	Data Tree for the Example in Section 3.1.	32
A.1.	Native Topology	32
A.2.	Network Slice Blue	36
	Authors' Addresses	42

1. Introduction

This document defines a YANG [[RFC7950](#)] data model for representing, managing, and controlling IETF network slices, defined in [[I-D.ietf-teas-ietf-network-slices](#)]

The defined data model is an interface between customers and providers for configurations and state retrievals, so as to support network slicing as a service. Through this model, a customer can learn the slicing capabilities and the available resources of the provider. A customer can request or negotiate with a network slicing provider to create an instance. The customer can incrementally update its requirements on individual topology elements in the slice instance, and retrieve the operational states of these elements. With the help of other mechanisms and data models defined in IETF, the telemetry information can be published to the customer.

As described in Section 3 of [[I-D.contreras-teas-slice-controller-models](#)], the data model defined in this document complements the data model defined in [[I-D.ietf-teas-ietf-network-slice-nbi-yang](#)]. In addition to the provider's view, the data model defined in this document models the Type 2 service defined in [[RFC8453](#)].

The YANG data model in this document conforms to the Network Management Datastore Architecture (NMDA) [[RFC8342](#)].

1.1. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [BCP 14](#), [[RFC2119](#)] [[RFC8174](#)] when, and only when, they appear in all capitals, as shown here.

The following terms are defined in [[RFC7950](#)] and are not redefined

here:

- o augment
- o data model
- o data node

[1.2.](#) Tree Diagrams

Tree diagrams used in this document follow the notation defined in [\[RFC8340\]](#).

[2.](#) Modeling Considerations

An IETF network slice is modeled as network topology defined in [\[RFC8345\]](#), with augmentations. A new network type "network-slice" is defined in this document. When a network topology data instance contains the network-slice network type, it represents an instance of an IETF network slice.

[2.1.](#) Relationships to Related Topology Models

There are several related YANG data models that have been defined in IETF. Some of these are:

Network Topology Model:

Defined in [\[RFC8345\]](#).

OTN Topology Model:

Defined in [\[I-D.ietf-ccamp-otn-topo-yang\]](#).

L2 Topology Model:

Defined in [\[I-D.ietf-i2rs-yang-l2-network-topology\]](#).

L3 Topology Model:

Defined in [\[RFC8346\]](#).

TE Topology Model:
Defined in [[RFC8795](#)].

Figure 1 shows the relationships among these models. The box of dotted lines denotes the model defined in this document.

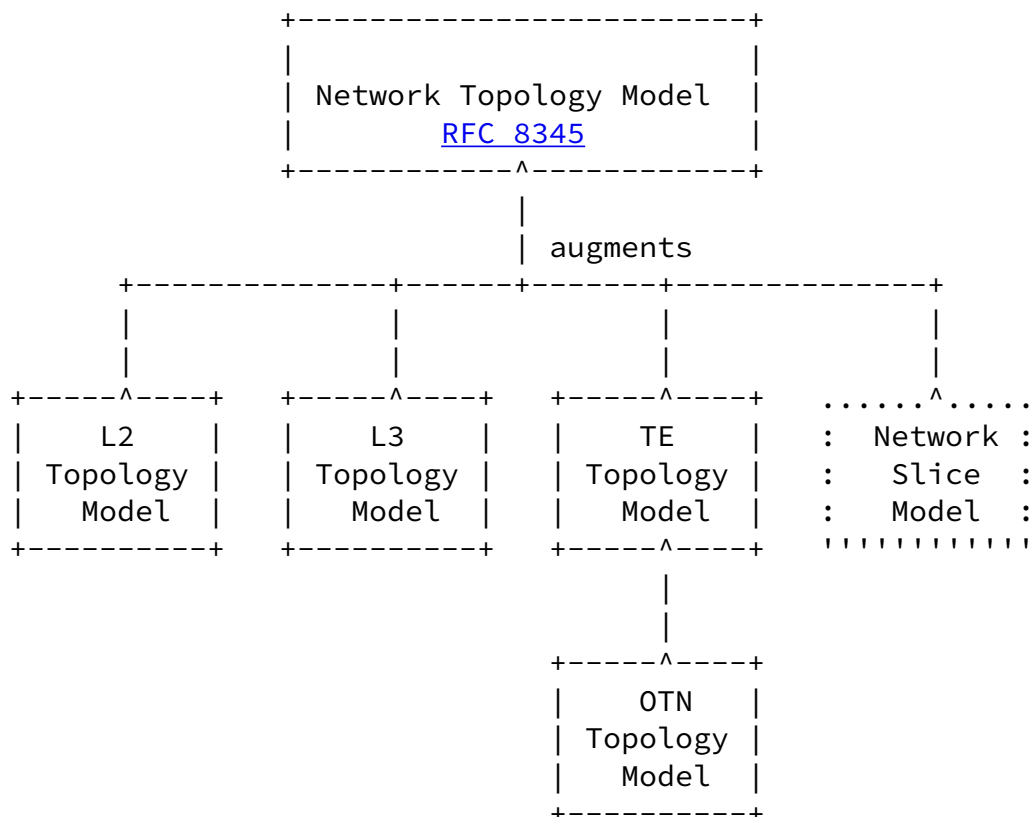
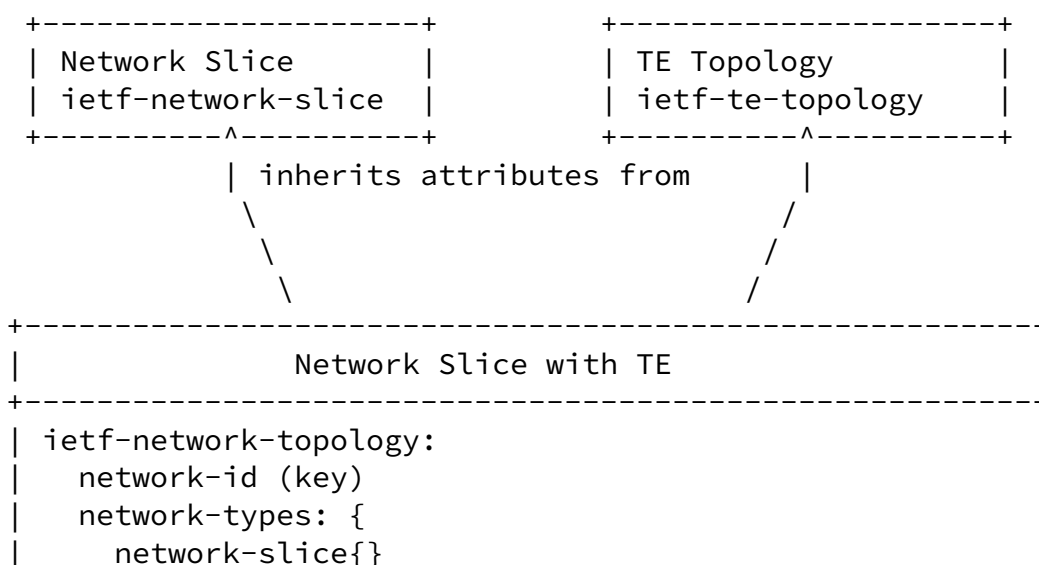


Figure 1: Model Relationships

2.2. Network Slice with TE

In many situations, an IETF network slice needs to have TE (Traffic Engineering) capabilities to achieve certain network characteristics. The TE Topology Model defined in [RFC8795] can be used to make an IETF network slice TE capable. To achieve this, an IETF network slice instance will be configured to have both "network-slice" and "te-topology" network types, taking advantage of the multiple inheritance capability featured by the network topology model [RFC8345]. The following diagram shows their relations.



te-topology{}	
}	
<other network topology attributes>	
ietf-network-slice:	ietf-te-topology:
<network slice attributes>	<TE attributes>

Figure 2: Network Slice with TE

This method can be applied to other types of network topology models too. For example, when a network topology instance is configured to have the types of "network-slice" defined in this document, "te-topology" defined in [RFC8795], and "l3-unicast-topology" defined in [RFC8346], this network topology instance becomes an IETF network slice instance that can perform layer 3 traffic engineering.

2.3. ACTN for Network Slicing

Since ACTN topology data models are based on the network topology model defined in [RFC8345], the augmentations defined in this document are effective augmentations to the ACTN topology data models, resulting in making the ACTN framework [RFC8453] and data models [I-D.ietf-teas-actn-yang] capable of slicing networks with the required network characteristics.

3. Model Applicability

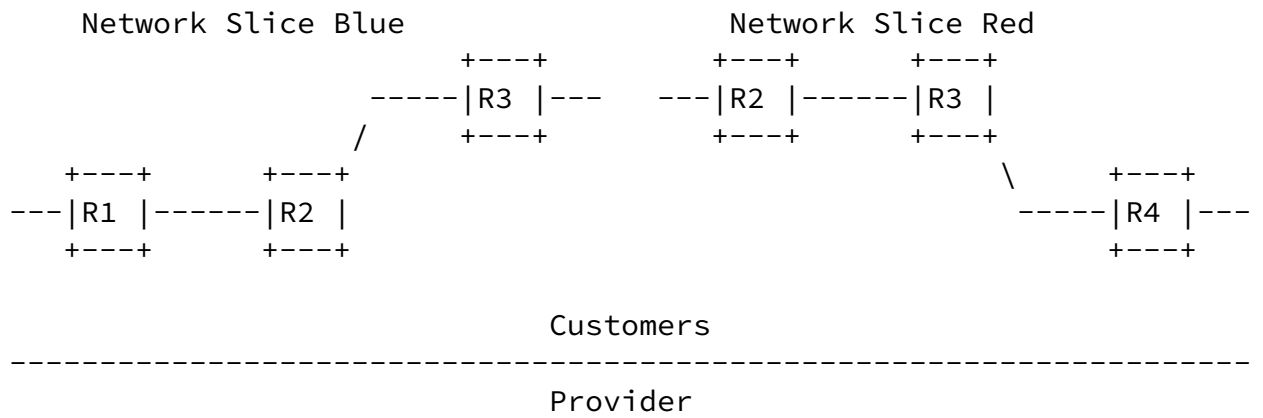
There are many technologies to achieve network slicing. The data model defined in this document can be applied to a wide ranges of cases. This section describes how this data model is applied to a few cases.

3.1. Network Slicing by Virtualization

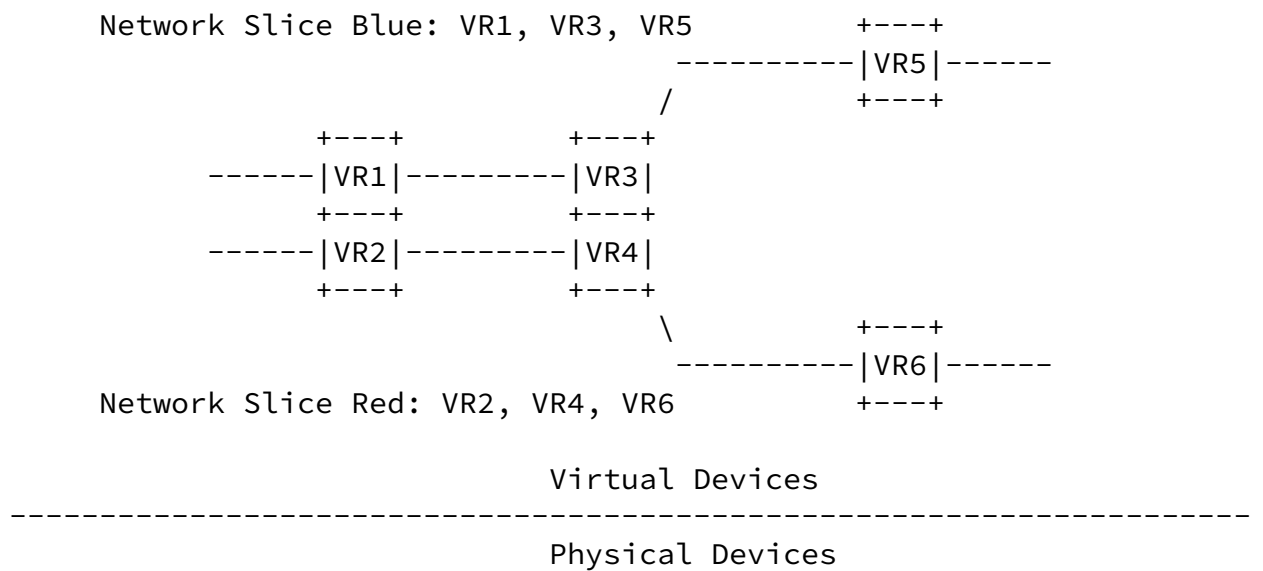
In the case shown in Figure 3, node virtualization is used to separate and allocate resources in physical devices. Two virtual routers VR1 and VR2 are created over physical router R1. Each of the virtual routers takes a portion of the resources such as ports and memory in the physical router. Depending on the requirements and the implementations, they may share certain resources such as processors,

ASICs, and switch fabric.

As an example, [Appendix A](#). shows the JSON encoded data instances of the native topology and the customized topology for Network Slice Blue.



Customized Topology
Provider Network with Virtual Devices



Native Topology
Provider Network with Physical Devices

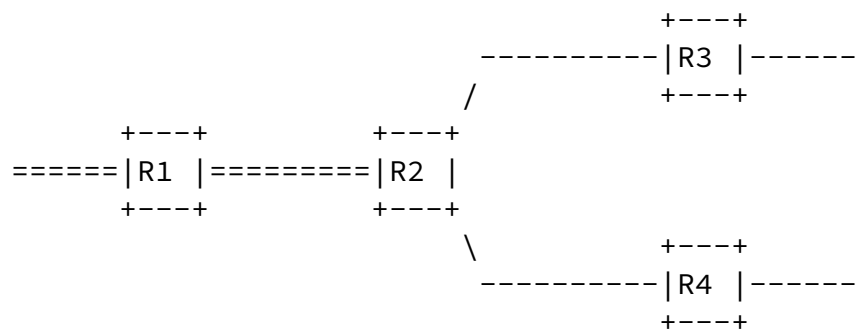


Figure 3: Network Slicing by Virtualization

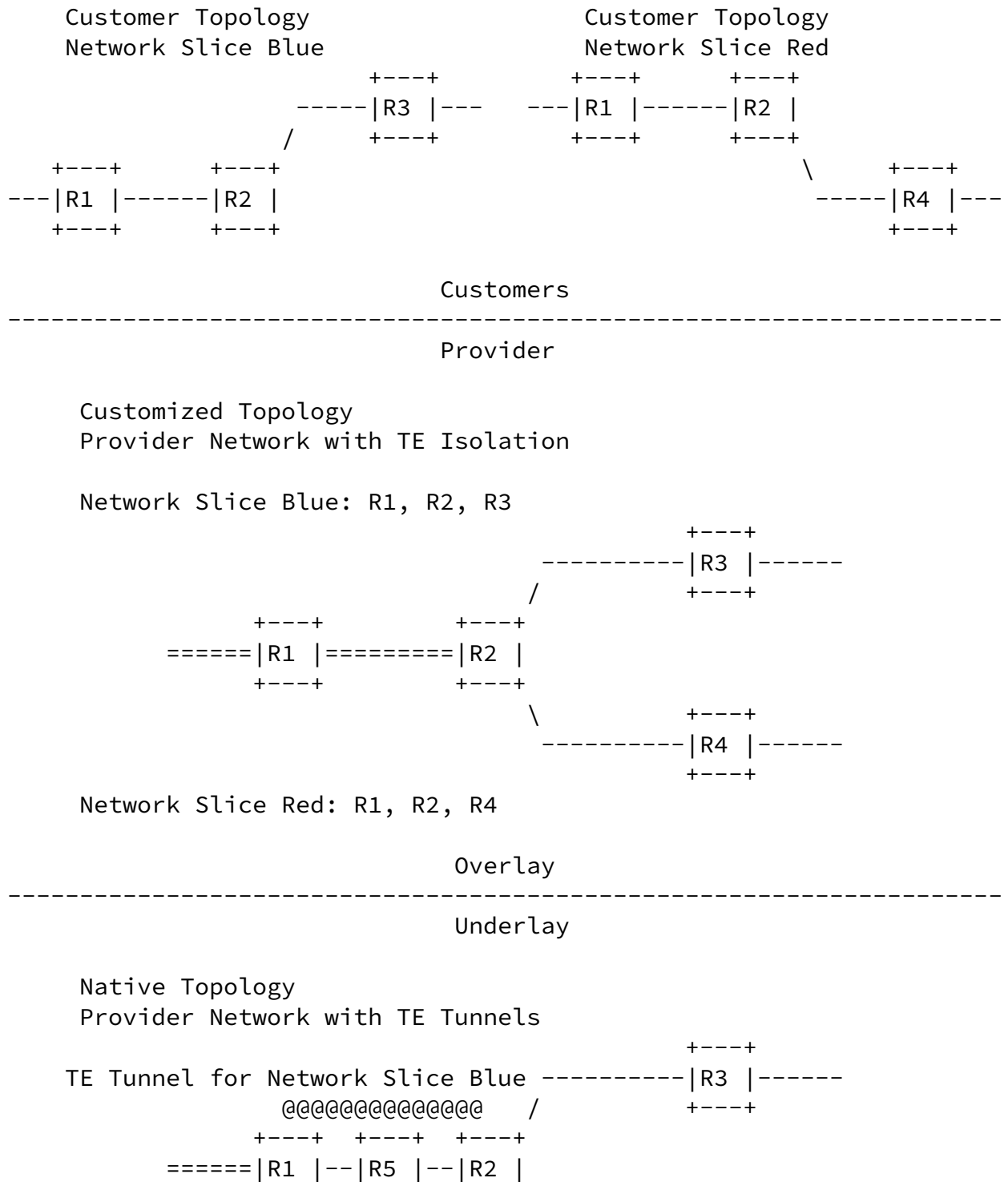
[3.2.](#) Network Slicing by TE Overlay

Figure 4 shows a case where TE (Traffic Engineering) overlay is applied to achieve logically separated customer IETF network slices. In the underlay TE capable network, TE tunnels are established to support the TE links in the overlay network. These links and tunnels maintain the characteristics required by the customers. The provider selects the proper logical nodes and links in the overlay network, assigns them to specific IETF network slices, and uses the data model defined in this document to send the results to the customers.

Internet-Draft

IETF Network Slice YANG Model

March 2022



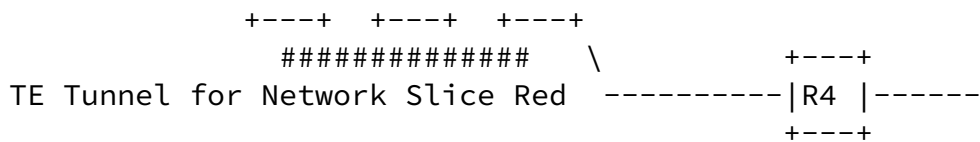


Figure 4: Network Slicing by TE Overlay

4. Model Communication Types

Section 3.2 of [[I-D.ietf-teas-ietf-network-slices](#)] describes various communication types that an IETF network slice may serve, including P2P, P2MP, MP2P, MP2MP, and A2A. The data models specified in [[RFC8345](#)] and [[RFC8795](#)] support only P2P and A2A. In this document, the YANG module `ietf-network-slice-connectivity` is defined to extend the capabilities to cover P2MP, MP2P, and MP2MP.

The YANG module `ietf-network-slice-connectivity` is defined in [Section 6.2](#) of this document, with its structure shown in [Section 5.2](#) of this document. This YANG module introduces two modeling constructs in each connectivity construct (that is called connectivity matrix entries in [[RFC8795](#)]):

Replication Group:

A replication group contains a list of connectivity constructs (that are called connectivity matrix entries in [RFC 8795](#)). When traffic is sent to one entry in this replication group, the traffic is replicated to all other entries in the same replication group.

Receiver Constraint Group:

A receiver constraint group contains a list of connectivity constructs (that are called connectivity matrix entries in [RFC 8795](#)). When traffic is sent to one or more entries in this receiver constraint group, the constraints specified in this receiver constraint group are applied to the receiver-side termination points referenced by all entries in this receiver constraint group.

The following sections describe some data examples:

NSE3 <-> NSC7 : Bidirectional P2P connectivity
NSE4 -> NSE8 : Unidirectional P2P connectivity

```
{
  "connectivity-matrices": {
    "connectivity-matrix": [
      {
        "id": 1,
        "from": {
          "tp-ref": "NSE3"
        },
        "to": {
          "tp-ref": "NSE7"
        }
      },
      {
        "id": 2,
        "from": {
          "tp-ref": "NSE7"
        },
        "to": {
          "tp-ref": "NSE3"
        }
      },
      {
        "id": 3,
```

```

    "from": {
      "tp-ref": "NSE4"
    },
    "to": {
      "tp-ref": "NSE8"
    }
  ]
}

```

[4.2.](#) P2MP

```

    NSE5 -> {NSC9, NSE10}
  {
    "connectivity-matrices": {
      "connectivity-matrix": [
        {
          "id": 1,
          "from": {
            "tp-ref": "NSE5"
          },
          "to": {
            "tp-ref": "NSE9"
          }
        },
        {
          "id": 2,
          "from": {
            "tp-ref": "NSE5"
          },
          "to": {

```

```

        "tp-ref": "NSE10"
    },
    ],
    "replication-group": [
        {
            "id": 1,
            "entry": [1, 2]
        }
    ]
}

```

4.3. MP2MP

{NSE14, NSE15} -> {NSE16, NSE17}

```

{
  "connectivity-matrices": {
    "connectivity-matrix": [
      {
        "id": 1,
        "from": {
          "tp-ref": "NSE14"
        },
        "to": {
          "tp-ref": "NSE16"
        }
      },
      {
        "id": 2,
        "from": {

```

```

        "tp-ref": "NSE14"
    },
    {
        "to": {
            "tp-ref": "NSE17"
        }
    }
  ],
  "connectivity-matrix": [
    {
      "id": 3,
      "from": {
        "tp-ref": "NSE15"
      },

```

```

        "to": {
          "tp-ref": "NSE16"
        }
      ],
      "connectivity-matrix": [
        {
          "id": 4,
          "from": {
            "tp-ref": "NSE15"
          },
          "to": {
            "tp-ref": "NSE17"
          }
        }
      ],
      "replication-group": [
        {
          "id": 1,
          "entry": [1, 2]
        }
      ],
      "replication-group": [
        {
          "id": 2,
          "entry": [3, 4]
        }
      ],
      "receiver-constraint-group": [
        {
          "id": 1,
          "entry": [1, 3]
        }
      ],
      "receiver-constraint-group": [
        {
          "id": 2,
          "entry": [2, 4]
        }
      ]
    }
  }
}

```

[4.4.](#) A2A

{NSE1, NSE2, NSE6} -> {NSE1, NSE2, NSE6}


```

{
  "connectivity-matrices": {
    "connectivity-matrix": [
      {
        "id": 1,
        "from": {
          "tp-ref": "NSE1"
        },
        "to": {
          "tp-ref": "NSE2"
        }
      },
      {
        "id": 2,
        "from": {
          "tp-ref": "NSE1"
        },
        "to": {
          "tp-ref": "NSE6"
        }
      },
      {
        "id": 3,
        "from": {
          "tp-ref": "NSE2"
        },
        "to": {
          "tp-ref": "NSE1"
        }
      },
      {
        "id": 4,
        "from": {
          "tp-ref": "NSE2"
        },
        "to": {
          "tp-ref": "NSE6"
        }
      },
      {
        "id": 5,
        "from": {
          "tp-ref": "NSE6"
        },

```

```
        "to": {
          "tp-ref": "NSE1"
        }
      ],
      "connectivity-matrix": [
        {
          "id": 6,
          "from": {
            "tp-ref": "NSE6"
          },
          "to": {
            "tp-ref": "NSE2"
          }
        }
      ]
    }
  }
}
```

[5. Model Tree Structure](#)

[5.1. Module ietf-network-slice](#)

TODO - Complete IETF network slice attributes that are technology-agnostic and common to all use cases.

```
module: ietf-network-slice
  augment /nw:networks/nw:network/nw:network-types:
    +--rw network-slice!
  augment /nw:networks/nw:network:
    +--rw network-slice
      +--rw optimization-criterion?   identityref
      +--rw delay-tolerance?          boolean
      +--rw periodicity*              uint64
      +--rw isolation-level?          identityref
  augment /nw:networks/nw:network/nw:node:
    +--rw network-slice
      +--rw isolation-level?          identityref
      +--rw compute-node-id?         string
      +--rw storage-id?              string
  augment /nw:networks/nw:network/nt:link:
    +--rw network-slice
      +--rw delay-tolerance?          boolean
      +--rw periodicity*              uint64
      +--rw isolation-level?          identityref
```

[5.2.](#) Module ietf-network-slice-connectivity

```
module: ietf-network-slice-connectivity
  augment /nw:networks/nw:network/nw:node/tet:te
    /tet:te-node-attributes/tet:connectivity-matrices
    /tet:connectivity-matrix:
  +--rw replication-group* [id]
  |   +--rw id          uint32
  |   +--rw entry*      -> ../../tet:id
  +--rw receiver-constraint-group* [id]
  |   +--rw id          uint32
  |   +--rw entry*      -> ../../tet:id
  |   +--rw te-bandwidth
  |       +--rw (technology)?
  |           +--:(generic)
  |               +--rw generic?   te-bandwidth
  augment /nw:networks/nw:network/nw:node/tet:te
    /tet:information-source-entry/tet:connectivity-matrices
    /tet:connectivity-matrix:
  +--ro replication-group* [id]
  |   +--ro id          uint32
  |   +--ro entry*      -> ../../tet:id
  +--ro receiver-constraint-group* [id]
  |   +--ro id          uint32
  |   +--ro entry*      -> ../../tet:id
  |   +--ro te-bandwidth
  |       +--ro (technology)?
  |           +--:(generic)
  |               +--ro generic?   te-bandwidth
```

[6.](#) YANG Modules

[6.1.](#) Module ietf-network-slice

This module references [\[RFC8345\]](#), [\[RFC8776\]](#), and [\[GSMA-NS-Template\]](#)

<CODE BEGINS> file "ietf-network-slice@2020-11-01.yang"

```
module ietf-network-slice {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-network-slice";
  prefix "ns";

  import ietf-network {
    prefix "nw";
    reference "RFC 8345: A YANG Data Model for Network Topologies";
  }
```

Liu, et al.

Expires September 7, 2022

[Page 17]

Internet-Draft

IETF Network Slice YANG Model

March 2022

```
}
import ietf-network-topology {
  prefix "nt";
  reference "RFC 8345: A YANG Data Model for Network Topologies";
}
import ietf-te-types {
  prefix "te-types";
  reference
    "RFC 8776: Traffic Engineering Common YANG Types";
}
```

```
organization
  "IETF Traffic Engineering Architecture and Signaling (TEAS)
  Working Group";
```

```
contact
```

```
"WG Web:  <http://tools.ietf.org/wg/teas/>
WG List:  <mailto:teas@ietf.org>
```

```
Editor:   Xufeng Liu
          <mailto:xufeng.liu.ietf@gmail.com>
```

```
Editor:   Jeff Tantsura
          <mailto:jefftant.ietf@gmail.com>
```

```
Editor:   Igor Bryskin
          <mailto:i\_bryskin@yahoo.com>
```

```
Editor:   Luis Miguel Contreras Murillo
          <mailto:luismiguel.contrerasmurillo@telefonica.com>
```

```
Editor:   Qin Wu
          <mailto:bill.wu@huawei.com>
```

Editor: Sergio Belotti
<mailto:sergio.belotti@nokia.com>

Editor: Reza Rokui
<mailto:reza.rokui@nokia.com>

";

description

"YANG data model for representing and managing network
slices.

Copyright (c) 2020 IETF Trust and the persons identified as
authors of the code. All rights reserved.

Liu, et al.

Expires September 7, 2022

[Page 18]

Internet-Draft

IETF Network Slice YANG Model

March 2022

Redistribution and use in source and binary forms, with or
without modification, is permitted pursuant to, and subject to
the license terms contained in, the Simplified BSD License set
forth in [Section 4.c](#) of the IETF Trust's Legal Provisions
Relating to IETF Documents
(<http://trustee.ietf.org/license-info>).

This version of this YANG module is part of RFC XXXX; see the
RFC itself for full legal notices.";

```
revision 2020-11-01 {  
  description "Initial revision";  
  reference  
    "RFC XXXX: YANG Data Model for Network Slices";  
}  
  
/*  
 * Identities  
 */  
identity isolation-level {  
  description  
    "Base identity for the isolation-level.";  
  reference  
    "GSMA-NS-Template: Generic Network Slice Template,  
    Version 3.0.";  
}
```

```

identity no-isolation {
    base isolation-level;
    description
        "Network slices are not separated.";
}
identity physical-isolation {
    base isolation-level;
    description
        "Network slices are physically separated (e.g. different rack,
        different hardware, different location, etc.).";
}
identity logical-isolation {
    base isolation-level;
    description
        "Network slices are logically separated.";
}
identity process-isolation {
    base physical-isolation;
    description
        "Process and threads isolation.";
}
identity physical-memory-isolation {

```

```

    base physical-isolation;
    description
        "Process and threads isolation.";
}
identity physical-network-isolation {
    base physical-isolation;
    description
        "Process and threads isolation.";
}
identity virtual-resource-isolation {
    base logical-isolation;
    description
        "A network slice has access to specific range of resources
        that do not overlap with other network slices
        (e.g. VM isolation).";
}
identity network-functions-isolation {
    base logical-isolation;
    description

```

```

        "NF (Network Function) is dedicated to the network slice, but
        virtual resources are shared.";
    }
    identity service-isolation {
        base logical-isolation;
        description
            "NSC data are isolated from other NSCs, but virtual
            resources and NFs are shared.";
    }

    /*
    * Groupings
    */
    grouping network-slice-topology-attributes {
        description "Network Slice topology scope attributes.";
        container network-slice {
            description
                "Containing Network Slice attributes.";
            leaf optimization-criterion {
                type identityref {
                    base te-types:objective-function-type;
                }
                description
                    "Optimization criterion applied to this topology.";
            }
            leaf delay-tolerance {
                type boolean;
                description
                    "'true' if is not too critical how long it takes to deliver

```

```

        the amount of data.";
        reference
            "GSMA-NS-Template: Generic Network Slice Template,
            Version 3.0.";
    }
    leaf-list periodicity {
        type uint64;
        units seconds;
        description
            "A list of periodicities supported by the network slice.";
        reference
            "GSMA-NS-Template: Generic Network Slice Template,

```

```

        Version 3.0.";
    }
    leaf isolation-level {
        type identityref {
            base isolation-level;
        }
        description
            "A network slice instance may be fully or partly, logically
            and/or physically, isolated from another network slice
            instance. This attribute describes different types of
            isolation:";
    }
} // network-slice
} // network-slice-topology-attributes

grouping network-slice-node-attributes {
    description "Network Slice node scope attributes.";
    container network-slice {
        description
            "Containing Network Slice attributes.";
        leaf isolation-level {
            type identityref {
                base isolation-level;
            }
            description
                "A network slice instance may be fully or partly, logically
                and/or physically, isolated from another network slice
                instance. This attribute describes different types of
                isolation:";
        }
        leaf compute-node-id {
            type string;
            description
                "Reference to a compute node instance specified in
                a data model specifying the computing resources.";
        }
    }
}

```

```

    leaf storage-id {
        type string;
        description
            "Reference to a storage instance specified in
            a data model specifying the storage resources.";
    }

```



```

    }
  } // network-slice
} // network-slice-node-attributes

grouping network-slice-link-attributes {
  description "Network Slice link scope attributes";
  container network-slice {
    description
      "Containing Network Slice attributes.";
    leaf delay-tolerance {
      type boolean;
      description
        "'true' if is not too critical how long it takes to deliver
        the amount of data.";
      reference
        "GSMA-NS-Template: Generic Network Slice Template,
        Version 3.0.";
    }
    leaf-list periodicity {
      type uint64;
      units seconds;
      description
        "A list of periodicities supported by the network slice.";
      reference
        "GSMA-NS-Template: Generic Network Slice Template,
        Version 3.0.";
    }
    leaf isolation-level {
      type identityref {
        base isolation-level;
      }
      description
        "A network slice instance may be fully or partly, logically
        and/or physically, isolated from another network slice
        instance. This attribute describes different types of
        isolation:";
    }
  } // network-slice
} // network-slice-link-attributes

/*
 * Data nodes
 */

```

```

augment "/nw:networks/nw:network/nw:network-types" {
  description
    "Defines the Network Slice topology type.";
  container network-slice {
    presence "Indicates Network Slice topology";
    description
      "Its presence identifies the Network Slice type.";
  }
}

augment "/nw:networks/nw:network" {
  when "nw:network-types/ns:network-slice" {
    description "Augment only for Network Slice topology.";
  }
  description "Augment topology configuration and state.";
  uses network-slice-topology-attributes;
}

augment "/nw:networks/nw:network/nw:node" {
  when "../nw:network-types/ns:network-slice" {
    description "Augment only for Network Slice topology.";
  }
  description "Augment node configuration and state.";
  uses network-slice-node-attributes;
}

augment "/nw:networks/nw:network/nt:link" {
  when "../nw:network-types/ns:network-slice" {
    description "Augment only for Network Slice topology.";
  }
  description "Augment link configuration and state.";
  uses network-slice-link-attributes;
}
}
<CODE ENDS>

```

6.2. Module ietf-network-slice-connectivity

This module references [\[RFC8345\]](#), [\[RFC8776\]](#), and [\[RFC8795\]](#)

```

<CODE BEGINS> file "ietf-network-slice-connectivity@2022-03-04.yang"
module ietf-network-slice-connectivity {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:"

```

Internet-Draft

IETF Network Slice YANG Model

March 2022

```
+ "ietf-network-slice-connectivity";
prefix "ns-con-types";

import ietf-network {
  prefix "nw";
  reference "RFC 8345: A YANG Data Model for Network Topologies";
}
import ietf-te-topology {
  prefix "tet";
  reference
    "RFC 8795: YANG Data Model for Traffic Engineering (TE)
    Topologies";
}
import ietf-te-types {
  prefix "te-types";
  reference
    "RFC 8776: Traffic Engineering Common YANG Types";
}

organization
  "IETF Traffic Engineering Architecture and Signaling (TEAS)
  Working Group";

contact
  "WG Web:    <http://tools.ietf.org/wg/teas/>
  WG List:    <mailto:teas@ietf.org>

  Editor:     Xufeng Liu
              <mailto:xufeng.liu.ietf@gmail.com>

  Editor:     Luis Miguel Contreras Murillo
              <mailto:luismiguel.contrerasmurillo@telefonica.com>

  Editor:     Sergio Belotti
              <mailto:sergio.belotti@nokia.com>
";

description
  "YANG augmentations to support various connectivity types for
  IETF network slices.
```

Copyright (c) 2022 IETF Trust and the persons identified as

authors of the code. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, is permitted pursuant to, and subject to the license terms contained in, the Simplified BSD License set forth in [Section 4.c](#) of the IETF Trust's Legal Provisions

Relating to IETF Documents
(<http://trustee.ietf.org/license-info>).

This version of this YANG module is part of RFC XXXX; see the RFC itself for full legal notices.";

```
revision 2022-03-04 {
  description "Initial revision";
  reference
    "RFC XXXX: YANG Data Model for Network Slices";
}

/*
 * Groupings
 */
grouping network-slice-connectivity-types {
  description "Network Slice topology scope attributes.";
  list replication-group {
    key "id";
    description
      "A list of replication groups. Each replication group
       contains a list of connectivity constructs
       (that are called connectivity matrix entries in RFC 8795).
       When traffic is sent to one entry in this replication group,
       the traffic is replicated to all other entries in the same
       replication group.";
    leaf id {
      type uint32;
      description
        "Identifies the replication group.";
    }
    leaf-list entry {
      type leafref {
        path "../tet:id";
      }
    }
  }
}
```

```

        description
            "References a connectivity matrix entry that belongs to
            this replication group.";
    }
}
list receiver-constraint-group {
    key "id";
    description
        "A list of receiver constraint groups. Each receiver
        constraint group contains a list of connectivity constructs
        (that are called connectivity matrix entries in RFC 8795).
        When traffic is sent to one or more entries in this
        receiver constraint group, the constraints specified in this

```

```

        receiver constraint group are applied to the receiver-side
        termination points referenced by all entries in this
        receiver constraint group.";
    leaf id {
        type uint32;
        description
            "Identifies the receiver constraint group.";
    }
    leaf-list entry {
        type leafref {
            path "../tet:id";
        }
        description
            "References a connectivity matrix entry that belongs to
            this receiver constraint group..";
    }
    uses te-types:te-bandwidth;
}
}

/*
 * Data nodes
 */
augment "/nw:networks/nw:network/nw:node/tet:te/"
+ "tet:te-node-attributes/tet:connectivity-matrices/"
+ "tet:connectivity-matrix" {
    description "Augment node configuration and state.";
    uses network-slice-connectivity-types;

```

```

    }

    augment "/nw:networks/nw:network/nw:node/tet:te/"
      + "tet:information-source-entry/tet:connectivity-matrices/"
      + "tet:connectivity-matrix" {
        description "Augment node configuration and state.";
        uses network-slice-connectivity-types;
      }
  }
}
<CODE ENDS>

```

7. IANA Considerations

RFC Ed.: In this section, replace all occurrences of 'XXXX' with the actual RFC number (and remove this note).

This document registers the following namespace URIs in the IETF XML registry [[RFC3688](#)]:

Liu, et al.	Expires September 7, 2022	[Page 26]
-------------	---------------------------	-----------

Internet-Draft	IETF Network Slice YANG Model	March 2022
----------------	-------------------------------	------------

```

-----
URI: urn:ietf:params:xml:ns:yang:ietf-network-slice
Registrant Contact: The IESG.
XML: N/A, the requested URI is an XML namespace.
-----

```

This document registers the following YANG modules in the YANG Module Names registry [[RFC6020](#)]:

```

-----
name:          ietf-l3-te-topology
namespace:     urn:ietf:params:xml:ns:yang:ietf-network-slice
prefix:        ns
reference:     RFC XXXX
-----

```

8. Security Considerations

The YANG module specified in this document defines a schema for data that is designed to be accessed via network management protocols such as NETCONF [[RFC6241](#)] or RESTCONF [[RFC8040](#)]. The lowest NETCONF layer

is the secure transport layer, and the mandatory-to-implement secure transport is Secure Shell (SSH) [RFC6242]. The lowest RESTCONF layer is HTTPS, and the mandatory-to-implement secure transport is TLS [RFC8446].

The Network Configuration Access Control Model (NACM) [RFC8341] provides the means to restrict access for particular NETCONF or RESTCONF users to a preconfigured subset of all available NETCONF or RESTCONF protocol operations and content.

There are a number of data nodes defined in this YANG module that are writable/creatable/deletable (i.e., config true, which is the default). These data nodes may be considered sensitive or vulnerable in some network environments. Write operations (e.g., edit-config) to these data nodes without proper protection can have a negative effect on network operations. These are the subtrees and data nodes and their sensitivity/vulnerability:

/nw:networks/nw:network/nw:network-types/ns:network-slice
This subtree specifies the network slice type. Modifying the configurations can make network slice type invalid and cause interruption to IETF network slices.

/nw:networks/nw:network/ns:network-slice
This subtree specifies the topology-wide configurations. Modifying the configurations here can cause traffic

characteristics changed in this IETF network slice and related networks.

/nw:networks/nw:network/nw:node/ns:network-slice
This subtree specifies the configurations of the nodes in a IETF network slice. Modifying the configurations in this subtree can change the traffic characteristics on this node and the related networks.

/nw:networks/nw:network/nt:link/ns:network-slice
This subtree specifies the configurations of the links in a IETF network slice. Modifying the configurations in this subtree can change the traffic characteristics on this link and the related networks.

Some of the readable data nodes in this YANG module may be considered sensitive or vulnerable in some network environments. It is thus important to control read access (e.g., via get, get-config, or notification) to these data nodes. These are the subtrees and data nodes and their sensitivity/vulnerability:

/nw:networks/nw:network/nw:network-types/ns:network-slice
Unauthorized access to this subtree can disclose the network slice type.

/nw:networks/nw:network/ns:network-slice
Unauthorized access to this subtree can disclose the topology-wide states.

/nw:networks/nw:network/nw:node/ns:network-slice
Unauthorized access to this subtree can disclose the operational state information of the nodes in a IETF network slice.

/nw:networks/nw:network/nt:link/ns:network-slice
Unauthorized access to this subtree can disclose the operational state information of the links in a IETF network slice.

[9.](#) Acknowledgements

The TEAS Network Slicing Design Team (NSDT) members included Aijun Wang, Dong Jie, Eric Gray, Jari Arkko, Jeff Tantsura, John E Drake, Luis M. Contreras, Rakesh Gandhi, Ran Chen, Reza Rokui, Ricard Vilalta, Ron Bonica, Sergio Belotti, Tomonobu Niwa, Xuesong Geng, and Xufeng Liu.

[10.](#) References

[10.1.](#) Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), DOI 10.17487/RFC2119, March 1997,

<<https://www.rfc-editor.org/info/rfc2119>>.

- [RFC3688] Mealling, M., "The IETF XML Registry", [BCP 81](#), [RFC 3688](#), DOI 10.17487/RFC3688, January 2004, <<https://www.rfc-editor.org/info/rfc3688>>.
- [RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", [RFC 6020](#), DOI 10.17487/RFC6020, October 2010, <<https://www.rfc-editor.org/info/rfc6020>>.
- [RFC6241] 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>>.
- [RFC6242] Wasserman, M., "Using the NETCONF Protocol over Secure Shell (SSH)", [RFC 6242](#), DOI 10.17487/RFC6242, June 2011, <<https://www.rfc-editor.org/info/rfc6242>>.
- [RFC7950] 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>>.
- [RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", [RFC 8040](#), DOI 10.17487/RFC8040, January 2017, <<https://www.rfc-editor.org/info/rfc8040>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in [RFC 2119](#) Key Words", [BCP 14](#), [RFC 8174](#), DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [RFC8341] Bierman, A. and M. Bjorklund, "Network Configuration Access Control Model", STD 91, [RFC 8341](#), DOI 10.17487/RFC8341, March 2018, <<https://www.rfc-editor.org/info/rfc8341>>.
- [RFC8342] Bjorklund, M., Schoenwaelder, J., Shafer, P., Watsen, K., and R. Wilton, "Network Management Datastore Architecture (NMDA)", [RFC 8342](#), DOI 10.17487/RFC8342, March 2018, <<https://www.rfc-editor.org/info/rfc8342>>.

- [RFC8345] Clemm, A., Medved, J., Varga, R., Bahadur, N., Ananthakrishnan, H., and X. Liu, "A YANG Data Model for Network Topologies", [RFC 8345](#), DOI 10.17487/RFC8345, March 2018, <<https://www.rfc-editor.org/info/rfc8345>>.
- [RFC8346] Clemm, A., Medved, J., Varga, R., Liu, X., Ananthakrishnan, H., and N. Bahadur, "A YANG Data Model for Layer 3 Topologies", [RFC 8346](#), DOI 10.17487/RFC8346, March 2018, <<https://www.rfc-editor.org/info/rfc8346>>.
- [RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", [RFC 8446](#), DOI 10.17487/RFC8446, August 2018, <<https://www.rfc-editor.org/info/rfc8446>>.
- [RFC8776] Saad, T., Gandhi, R., Liu, X., Beeram, V., and I. Bryskin, "Common YANG Data Types for Traffic Engineering", [RFC 8776](#), DOI 10.17487/RFC8776, June 2020, <<https://www.rfc-editor.org/info/rfc8776>>.
- [RFC8795] Liu, X., Bryskin, I., Beeram, V., Saad, T., Shah, H., and O. Gonzalez de Dios, "YANG Data Model for Traffic Engineering (TE) Topologies", [RFC 8795](#), DOI 10.17487/RFC8795, August 2020, <<https://www.rfc-editor.org/info/rfc8795>>.
- [GSMA-NS-Template] GSM Association, "Generic Network Slice Template, Version 3.0", NG.116, May 2020.
- [I-D.ietf-teas-ietf-network-slices] Farrel, A., Drake, J., Rokui, R., Homma, S., Makhijani, K., Contreras, L. M., and J. Tantsura, "Framework for IETF Network Slices", [draft-ietf-teas-ietf-network-slices-07](#) (work in progress), March 2022.

10.2. Informative References

- [RFC7951] Lhotka, L., "JSON Encoding of Data Modeled with YANG", [RFC 7951](#), DOI 10.17487/RFC7951, August 2016, <<https://www.rfc-editor.org/info/rfc7951>>.
- [RFC8340] Bjorklund, M. and L. Berger, Ed., "YANG Tree Diagrams", [BCP 215](#), [RFC 8340](#), DOI 10.17487/RFC8340, March 2018, <<https://www.rfc-editor.org/info/rfc8340>>.

Internet-Draft

IETF Network Slice YANG Model

March 2022

- [RFC8453] Ceccarelli, D., Ed. and Y. Lee, Ed., "Framework for Abstraction and Control of TE Networks (ACTN)", [RFC 8453](#), DOI 10.17487/RFC8453, August 2018, <<https://www.rfc-editor.org/info/rfc8453>>.
- [I-D.ietf-ccamp-otn-topo-yang]
Zheng, H., Busi, I., Liu, X., Belotti, S., and O. G. D. Dios, "A YANG Data Model for Optical Transport Network Topology", [draft-ietf-ccamp-otn-topo-yang-13](#) (work in progress), July 2021.
- [I-D.ietf-i2rs-yang-l2-network-topology]
Dong, J., Wei, X., Wu, Q., Boucadair, M., and A. Liu, "A YANG Data Model for Layer 2 Network Topologies", [draft-ietf-i2rs-yang-l2-network-topology-18](#) (work in progress), September 2020.
- [I-D.ietf-teas-actn-yang]
Lee, Y., Zheng, H., Ceccarelli, D., Yoon, B. Y., and S. Belotti, "Applicability of YANG models for Abstraction and Control of Traffic Engineered Networks", [draft-ietf-teas-actn-yang-08](#) (work in progress), September 2021.
- [I-D.ietf-teas-ietf-network-slice-nbi-yang]
Wu, B., Dhody, D., Rokui, R., Saad, T., and L. Han, "IETF Network Slice Service YANG Model", [draft-ietf-teas-ietf-network-slice-nbi-yang-01](#) (work in progress), March 2022.
- [I-D.contreras-teas-slice-controller-models]
Contreras, L. M., Rokui, R., Tantsura, J., Wu, B., Liu, X., Dhody, D., and S. Bellotti, "IETF Network Slice Controller and its associated data models", [draft-contreras-teas-slice-controller-models-02](#) (work in progress), March 2022.

[Appendix A](#). Data Tree for the Example in [Section 3.1](#).

[A.1](#). Native Topology

This section contains an example of an instance data tree in the JSON encoding [[RFC7951](#)]. The example instantiates "ietf-network" for the native topology depicted in Figure 3.

```
{
  "ietf-network:networks": {
    "network": [
      {
        "network-id": "example-native-topology",
        "network-types": {
        },
        "node": [
          {
            "node-id": "R1",
            "ietf-network-topology:termination-point": [
              {
                "tp-id": "1-0-1"
              },
              {
                "tp-id": "1-0-2"
              },
              {
                "tp-id": "1-2-1"
              },
              {
                "tp-id": "1-2-2"
              }
            ]
          },
          {
            "node-id": "R2",
            "ietf-network-topology:termination-point": [
```

```

    {
      "tp-id":"2-1-1"
    },
    {
      "tp-id":"2-1-2"
    },
    {
      "tp-id":"2-3-1"
    },
    {
      "tp-id":"2-4-1"
    }
  }

```

```

    ]
  },
  {
    "node-id":"R3",
    "ietf-network-topology:termination-point": [
      {
        "tp-id":"3-0-1"
      },
      {
        "tp-id":"3-2-1"
      }
    ]
  },
  {
    "node-id":"R4",
    "ietf-network-topology:termination-point": [
      {
        "tp-id":"4-0-1"
      },
      {
        "tp-id":"4-2-1"
      }
    ]
  }
],
"ietf-network-topology:link": [
  {
    "link-id":"R1,1-0-1,,",
    "source": {

```

```

        "source-node": "R1",
        "source-tp": "1-0-1"
    },
    {
        "link-id": ",,R1,1-0-1",
        "destination": {
            "dest-node": "R1",
            "dest-tp": "1-0-1"
        }
    },
    {
        "link-id": "R1,1-0-2,,",
        "source": {
            "source-node": "R1",
            "source-tp": "1-0-2"
        }
    },
    {

```

```

        "link-id": ",,R1,1-0-2",
        "destination": {
            "dest-node": "R1",
            "dest-tp": "1-0-2"
        }
    },
    {
        "link-id": "R1,1-2-1,R2,2-1-1",
        "source": {
            "source-node": "R1",
            "source-tp": "1-2-1"
        },
        "destination": {
            "dest-node": "R2",
            "dest-tp": "2-1-1"
        }
    },
    {
        "link-id": "R2,2-1-1,R1,1-2-1",
        "source": {
            "source-node": "R2",
            "source-tp": "2-1-1"
        }
    }

```

```

    },
    "destination": {
      "dest-node": "R1",
      "dest-tp": "1-2-1"
    }
  },
  {
    "link-id": "R1,1-2-2,R2,2-1-2",
    "source": {
      "source-node": "R1",
      "source-tp": "1-2-2"
    },
    "destination": {
      "dest-node": "R2",
      "dest-tp": "2-1-2"
    }
  },
  {
    "link-id": "R2,2-1-2,R1,1-2-2",
    "source": {
      "source-node": "R2",
      "source-tp": "2-1-2"
    },
    "destination": {
      "dest-node": "R1",
      "dest-tp": "1-2-2"
    }
  }
}

```

```

  }
},
{
  "link-id": "R2,2-3-1,R3,3-2-1",
  "source": {
    "source-node": "R2",
    "source-tp": "2-3-1"
  },
  "destination": {
    "dest-node": "R3",
    "dest-tp": "3-2-1"
  }
},
{
  "link-id": "R3,3-2-1,R2,2-3-1",

```

```

    "source": {
      "source-node": "R3",
      "source-tp": "3-2-1"
    },
    "destination": {
      "dest-node": "R2",
      "dest-tp": "2-3-1"
    }
  },
  {
    "link-id": "R2,2-4-1,R4,4-2-1",
    "source": {
      "source-node": "R2",
      "source-tp": "2-4-1"
    },
    "destination": {
      "dest-node": "R4",
      "dest-tp": "4-2-1"
    }
  },
  {
    "link-id": "R4,4-2-1,R2,2-4-1",
    "source": {
      "source-node": "R4",
      "source-tp": "4-2-1"
    },
    "destination": {
      "dest-node": "R2",
      "dest-tp": "2-4-1"
    }
  },
  {
    "link-id": "R3,3-0-1,,",

```

```

    "source": {
      "source-node": "R3",
      "source-tp": "3-0-1"
    }
  },
  {
    "link-id": "", "R3,3-0-1",
    "destination": {

```



```

        "dest-node": "R3",
        "dest-tp": "3-0-1"
      }
    },
    {
      "link-id": "R4,4-0-1,,",
      "source": {
        "source-node": "R4",
        "source-tp": "4-0-1"
      }
    },
    {
      "link-id": ",,R4,4-0-1",
      "destination": {
        "dest-node": "R4",
        "dest-tp": "4-0-1"
      }
    }
  ]
}

```

A.2. Network Slice Blue

This section contains an example of an instance data tree in the JSON encoding [[RFC7951](#)]. The example instantiates "ietf-network-slice" for the topology customized for Network Slice Blue depicted in Figure 3.

```

{
  "ietf-network:networks": {
    "network": [
      {
        "network-id": "example-customized-blue-topology",
        "network-types": {
          "ietf-network-slice:network-slice": {

```

```

    }

```

```

},
"supporting-network": [
  {
    "network-ref": "example-native-topology"
  }
],
"node": [
  {
    "node-id": "VR1",
    "supporting-node": [
      {
        "network-ref": "example-native-topology",
        "node-ref": "R1"
      }
    ],
    "ietf-network-slice:network-slice": {
      "isolation-level":
        "ietf-network-slice:physical-memory-isolation"
    },
    "ietf-network-topology:termination-point": [
      {
        "tp-id": "1-0-1"
      },
      {
        "tp-id": "1-3-1"
      }
    ]
  },
  {
    "node-id": "VR3",
    "supporting-node": [
      {
        "network-ref": "example-native-topology",
        "node-ref": "R2"
      }
    ],
    "ietf-network-slice:network-slice": {
      "isolation-level":
        "ietf-network-slice:physical-memory-isolation"
    },
    "ietf-network-topology:termination-point": [
      {
        "tp-id": "3-1-1"
      },
      {
        "tp-id": "3-5-1"
      }
    ]
  }
]

```

```
    ]
  },
  {
    "node-id": "VR5",
    "supporting-node": [
      {
        "network-ref": "example-native-topology",
        "node-ref": "R3"
      }
    ],
    "ietf-network-slice:network-slice": {
      "isolation-level":
        "ietf-network-slice:physical-memory-isolation"
    },
    "ietf-network-topology:termination-point": [
      {
        "tp-id": "5-3-1"
      },
      {
        "tp-id": "5-0-1"
      }
    ]
  }
],
"ietf-network-topology:link": [
  {
    "link-id": "VR1,1-0-1,,",
    "source": {
      "source-node": "VR1",
      "source-tp": "1-0-1"
    },
    "supporting-link": [
      {
        "network-ref": "example-native-topology",
        "link-ref": "R1,1-0-1,,"
      }
    ],
    "ietf-network-slice:network-slice": {
      "isolation-level":
        "ietf-network-slice:physical-network-isolation"
    }
  },
  {
    "link-id": ",,VR1,1-0-1",
    "destination": {
      "dest-node": "VR1",
```

```
    "dest-tp": "1-0-1"
  },
```

```
    "supporting-link": [
      {
        "network-ref": "example-native-topology",
        "link-ref": ",,R1,1-0-1"
      }
    ],
    "ietf-network-slice:network-slice": {
      "isolation-level":
        "ietf-network-slice:physical-network-isolation"
    }
  },
  {
    "link-id": "VR1,1-3-1,VR3,3-1-1",
    "source": {
      "source-node": "VR1",
      "source-tp": "1-3-1"
    },
    "destination": {
      "dest-node": "VR3",
      "dest-tp": "3-1-1"
    },
    "supporting-link": [
      {
        "network-ref": "example-native-topology",
        "link-ref": "R1,1-2-1,R2,2-1-1"
      }
    ],
    "ietf-network-slice:network-slice": {
      "isolation-level":
        "ietf-network-slice:physical-network-isolation"
    }
  },
  {
    "link-id": "VR3,3-1-1,VR1,1-3-1",
    "source": {
      "source-node": "VR3",
      "source-tp": "3-1-1"
    },
    "destination": {
```

```

        "dest-node": "R1",
        "dest-tp": "1-3-1"
    },
    "supporting-link": [
        {
            "network-ref": "example-native-topology",
            "link-ref": "R2,2-1-1,R1,1-2-1"
        }
    ],

```

```

    "ietf-network-slice:network-slice": {
        "isolation-level":
            "ietf-network-slice:physical-network-isolation"
    }
},
{
    "link-id": "VR3,3-5-1,VR5,5-3-1",
    "source": {
        "source-node": "VR3",
        "source-tp": "3-5-1"
    },
    "destination": {
        "dest-node": "VR5",
        "dest-tp": "5-3-1"
    },
    "supporting-link": [
        {
            "network-ref": "example-native-topology",
            "link-ref": "R2,2-3-1,R3,3-2-1"
        }
    ],
    "ietf-network-slice:network-slice": {
        "isolation-level":
            "ietf-network-slice:physical-network-isolation"
    }
},
{
    "link-id": "VR5,5-3-1,VR3,3-5-1",
    "source": {
        "source-node": "VR5",
        "source-tp": "5-3-1"
    },
    "destination": {
        "dest-node": "VR3",
        "dest-tp": "3-5-1"
    },
    "supporting-link": [
        {
            "network-ref": "example-native-topology",
            "link-ref": "R2,2-3-1,R3,3-2-1"
        }
    ],
    "ietf-network-slice:network-slice": {
        "isolation-level":
            "ietf-network-slice:physical-network-isolation"
    }
},

```

```

    "destination": {
      "dest-node": "VR3",
      "dest-tp": "3-5-1"
    },
    "supporting-link": [
      {
        "network-ref": "example-native-topology",
        "link-ref": "R3,3-2-1,R2,2-3-1"
      }
    ],
    "ietf-network-slice:network-slice": {
      "isolation-level":
        "ietf-network-slice:physical-network-isolation"
    }
  },
  {

```

```

    "link-id": "VR5,5-0-1,,",
    "source": {
      "source-node": "VR5",
      "source-tp": "5-0-1"
    },
    "supporting-link": [
      {
        "network-ref": "example-native-topology",
        "link-ref": "R3,3-0-1,,"
      }
    ],
    "ietf-network-slice:network-slice": {
      "isolation-level":
        "ietf-network-slice:physical-network-isolation"
    }
  },
  {
    "link-id": ",,VR5,5-0-1",
    "destination": {
      "dest-node": "VR5",
      "dest-tp": "5-0-1"
    },
    "supporting-link": [
      {
        "network-ref": "example-native-topology",

```

```

        "link-ref":",,R3,3-0-1"
    }
],
"ietf-network-slice:network-slice": {
    "isolation-level":
    "ietf-network-slice:physical-network-isolation"
}
}
],
"ietf-network-slice:network-slice": {
    "optimization-criterion":
    "ietf-te-types:of-minimize-cost-path",
    "isolation-level":
    "ietf-network-slice:physical-isolation"
}
}
]
}
}

```

Authors' Addresses

Xufeng Liu
IBM Corporation

EMail: xufeng.liu.ietf@gmail.com

Jeff Tantsura
Microsoft

EMail: jefftant.ietf@gmail.com

Igor Bryskin
Individual

EMail: i_bryskin@yahoo.com

Luis Miguel Contreras Murillo
Telefonica

EMail: luismiguel.contrerasmurillo@telefonica.com

Qin Wu
Huawei

EMail: bill.wu@huawei.com

Sergio Belotti
Nokia

EMail: sergio.belotti@nokia.com

Reza Rokui
Ciena
Canada

EMail: rrokui@Ciena.com