Data Models Configuration Guide for Cisco NCS 1002, Cisco IOS XR Releases

First Published: 2016-03-14

Last Modified: 2023-09-22

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED “AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at www.cisco.com/go/offices.

CONTENTS

CHAPTER 1

Data models

Data modeling is standard based approach to model configuration and operational data in networking devices. Using data models, customers can automate and simplify network wide visibility and configuration.

Data Models - Programmatic and Standards-based Configuration

Cisco IOS XR software supports the automation of configuration of multiple routers across the network using Data models. Configuring routers using data models overcomes drawbacks posed by traditional router management techniques.

CLIs are widely used for configuring a router and for obtaining router statistics. Other actions on the router, such as, switch-over, reload, process restart are also CLI-based. Although, CLIs are heavily used, they have many restrictions.

Customer needs are fast evolving. Typically, a network center is a heterogenous mix of various devices at multiple layers of the network. Bulk and automatic configurations need to be accomplished. CLI scraping is not flexible and optimal. Re-writing scripts many times, even for small configuration changes is cumbersome. Bulk configuration changes through CLIs are error-prone and may cause system issues. The solution lies in using data models - a programmatic and standards-based way of writing configurations to any network device, replacing the process of manual configuration. Data models are written in a standard, industry-defined language. Although configurations using CLIs are easier (more human-friendly), automating the configuration using data models results in scalability.

Cisco IOS XR supports the YANG data modeling language. YANG can be used with Network Configuration Protocol (NETCONF) to provide the desired solution of automated and programmable network operations.

YANG model

YANG is a data modeling language used to describe configuration and operational data, remote procedure calls and notifications for network devices. The salient features of YANG are:

- Human-readable format, easy to learn and represent

- Supports definition of operations
- Reusable types and groupings
- Data modularity through modules and submodules
- Supports the definition of operations (RPCs)
- Well-defined versioning rules
- Extensibility through augmentation

For more details of YANG, refer RFC 6020 and 6087.

NETCONF and gRPC (Google Remote Procedute Call) provide a mechanism to exchange configuration and operational data between a client application and a router and the YANG models define a valid structure for the data (that is being exchanged).

Protocol	Transport	Encoding/ Decoding
NETCONF	SSH	XML
gRPC	HTTP/2	XML, JSON

Each feature has a defined YANG model. Cisco-specific YANG models are referred to as synthesized models. Some of the standard bodies, such as IETF , IEEE and Open Config, are working on providing an industry-wide standard YANG models that are referred to as common models.

Components of a YANG Model

A module defines a single data model. However, a module can reference definitions in other modules and submodules by using the **import** statement to import external modules or the **include** statement to include one or more submodules. A module can provide augmentations to another module by using the **augment** statement to define the placement of the new nodes in the data model hierarchy and the **when** statement to define the conditions under which the new nodes are valid. **Prefix** is used when referencing definitions in the imported module.

YANG models are available for configuring a feature and to get operational state (similar to show commands)

This is the configuration YANG model for AAA (denoted by - cfg)

```
(snippet)
module Cisco-IOS-XR-aaa-locald-cfg {

  /*** NAMESPACE / PREFIX DEFINITION ***/

  namespace "http://cisco.com/ns/yang/Cisco-IOS-XR-aaa-locald-cfg";


  prefix "aaa-locald-cfg";

  /*** LINKAGE (IMPORTS / INCLUDES) ***/

  import Cisco-IOS-XR-types { prefix "xr"; }

  import Cisco-IOS-XR-aaa-lib-cfg { prefix "a1"; }

  /*** META INFORMATION ***/
  organization "Cisco Systems, Inc.";
          ........................
          ........................ (truncated)
```

This is the operational YANG model for AAA (denoted by -oper)

```
 (snippet)
module Cisco-IOS-XR-aaa-locald-oper {

  /*** NAMESPACE / PREFIX DEFINITION ***/

  namespace "http://cisco.com/ns/yang/Cisco-IOS-XR-aaa-locald-oper";


  prefix "aaa-locald-oper";

  /*** LINKAGE (IMPORTS / INCLUDES) ***/

  import Cisco-IOS-XR-types { prefix "xr"; }

  include Cisco-IOS-XR-aaa-locald-oper-sub1 {
    revision-date 2015-01-07;
  }

  /*** META INFORMATION ***/

  organization "Cisco Systems, Inc.";
    .......................
    ....................... (truncated)
```

Note A module may include any number of sub-modules, but each sub-module may belong to only one module. The names of all standard modules and sub-modules must be unique.

Structure of YANG models

YANG data models can be represented in a hierarchical, tree-based structure with nodes, which makes them more easily understandable. YANG defines four nodes types. Each node has a name, and depending on the node type, the node might either define a value or contain a set of child nodes. The nodes types (for data modeling) are:

- leaf node - contains a single value of a specific type
- list node - contains a sequence of list entries, each of which is uniquely identified by one or more key leafs
- leaf-list node - contains a sequence of leaf nodes
- container node - contains a grouping of related nodes containing only child nodes, which can be any of the four node types

Data Types

YANG defines data types for leaf values. These data types help the user in understanding the relevant input for a leaf.

Name	Description
binary	Any binary data
bits	A set of bits or flags
boolean	"true" or "false"
decimal64	64-bit signed decimal number
empty	A leaf that does not have any value
enumeration	Enumerated strings
identityref	A reference to an abstract identity
instance-identifier	References a data tree node
int (integer-defined values)	8-bit, 16-bit, 32-bit, 64-bit signed integers
leafref	A reference to a leaf instance
uint	8-bit, 16-bit, 32-bit, 64-bit unsigned intergers
string	Human-readable string
union	Choice of member types

Data Model and CLI Comparison

Each feature has a defined YANG model that is synthesized from the schemas. A model in a tree format includes:

- Top level nodes and their subtrees
- Subtrees that augment nodes in other yang models
- Custom RPCs

The options available using the CLI are defined as leaf-nodes in data models. The defined data types, indicated corresponding to each leaf-node, help the user to understand the required inputs.

Supported YANG Models in NCS 1004

The supported config and oper YANG models for NCS 1004 are listed below:

Config Yang Models	Oper Yang Models
Cisco-IOS-XR-osa-cfg.yang	Cisco-IOS-XR-osa-oper.yang
Cisco-IOS-XR-controller-optics-cfg.yang	Cisco-IOS-XR-controller-optics-oper.yang
Cisco-IOS-XR-pmengine-cfg.yang	Cisco-IOS-XR-pmengine-oper.yang

Config Yang Models	Oper Yang Models
Cisco-IOS-XR-ethernet-lldp-cfg.yang	Cisco-IOS-XR-ethernet-lldp-oper.yang
Cisco-IOS-XR-ifmgr-cfg.yang	Cisco-IOS-XR-telemetry-model-driven-oper.yang
Cisco-IOS-XR-telemetry-model-driven-cfg.yang	Cisco-IOS-XR-fpd-infra-oper.yang
Cisco-IOS-XR-fpd-infra-cfg.yang	Cisco-IOS-XR-ikev2-oper.yang
Cisco-IOS-XR-ikev2-cfg.yang	Cisco-IOS-XR-otnsec-oper.yang
Cisco-IOS-XR-um-ncs1004-hw-module-osa-cfg	

The supported versions of Open Config model are listed below:

- openconfig-platform.yang
- openconfig-platform-transceiver.yang
- openconfig-terminal-device.yang
- openconfig-interfaces.yang
- openconfig-system.yang

Note openconfig-platform-transceiver.yang model is the augmented model of openconfig-platform.yang model.

Enabling Netconf

This task enables Netconf over SSH.

Before you begin

- Install the required packages (k9sec and mgbl)
- Generate relevant crypto keys

Step 1 **netconf-yang agent ssh**

Enables the Netconf agent process.

Step 2 **ssh server netconf**

Enables Netconf.

Step 3 **ssh server v2**

Enables SSH on the device and enables Netconf on port 22 if the Netconf agent process is enabled.

What to do next

The **netconf-yang agent session** command enables the user to set session parameters.

```
netconf-yang agent session {limit value |  absolute-timeout value |  idle-timeout value}
```

where,

- **limit** *value*- sets the maximum count for concurrent netconf-yang sessions. Range is 1 to 1024. The default value is 50.
- **absolute-timeout** *value*- sets the absolute session lifetime. Range is 1 to 1440 (in minutes).
- **idle-timeout** *value*- sets the idle session lifetime. Range is 1 to 1440 (in minutes).

gRPC

gRPC is a language-neutral, open source, RPC (Remote Procedute Call) system developed by Google. By default, it uses protocol buffers as the binary serialization protocol. It can be used with other serialization protocols as well such as JSON, XML etc. The user needs to define the structure by defining protocol buffer message types in*.proto* files. Each protocol buffer message is a small logical record of information, containing a series of name-value pairs.

gRPC encodes requests and responses in binary. Although Protobufs was the only format supported in the initial release, gRPC is extensible to other content types. The Protobuf binary data object in gRPC is transported using HTTP/2 (RFC 7540). HTTP/2 is a replacement for HTTP that has been optimized for high performance. HTTP/2 provides many powerful capabilities including bidirectional streaming, flow control, header compression and multi-plexing. gRPC builds on those features, adding libraries for application-layer flow-control, load-balancing and call-cancellation.

gRPC supports distributed applications and services between a client and server. gRPC provides the infrastructure to build a device management service to exchange configuration and operational data between a client and a server in which the structure of the data is defined by YANG models.

Cisco gRPC IDL

The protocol buffers interface definition language (IDL) is used to define service methods, and define parameters and return types as protocol buffer message types.

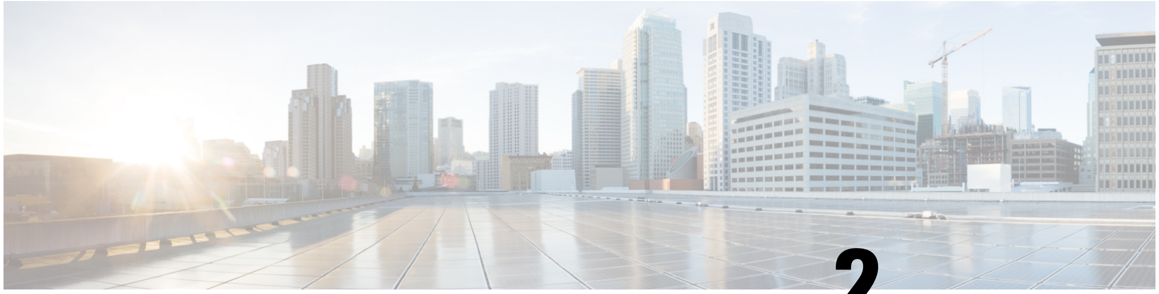
gRPC requests can be encoded and sent across to the router using JSON. gRPC IDL also supports the exchange of CLI.

For gRPC transport, gRPC IDL is defined in .proto format. Clients can invoke the RPC calls defined in the IDL to program XR. The supported operations are - Get, Merge, Delete, Replace. The gRPC JSON arguments are defined in the IDL.

```
syntax = "proto3";

package IOSXRExtensibleManagabilityService;

service gRPCConfigOper {

    rpc GetConfig(ConfigGetArgs) returns(stream ConfigGetReply) {};

    rpc MergeConfig(ConfigArgs) returns(ConfigReply) {};

    rpc DeleteConfig(ConfigArgs) returns(ConfigReply) {};
    rpc ReplaceConfig(ConfigArgs) returns(ConfigReply) {};

    rpc CliConfig(CliConfigArgs) returns(CliConfigReply) {};

}
```

gRPC Operations

- oper get-config—Retrieves a configuration
- oper merge-config— Appends to an existing configuration
- oper delete-config—Deletes a configuration
- oper replace-config—Modifies a part of an existing configuration
- oper get-oper—Gets operational data using JSON
- oper cli-config—Performs a configuration
- oper showcmdtextoutput



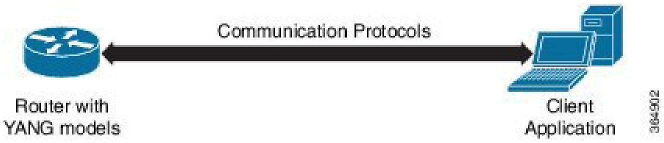
CHAPTER 2

Using Data models

This section explains the required configurations and procedures for using data models.

Using Data models

Figure 1: Workflow for using Data models



The above illustration gives a quick snap shot of how YANG can be used with Netconf in configuring a network device using a client application.

The tasks that help the user to implement Data model configuration are listed here.

1. Load the software image ; the YANG models are a part of the software image. Alternatively, the YANG models can also be downloaded from:

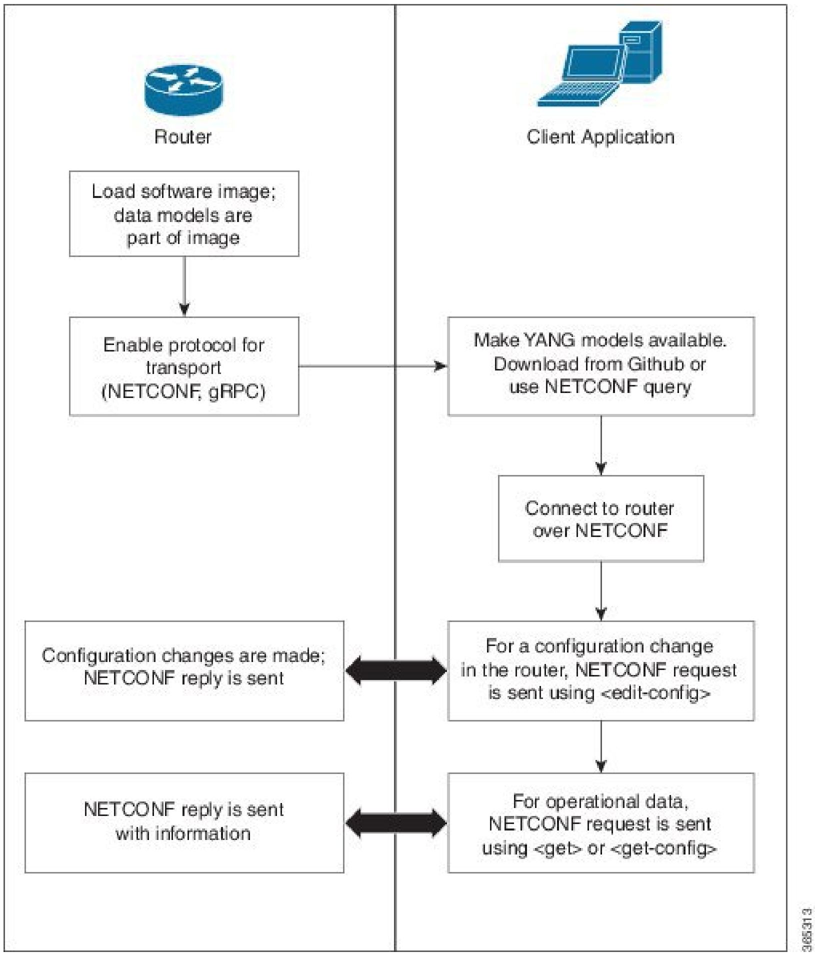
   ```
   https://github.com/YangModels/yang/tree/master/vendor/cisco/xr
   ```

   Users can also query using NETCONF to get the list of models.

   ```
   <?xml version="1.0" encoding="utf-8"?>
   <rpc message-id="100" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
       <get>
           <filter type="subtree">
               <netconf-state xmlns="urn:ietf:params:xml:ns:yang:ietf-netconf-monitoring">

                   <schemas/>
               </netconf-state>
           </filter>
       </get>
   </rpc>
   ```

2. Communication between the router and the application happens by SSH on Netconf. Enable Netconf on the router on a suitable port.
3. From the client application, connect to the router using Netconf on SSH. Run Netconf operations to make configuration changes or get operational data.

Figure 2: Lane Diagram to show the router and client application operations


Router
Client Application
Load software image; data models are part of image
Enable protocol for transport (NETCONF, gRPC)
Make YANG models available. Download from Github or use NETCONF query
Connect to router over NETCONF
Configuration changes are made; NETCONF reply is sent
For a configuration change in the router, NETCONF request is sent using <edit-config>
NETCONF reply is sent with information
For operational data, NETCONF request is sent using <get> or <get-config>
365313

Enabling Netconf

This task enables Netconf over SSH.

Before you begin

- Install the required packages (k9sec and mgbl)
- Generate relevant crypto keys

Step 1 **netconf-yang agent ssh**

Enables the Netconf agent process.

Step 2 **ssh server netconf**

Enables Netconf.

Step 3 **ssh server v2**

Enables SSH on the device and enables Netconf on port 22 if the Netconf agent process is enabled.

What to do next

The **netconf-yang agent session** command enables the user to set session parameters.

```
netconf-yang agent session {limit value |  absolute-timeout value |  idle-timeout value}
```

where,

- **limit** *value*- sets the maximum count for concurrent netconf-yang sessions. Range is 1 to 1024. The default value is 50.
- **absolute-timeout** *value*- sets the absolute session lifetime. Range is 1 to 1440 (in minutes).
- **idle-timeout** *value*- sets the idle session lifetime. Range is 1 to 1440 (in minutes).

Enabling gRPC

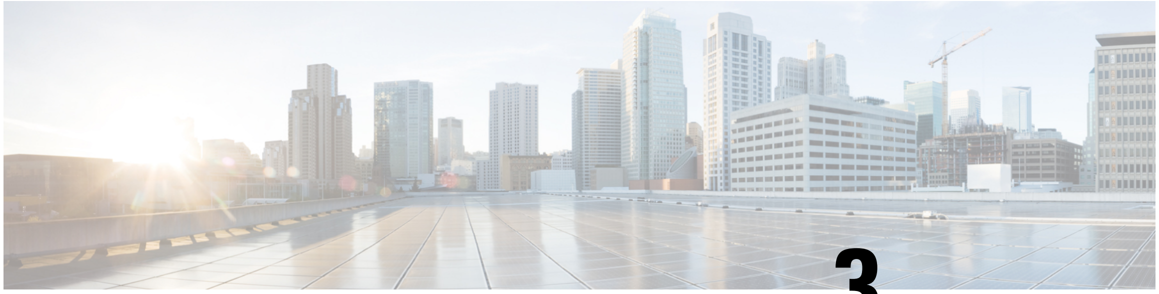
Use the following procedure to enable gRPC over HTTPS/2. gRPC supports both, the IPv4 and IPv6 address families (default is IPv4).

Step 1 Install the GO client. For more details on installing the GO client, see https://golang.org/doc/install.

Step 2 Configure the gRPC port, using the **grpc port** command.

```
RP/0/RP0/CPU0:ios(config)#grpc
RP/0/RP0/CPU0:ios(config)#port 57400
RP/0/RP0/CPU0:ios(config)#tls
RP/0/RP0/CPU0:ios(config)#commit
```

Port can range from 57344 to 57999. If a port is unavailable, an error is displayed.



CHAPTER 3

Configuring NCS 1002 Using Data Models

This section includes examples for configuring NCS 1002 using Data models.

Supported YANG Models in NCS 1002

The supported config and oper YANG models for NCS 1002 are listed below:

Cfg. yang	**Oper. yang**
Cisco-IOS-XR-pmengine-cfg.yang	Cisco-IOS-XR-pmengine-oper.yang
Cisco-IOS-XR-controller-optics-cfg.yang	Cisco-IOS-XR-controller-optics-oper.yang
Cisco-IOS-XR-controller-otu-cfg.yang	Cisco-IOS-XR-controller-otu-oper.yang
Cisco-IOS-XR-ncs1k-mxp-cfg	Cisco-IOS-XR-alarmgr-server-oper.yang
Cisco-IOS-XR-lib-keychain-macsec-cfg	Cisco-IOS-XR-ncs1k-mxp-headless-oper.yang
Cisco-IOS-XR-crypto-macsec-mka-cfg	Cisco-IOS-XR-plat-chas-invmgr-oper.yang
Cisco-IOS-XR-ifmgr-cfg	Cisco-IOS-XR-ncs1k-mxp-lldp-oper.yang Cisco-IOS-XR-pfi-im-cmd-ctrlr-oper.yang

Cfg. yang	Oper. yang
Cisco-IOS-XR-crypto-macsec-mka-if-cfg	Cisco-IOS-XR-crypto-macsec-mka-oper.yang Cisco-IOS-XR-crypto-macsec-secy-oper.yang

The supported versions of Open Config model are listed below:

- openconfig-platform.yang 0.4.0
- openconfig-platform-transceiver.yang 0.1.0
- openconfig-terminal-device.yang 0.3.0
- openconfig-interfaces.yang 1.0.2

Configure Slice

Step 1 Use the Cisco-IOS-XR-ncs1k-mxp-cfg.yang YANG model for provisioning the slice with traffic on the client and trunk ports.

All the five client ports of the slice need to be configured at the same bitrate except for mixed mode configuration. Both the trunk ports are always set with the same FEC mode. In mixed mode configuration, the client ports are configured at different bitrates.

YANG model	Example
Cisco-IOS-XR-ncs1k-mxp-cfg.yang	<?xml version="1.0"?> <rpc message-id="102" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"> <edit-config> <target> <candidate/> </target> <config xmlns:xc="urn:ietf:params:xml:ns:netconf:base:1.0"> <hardware-module xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-ncs1k-mxp-cfg"> <node> <location>0_RP0_CPU0</location> <values> <value> <slice-id >3</slice-id> <client-rate>ten-gig</client-rate> <trunk-rate>two-hundred-gig</trunk-rate> <fec>sd20</fec> </value> <value> <slice-id >2</slice-id> <client-rate>ten-gig</client-rate> <trunk-rate>two-hundred-gig</trunk-rate> <fec>sd20</fec> </value> <value> <slice-id >1</slice-id> <client-rate>ten-gig</client-rate> <trunk-rate>two-hundred-gig</trunk-rate> <fec>sd20</fec> </value>

YANG model	Example
	<value> <slice-id >0</slice-id> <client-rate>ten-gig</client-rate> <trunk-rate>two-hundred-gig</trunk-rate> <fec>sd20</fec> </value> </values> </node> </hardware-module> <interface-configurations xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-ifmgr-cfg"> <interface-configuration> <active>act</active> <interface-name>Optics0/0/0/2</interface-name> <optics xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-controller-optics-cfg"> <optics-dwdm-carrier> <grid-type>50g-hz-grid</grid-type> <param-type>itu-ch</param-type> <param-value>1</param-value> </optics-dwdm-carrier> </optics> </interface-configuration> </interface-configurations> </config> </edit-config> </rpc>

YANG model	Example
Cisco-IOS-XR-ncs1k-mxp-cfg.yang	<?xml version="1.0"?> <rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"> <edit-config> <target> <candidate/> </target> <config xmlns:xc="urn:ietf:params:xml:ns:netconf:base:1.0"> <hardware-module xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-ncs1k-mxp-cfg"> <node> <location>0_RP0_CPU0</location> <slice> <values> <client-rate>ten-and-hundred-gig</client-rate> <trunk-rate>two-hundred-gig</trunk-rate> <fec>sd7</fec> </values> <slice-id>0</slice-id> </slice> </node> </hardware-module> </config> </edit-config> </rpc>

Step 2 Use the Cisco-IOS-XR-ncs1k-mxp-oper.yang YANG model to verify the slice configuration.

YANG model	Example

Cisco-IOS-XR-ncs1k-mxp-oper.yang	<?xml version="1.0" ?>
	<rpc message-id="856612"
	xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
	<get>
	<filter>
	<hw-module
	xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-ncs1k-mxp-oper" >
	   <slice-all>
	    <slice-info>
	     <slice-id>0</slice-id>
	    </slice-info>
	   </slice-all>
	
	   <slice-all>
	    <slice-info>
	     <slice-id>1</slice-id>
	    </slice-info>
	   </slice-all>
	
	   <slice-all>
	    <slice-info>
	     <slice-id>2</slice-id>
	    </slice-info>
	   </slice-all>
	
	    <slice-all>
	    <slice-info>
	     <slice-id>3</slice-id>
	    </slice-info>
	   </slice-all>
	
	 </hw-module>
	</filter>
	</get>
	</rpc>

Configure Optics Controller

Step 1 Use the Cisco-IOS-XR-ifmgr-cfg.yang YANG model for configuring the optics controller.

YANG model	Example
Cisco-IOS-XR-ifmgr-cfg.yang	<?xml version="1.0"?> <rpc message-id="102" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"> <edit-config> <target> <candidate/> </target> <config xmlns:xc="urn:ietf:params:xml:ns:netconf:base:1.0"> <interface-configurations xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-ifmgr-cfg"> <interface-configuration> <active>act</active>

YANG model	Example
	<interface-name>Optics0/0/0/5</interface-name> <shutdown></shutdown> </interface-configuration> <interface-configuration> <active>act</active> <interface-name>Optics0/0/0/6</interface-name> <shutdown></shutdown> </interface-configuration> <interface-configuration> <active>act</active> <interface-name>Optics0/0/0/12</interface-name> <shutdown></shutdown> </interface-configuration> <interface-configuration> <active>act</active> <interface-name>Optics0/0/0/13</interface-name> <shutdown></shutdown> </interface-configuration> <interface-configuration> <active>act</active> <interface-name>Optics0/0/0/19</interface-name> <shutdown></shutdown> </interface-configuration> <interface-configuration> <active>act</active> <interface-name>Optics0/0/0/20</interface-name> <shutdown></shutdown> </interface-configuration> <interface-configuration> <active>act</active> <interface-name>Optics0/0/0/26</interface-name> <shutdown></shutdown> </interface-configuration> <interface-configuration> <active>act</active> <interface-name>Optics0/0/0/27</interface-name> <shutdown></shutdown> </interface-configuration> </interface-configurations> </config> </edit-config> </rpc>

Step 2 Use the Cisco-IOS-XR-controller-optics-cfg.yang YANG model for configuring the wavelength on the trunk port.

YANG model	Example

Cisco-IOS-XR-controller-optics-cfg.yang	<?xml version="1.0"?>
	<rpc message-id="102" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
	<edit-config>
	<target>
	<candidate/>
	</target>
	<config xmlns:xc="urn:ietf:params:xml:ns:netconf:base:1.0">
	   <interface-configurations
	      xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-ifmgr-cfg">
	      <interface-configuration>
	      <active>act</active>
	          <interface-name>Optics0/0/0/2</interface-name>
	          <optics
	xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-controller-optics-cfg">

YANG model	Example
	<optics-dwdm-carrier> <grid-type>50g-hz-grid</grid-type> <param-type>itu-ch</param-type> <param-value>1</param-value> </optics-dwdm-carrier> </optics> </interface-configuration> </interface-configurations> </config> </edit-config> </rpc>

Step 3 Use the Cisco-IOS-XR-controller-optics-oper.yang YANG model to verify the wavelength and channel mapping for trunk optics controllers.

YANG model	Example

Cisco-IOS-XR-controller-optics-oper.yang	<?xml version="1.0" ?>
	<rpc message-id="8566" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
	<get>
	<filter type="subtree">
	
	  <optics-oper
	xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-controller-optics-oper">
	
	   <optics-ports>
	    <optics-port>
	     <name>Optics0/0/0/13</name>
	     <optics-dwdm-carrrier-channel-map>
	
	     </optics-dwdm-carrrier-channel-map>
	     </optics-port>
	     </optics-ports>
	     </optics-oper>
	
	</filter>
	</get>
	</rpc>

Step 4 Use the Cisco-IOS-XR-pfi-im-cmd-ctrlr-oper.yang YANG model to display the name, status, and port description of the optics controller.

YANG model	Example

Cisco-IOS-XR-pfi-im-cmd-ctrlr-oper.yang	
	<?xml version="1.0" ?>
	<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
	<get>
	<filter>
	<controllers
	xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-pfi-im-cmd-ctrlr-oper">
	
	     <controllers>
	        <controller>
	            <interafce-name>Optics0/0/0/12
	            </interafce-name>
	        </controller>
	     </controllers>
	</controllers>
	</filter>

YANG model	Example
	</get> </rpc>

Configure Ethernet and Coherent DSP Controllers

Step 1 Use the Cisco-IOS-XR-ifmgr-cfg.yang YANG model to configure the Ethernet controller.

YANG model	Example

Cisco-IOS-XR-ifmgr-cfg.yang	<?xml version="1.0"?>
	<rpc message-id="102" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
	  <edit-config>
	        <target>
	        <candidate/>
	        </target>
	  <config xmlns:xc="urn:ietf:params:xml:ns:netconf:base:1.0">
	   <interface-configurations
	      xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-ifmgr-cfg">
	
	      <interface-configuration>
	      <active>act</active>
	          <interface-name>TenGigECtrlr0/0/0/0/1</interface-name>
	        <shutdown xc:operation="delete" />
	        </interface-configuration>
	      </interface-configurations>
	    </config>
	  </edit-config>
	</rpc>

Step 2 Use the Cisco-IOS-XR-ifmgr-cfg.yang YANG model to configure the Coherent DSP controller.

YANG model	Example

Cisco-IOS-XR-ifmgr-cfg.yang	<?xml version="1.0"?>
	<rpc message-id="102" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
	  <edit-config>
	        <target>
	        <candidate/>
	        </target>
	  <config xmlns:xc="urn:ietf:params:xml:ns:netconf:base:1.0">
	   <interface-configurations
	      xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-ifmgr-cfg">
	      <interface-configuration>
	      <active>act</active>
	    <interface-name>CoherentDSP0/0/0/6</interface-name>
	    <shutdown xc:operation="delete" />
	   </interface-configuration>
	
	         <interface-configuration>
	      <active>act</active>
	    <interface-name>CoherentDSP0/0/0/13</interface-name>
	    <shutdown></shutdown>
	   </interface-configuration>

YANG model	Example

```
         <interface-configuration>
      <active>act</active>
    <interface-name>CoherentDSP0/0/0/20</interface-name>
    <shutdown></shutdown>
   </interface-configuration>

         <interface-configuration>
      <active>act</active>
    <interface-name>CoherentDSP0/0/0/27</interface-name>
    <shutdown></shutdown>
   </interface-configuration>
      </interface-configurations>
    </config>
  </edit-config>
</rpc>
```

Step 3 Use the Cisco-IOS-XR-pfi-im-cmd-ctrlr-oper.yang YANG model to display the name, status, and port description of the Ethernet controller.

YANG model	Example
Cisco-IOS-XR-pfi-im-cmd-ctrlr-oper.yang	

```
<?xml version="1.0" ?>
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<get>
<filter>
<controllers
xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-pfi-im-cmd-ctrlr-oper">

     <controllers>
        <controller>
           <interafce-name>HundredGigECtrlr0/0/0/8
           </interafce-name>
        </controller>
     </controllers>
</controllers>
</filter>
</get>
</rpc>
```

Step 4 Use the Cisco-IOS-XR-pfi-im-cmd-ctrlr-oper.yang YANG model to display the name, status, and port description of the Coherent DSP controller.

YANG model	Example
Cisco-IOS-XR-pfi-im-cmd-ctrlr-oper.yang	

```
<?xml version="1.0" ?>
Query:
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<get>
<filter>
<controllers
xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-pfi-im-cmd-ctrlr-oper">

     <controllers>
        <controller>
           <interafce-name>CoherentDSP0/0/0/19
           </interafce-name>
        </controller>
     </controllers>
</controllers>
```

YANG model	Example
	</filter> </get> </rpc>

Configure Performance Monitoring

Step 1 Use the Cisco-IOS-XR-ifmgr-cfg.yang and Cisco-IOS-XR-pmengine-cfg.yang YANG models for configuring the performance monitoring parameters for the Optics, Ethernet, and coherentDSP controllers.

Step 2 Use the Cisco-IOS-XR-pmengine-oper.yang YANG models to view the performance monitoring parameters for the Optics, Ethernet, and coherentDSP controllers.

The table below shows an example that displays all the PM parameters for the optics controller. You can use specific filters for the required the output.

YANG model	Example
Cisco-IOS-XR-pmengine-oper.yang	<?xml version="1.0" ?> <rpc message-id="856612" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"> <get> <filter type="subtree"> <performance-management xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-pmengine-oper"> <optics> <optics-ports> <optics-port>Optics0/0/0/1</optics-port> </optics-ports> </optics> </performance-management> </filter> </get> </rpc>

The table below shows an example that displays current 15 minute FEC PM for the Coherent DSP controller.

YANG model	Example
Cisco-IOS-XR-pmengine-oper.yang	<?xml version="1.0" ?> <rpc message-id="856612" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"> <get> <filter type="subtree"> <performance-management xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-pmengine-oper"> <otu> <otu-ports> <otu-port> <name>CoherentDSP0/0/0/12</name> <otu-current> <otu-minute15> <otu-minute15fecs/> </otu-minute15>

YANG model	Example
	</otu-current> </otu-port> </otu-ports> </otu> </performance-management> </filter> </get> </rpc>

Configure Loopback

Step 1 Use the Cisco-IOS-XR-ifmgr-cfg.yang and Cisco-IOS-XR-controller-otu-cfg YANG models for configuring Loopback.

YANG model	Example

Cisco-IOS-XR-ifmgr-cfg.yang
Cisco-IOS-XR-controller-otu-cfg.yang
	<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
	message-id="101">
	  <edit-config>
	    <target>
	      <candidate/>
	    </target>
	    <config>
	      <interface-configurations
	xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-ifmgr-cfg">
	        <interface-configuration>
	          <active>act</active>
	          <interface-name>CoherentDSP0/1/0/0</interface-name>
	          <otu
	xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-controller-otu-cfg">
	            <otn-send-tti>
	
	<string-type>send-tti-full-ascii/full-ascii</string-type>
	              <full-ascii-string>test1234</full-ascii-string>
	            </otn-send-tti>
	            <otn-expected-tti>
	
	<string-type>exp-tti-full-ascii/full-ascii</string-type>
	              <full-ascii-string>test1234</full-ascii-string>
	            </otn-expected-tti>
	          </otu>
	        </interface-configuration>
	      </interface-configurations>
	    </config>
	  </edit-config>
	</rpc>

Step 2 Use the Cisco-IOS-XR-ifmgr-cfg.yang and Cisco-IOS-XR-drivers-media-eth-cfg.yang YANG models for configuring the maintenance mode and loopback on an Ethernet controller.

YANG model	Example
Cisco-IOS-XR-ifmgr-cfg.yang Cisco-IOS-XR-drivers-media-eth-cfg.yang	<rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:34d98974-474a-4396-ad1a-6dd4ddfa20bc">

YANG model	Example
	<ok/> </rpc-reply>

Configure MACsec Encryption

Step 1 Use the Cisco-IOS-XR-ncs1k-mxp-cfg.yang YANG model to create an encrypted slice.

YANG model	Example
Cisco-IOS-XR-ncs1k-mxp-cfg.yang	<?xml version="1.0"?> <rpc message-id="102" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"> <edit-config> <target> <candidate/> </target> <config xmlns:xc="urn:ietf:params:xml:ns:netconf:base:1.0"> <hardware-module xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-ncs1k-mxp-cfg"> <node> <location>0_RP0_CPU0</location> <values> <value> <slice-id>1</slice-id> <client-rate>hundred-gig</client-rate> <trunk-rate>two-hundred-gig</trunk-rate> <fec>sd20</fec> <encrypted>true</encrypted> </value> </values> </node> </hardware-module> </config> </edit-config> </rpc>

Step 2 Use the Cisco-IOS-XR-lib-keychain-macsec-cfg.yang YANG model to configure the MACsec key chain.

YANG model	Example
Cisco-IOS-XR-lib-keychain-macsec-cfg.yang	<?xml version="1.0"?> <rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"> <edit-config> <target> <candidate/> </target> <config > <mac-sec-keychains xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-lib-keychain-macsec-cfg"> <mac-sec-keychain> <chain-name>keychain1</chain-name> <keies> <key> <key-id>kc1</key-id>

YANG model	Example
	<key-string> <string>055A575E701D1F58485446435A5D557B7A7579626473425647525007080902055F524947080906020304055A0A57560906554257550A5A575E701D1F5848544643</string> <cryptographic-algorithm>aes-256-cmac</cryptographic-algorithm> </key-string> <lifetime> <start-hour>10</start-hour> <start-minutes>10</start-minutes> <start-seconds>10</start-seconds> <start-date>1</start-date> <start-month>jan</start-month> <start-year>2016</start-year> <infinite-flag>true</infinite-flag> </lifetime> </key> <key> <key-id>kc2</key-id> <key-string> <string>0553515974181D5B485D40445E5857787A757A60617745504E5253050D0D0503565B4F400C0C0401030406580F53510F0F5C4450510F58545E701E1D5D4C53404A</string> <cryptographic-algorithm>aes-256-cmac</cryptographic-algorithm> </key-string> <lifetime> <start-hour>10</start-hour> <start-minutes>10</start-minutes> <start-seconds>10</start-seconds> <start-date>13</start-date> <start-month>sep</start-month> <start-year>2016</start-year> <life-time>86400</life-time> </lifetime> </key> <key> <key-id>kc3</key-id> <key-string> <string>00554155500E5D5157701E1D5D4C53404A5A5E577E7E727F6B647040534355560E010F05015A504A47010F01060606065A0351510D035741575C0C5D535B721E1F</string> <cryptographic-algorithm>aes-256-cmac</cryptographic-algorithm> </key-string> <lifetime> <start-hour>10</start-hour> <start-minutes>10</start-minutes> <start-seconds>10</start-seconds> <start-date>25</start-date> <start-month>dec</start-month> <start-year>2016</start-year> <end-hour>10</end-hour> <end-minutes>10</end-minutes> <end-seconds>10</end-seconds> <end-date>1</end-date> <end-month>jan</end-month> <end-year>2017</end-year> </lifetime> </key> </keies> </mac-sec-keychain> </mac-sec-keychains> </config>

YANG model	Example
	</edit-config> </rpc>

Step 3 Use the Cisco-IOS-XR-crypto-macsec-mka-cfg.yang YANG model to configure a MACsec policy.

YANG model	Example
Cisco-IOS-XR-crypto-macsec-mka-cfg.yang	<?xml version="1.0"?> <rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"> <edit-config> <target> <candidate/> </target> <config > <macsec xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-crypto-macsec-mka-cfg"> <policy> <name>mac_policy</name> <key-server-priority>255</key-server-priority> <conf-offset>conf-off-set-0</conf-offset> <security-policy>must-secure</security-policy> <window-size>100</window-size> <cipher-suite>gcm-aes-xpn-256</cipher-suite> </policy> </macsec> </config> </edit-config> </rpc>

Step 4 Use the Cisco-IOS-XR-ifmgr-cfg.yang and Cisco-IOS-XR-crypto-macsec-mka-if-cfg.yang YANG model to configure MACsec on a MACsec controller.

YANG model	Example
Cisco-IOS-XR-ifmgr-cfg Cisco-IOS-XR-crypto-macsec-mka-if-cfg.yang	<?xml version="1.0"?> <rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1. <edit-config> <target> <candidate/> </target> <config > <interface-configurations xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-ifmgr-cfg"> <interface-configuration> <active>act</active> <interface-name>MACSecCtrlr0/0/0/10</interface-name> <macsec xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-crypto-macsec-mka-if-cf <psk-key-chain> <key-chain-name>kc</key-chain-name> <policy-name>mac_policy</policy-name> </psk-key-chain> </macsec> </interface-configuration> </interface-configurations> </config> </edit-config> </rpc>

Step 5 Use the Cisco-IOS-XR-crypto-macsec-mka-oper.yang YANG model to verify the MACsec configuration and MKA session details of all the configured interfaces.

YANG model	Example
Cisco-IOS-XR-crypto-macsec-mka-oper.yang	<?xml version="1.0"?> rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"> <get> <filter> <macsec xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-crypto-macsec-mka-oper" > <mka> </mka> </macsec> </filter> </get> </rpc>

Step 6 Use the Cisco-IOS-XR-crypto-macsec-secy-oper.yang YANG model to verify the MACsec SecY statistics for all the MACsec Key Agreement protocol (MKA) sessions.

YANG model	Example
Cisco-IOS-XR-crypto-macsec-secy-oper.yang	<?xml version="1.0"?> <rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"> <get> <filter> <macsec xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-crypto-macsec-secy-oper" > <secy> </secy> </macsec> </filter> </get> </rpc>

Configure Breakout Patch Panel

Step 1 Use the Cisco-IOS-XR-patch-panel-cfg.yang YANG model to configure the breakout patch panel.

YANG model	Example

Cisco-IOS-XR-patch-panel-cfg.yang	<?xml version="1.0"?>
	<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="101">
	
	  <edit-config>
	    <target>
	      <candidate/>
	    </target>
	    <config type="subtree">
	     <patch-panel
	xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-patch-panel-cfg">

YANG model	Example
	<ipv4>169.254.1.4</ipv4> <user-name>SysAdmin</user-name> <password>!Password1</password> </patch-panel> </config> </edit-config> </rpc>

Step 2 Use the Cisco-IOS-XR-patch-panel-cfg.yang YANG model to delete the breakout patch panel.

YANG model	Example
Cisco-IOS-XR-patch-panel-cfg.yang	<?xml version="1.0"?> <rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="101"> <edit-config> <target> <candidate/> </target> <config> <patch-panel xmlns:ns0="urn:ietf:params:xml:ns:netconf:base:1.0" xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-patch-panel-cfg" ns0:operation="delete"/> </config> </edit-config> </rpc>

Configure LLDP Drop

Step 1 Use the Cisco-IOS-XR-ncs1k-mxp-cfg.yang YANG model to configure LLDP drop.

YANG model	Example
Cisco-IOS-XR-ncs1k-mxp-cfg.yang	<?xml version="1.0"?> <rpc message-id="102" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"> <edit-config> <target> <candidate/> </target> <config xmlns:xc="urn:ietf:params:xml:ns:netconf:base:1.0"> <hardware-module xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-ncs1k-mxp-cfg"> <node> <location>0_RP0_CPU0</location> <slice> <slice-id>0</slice-id> <lldp>true</lldp> </slice> </node> </hardware-module> </config> </edit-config> </rpc>

Step 2 Use the Cisco-IOS-XR-ncs1k-mxp-cfg.yang YANG model to delete LLDP drop configuration.

YANG model	Example
Cisco-IOS-XR-ncs1k-mxp-cfg.yang	<?xml version="1.0"?> <rpc message-id="102" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"> <edit-config> <target> <candidate/> </target> <config xmlns:xc="urn:ietf:params:xml:ns:netconf:base:1.0"> <hardware-module xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-ncs1k-mxp-cfg"> <node> <location>0_RP0_CPU0</location> <slice> <slice-id>0</slice-id> <lldp>false</lldp> </slice> </node> </hardware-module> </config> </edit-config> </rpc>

Step 3 Use the Cisco-IOS-XR-ncs1k-mxp-cfg.yang YANG model to retrieve operational data for LLDP drop.

YANG model	Example
Cisco-IOS-XR-ncs1k-mxp-cfg.yang	<?xml version="1.0"?> <rpc message-id="856615" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"> <get> <filter> <lldp-snoop-data xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-ncs1k-mxp-lldp-oper"/> </filter> </get> </rpc>

Configure PRBS on Trunk Ports

Step 1 Use Cisco-IOS-XR-ifmgr-cfg and Cisco-IOS-XR-controller-otu-cfg yang models to configure PRBS feature on trunk ports.

YANG model	Example
Cisco-IOS-XR-ifmgr-cfg.yang Cisco-IOS-XR-controller-otu-cfg.yang	<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"> <edit-config> <target><candidate/></target> <config xmlns:xc="urn:ietf:params:xml:ns:netconf:base:1.0"> <interface-configurations xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-ifmgr- xmlns:controller-otu-cfg="http://cisco.com/ns/yang/Cisco-IOS-XR-controller-ot <interface-configuration> <active>act</active> <interface-name>CoherentDSP0/0/0/27</interface-name>

YANG model	Example

	`<controller-otu-cfg:otu>`
	`<controller-otu-cfg:prbs>`
	`<controller-otu-cfg:mode-value>mode-source-sink</controller-otu-cfg:mode-v`
	`<controller-otu-cfg:patternvalue>pattern-pn31</controller-otu-cfg:patternv`
	`</controller-otu-cfg:prbs>`
	`<controller-otu-cfg:secondary-admin-state>maintenance</controller-otu-cfg:s`
	`</controller-otu-cfg:otu>`
	`</interface-configuration>`
	`</interface-configurations>`
	`</config>`
	`</edit-config>`
	`</rpc>`

Step 2 Use Cisco-IOS-XR-ifmgr-cfg and Cisco-IOS-XR-controller-otu-cfg yang models to retrieve PRBS configuration on the trunk ports.

YANG model	Example

Cisco-IOS-XR-ifmgr-cfg.yang
Cisco-IOS-XR-controller-otu-cfg.yang
	`<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">`
	`<edit-config>`
	`<target><candidate/></target>`
	`<config xmlns:xc="urn:ietf:params:xml:ns:netconf:base:1.0">`
	`<interface-configurations xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-ifm`
	`xmlns:controller-otu-cfg="http://cisco.com/ns/yang/Cisco-IOS-XR-controller`
	`<interface-configuration>`
	`<active>act</active>`
	`<interface-name>CoherentDSP0/0/0/27</interface-name>`
	`<controller-otu-cfg:otu>`
	`<controller-otu-cfg:prbs>`
	`<controller-otu-cfg:mode-value>mode-source-sink</controller-otu-cfg:mode-v`
	`<controller-otu-cfg:patternvalue>pattern-pn31</controller-otu-cfg:patternv`
	`</controller-otu-cfg:prbs>`
	`<controller-otu-cfg:secondary-admin-state>maintenance</controller-otu-cfg:s`
	`</controller-otu-cfg:otu>`
	`</interface-configuration>`
	`</interface-configurations>`
	`</config>`
	`</edit-config>`
	`</rpc>`

Step 3 Use Cisco-IOS-XR-controller-otu-oper yang model to retrieve PRBS status on the trunk ports.

YANG model	Example

Cisco-IOS-XR-controller-otu-oper.yang	`<rpc message-id="101"`
	`xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">`
	`<get>`
	`<filter>`
	`<otu`
	`xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-controller-otu-oper">`
	`<controllers>`
	`<controller>`
	`<prbs/>`
	`</controller>`
	`</controllers>`

Monitor Headless Statistics

In the headless mode, the data path and statistics are maintained for at least 72 hours. The collected statistics are preserved for a slice until the hardware module configuration is removed or changed on that slice. These statistics are automatically cleared during the next reload or CPU-OIR operation.

Use the Cisco-IOS-XR-ncs1k-mxp-headless-oper YANG model for monitoring the headless statistics.

YANG model	Example

Cisco-IOS-XR-ncs1k-mxp-headless-oper	`<?xml version="1.0" ?>`
	`<rpc message-id="856615"`
	`xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">`
	`<get>`
	`<filter>`
	`<headless-func-data`
	`xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-ncs1k-mxp-headless-oper"`
	` />`
	`</filter>`
	`</get>`
	`</rpc>`

Open Configuration Model for Client FEC and Laser-Squelch

Table 1: Feature History

Feature Name	Release	Description
OC (Open Configuration) Model for Client FEC and Laser Squelch	Cisco IOS XR Release 7.3.1	The OC model for configuring client FEC and Laser Squelch is available. This feature enables you to perform the configuration using scripts, which is less time-consuming. Also, the Open Configuration model supports the use of vendor-neutral data models to configure and manage the network.

Step 1 You can enable FEC (Forward Error Correction) on clients using the following scripts:

```
"openconfig-platform:components": {
 "component": [
  {
   "name": "0/0-Optics0/0/0/2",
   "config": {
    "name": "0/0-Optics0/0/0/2"
   },
   "openconfig-platform-transceiver:transceiver": {
    "config": {
     "fec-mode": "openconfig-platform-types:FEC_ENABLED"
    }
   }
  }
```

Step 2 You can get operational data using GNMI.

```
"state": {
    "connector-type": "openconfig-transport-types:LC_CONNECTOR",
    "date-code": "2019-08-05T00:00:00Z+00:00",
    "fault-condition": false,
    "fec-mode": "openconfig-platform-types:FEC_ENABLED",
    "fec-uncorrectable-words": 0,
    "form-factor": "openconfig-transport-types:QSFP28",
    "otn-compliance-code": "openconfig-transport-types:OTN_UNDEFINED",
    "present": "PRESENT",
    "serial-no": "INL23321878",
    "sonet-sdh-compliance-code": "openconfig-transport-types:SONET_UNDEFINED",
    "vendor": "CISCO-INNOLIGHT",
    "vendor-part": "10-3220-02",
    "vendor-rev": "1C"
}
```

IPv4 PING Over NETCONF

Use the Cisco-IOS-XR-ping-act YANG model to do the ping test to the destination IPv4 addresses.The following example shows the RPC request and RPC response messages for a successful ping test. The destination host is reachable and the success rate is 100%.

YANG Model	Example
Cisco-IOS-XR-ping-act.yang	<nc:rpc xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:28170002-365f-45be-a8e1-e1f54d8b64b5"><ping xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-ping-act"> <destination> <destination>10.127.60.1</destination> </destination> </ping> </nc:rpc> <rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:28170002-365f-45be-a8e1-e1f54d8b64b5"> <ping-response xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-ping-act"> <ipv4>

YANG Model	Example
	<destination>10.127.60.1</destination> <data-size>100</data-size> <timeout>2</timeout> <pattern>abcd</pattern> <rotate-pattern>false</rotate-pattern> <replies> <reply> <reply-index>1</reply-index> <result>!</result> </reply> <reply> <reply-index>2</reply-index> <result>!</result> </reply> <reply> <reply-index>3</reply-index> <result>!</result> </reply> <reply> <reply-index>4</reply-index> <result>!</result> </reply> <reply> <reply-index>5</reply-index> <result>!</result> </reply> </replies> <hits>5</hits> <total>5</total> <success-rate>100</success-rate> <rtt-min>1</rtt-min> <rtt-avg>1</rtt-avg> <rtt-max>2</rtt-max>

YANG Model	Example
	</ipv4> </ping-response> </rpc-reply>

The following example shows the RPC request and RPC response messages for a failure ping test. The destination host is not reachable and the success rate is 0%.

YANG model	Example
Cisco-IOS-XR-ping-act.yang	<nc:rpc xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:28170002-365f-45be-a8e1-e1f54d8b64b5"><ping xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-ping-act"> <destination> <destination>10.127.60.1</destination> </destination> </ping> </nc:rpc> <rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:02800209-6ebf-4955-8588-f6cdfd6f2750"> <ping-response xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-ping-act"> <ipv4> <destination>10.127.60.171</destination> <data-size>100</data-size> <timeout>2</timeout> <pattern>abcd</pattern> <rotate-pattern>false</rotate-pattern> <replies> <reply> <reply-index>1</reply-index> <result>.</result> </reply> <reply> <reply-index>2</reply-index> <result>.</result> </reply> <reply>

YANG model	Example
	<reply-index>3</reply-index> <result>.</result> </reply> <reply> <reply-index>4</reply-index> <result>.</result> </reply> <reply> <reply-index>5</reply-index> <result>.</result> </reply> </replies> <hits>0</hits> <total>5</total> <success-rate>0</success-rate> </ipv4> </ping-response> </rpc-reply>

IPv6 PING Over NETCONF

Table 2: Feature History

Feature Name	Release	Description
NETCONF Support for READ, WRITE, and Execute or Administrative Commands.	Cisco IOS XR Release 7.3.1	Support for IPv4 and IPv6 Ping test using the Cisco-IOS-XR-ping-act YANG model, instead of using CLI commands, is available. RPC (Remote Procedure Call) Request and Response messages are used to do the ping test, which is automated using scripts. This enables you to perform the ping test in a less time-consuming manner and to enhance network scalability.

Use the Cisco-IOS-XR-ping-act YANG model to do the ping test to the destination IPv6 addresses. The following example shows the RPC request and RPC response messages for a successful ping test. The destination host is reachable and the success rate is 100%.

YANG model	Example
Cisco-IOS-XR-ping-act.yang	<nc:rpc xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:28170002-365f-45be-a8e1-e1f54d8b64b5"><ping xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-ping-act"> <destination> <destination>2001:420:5446:2014::281:178</destination> </destination> </ping> </nc:rpc> <rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:15798adc-f9f9-41b2-9aa5-a1c88dd788e8"> <ping-response xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-ping-act"> <ipv6> <destination>2001:420:5446:2014::281:178</destination> <repeat-count>50</repeat-count> <data-size>100</data-size> <timeout>2</timeout> <pattern>abcd</pattern> <rotate-pattern>false</rotate-pattern> <replies> <reply> <reply-index>1</reply-index> <result>!</result> </reply> <reply> <reply-index>2</reply-index> <result>!</result> </reply> <reply> <reply-index>3</reply-index> <result>!</result>

YANG model	Example
	</reply> <reply> <reply-index>4</reply-index> <result>!</result> </reply> <reply> <reply-index>5</reply-index> <result>!</result> </reply> </replies> <hits>5</hits> <total>5</total> <success-rate>100</success-rate> <rtt-min>1</rtt-min> <rtt-avg>1</rtt-avg> <rtt-max>2</rtt-max> </ipv6> </ping-response> </rpc-reply>

The following example shows the RPC request and RPC response messages for a failure ping test. The destination host is not reachable and the success rate is 0%.

YANG model	Example
Cisco-IOS-XR-ping-act.yang	<nc:rpc xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:28170002-365f-45be-a8e1-e1f54d8b64b5"><ping xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-ping-act"> <destination> <destination>2001:420:5446:2014::281:178</destination> </destination> </ping> </nc:rpc> <rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:02800209-6ebf-4955-8588-f6cdfd6f2750"> <ping-response xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-ping-act">

YANG model	Example
	<ipv6> <destination>2001:420:5446:2014::281:178</destination> <data-size>100</data-size> <timeout>2</timeout> <pattern>abcd</pattern> <replies> <reply> <reply-index>1</reply-index> <result>.</result> </reply> <reply> <reply-index>2</reply-index> <result>.</result> </reply> <reply> <reply-index>3</reply-index> <result>.</result> </reply> <reply> <reply-index>4</reply-index> <result>.</result> </reply> <reply> <reply-index>5</reply-index> <result>.</result> </reply> </replies> <hits>0</hits> <total>5</total> <success-rate>0</success-rate> </ipv6> </ping-response>

YANG model	Example
	</rpc-reply>

Examples Using gRPC

Example—Verify the Slice Configuration Using gRPC

Set-up:

- Client—client_v3
- Client IP address and configured grpc port—192.0.2.198:57500

```
./client_v3 -server 192.0.2.198:57500 -oper show-cmd-text -cli_input_file show-hw-module
```

The slice configuration is displayed.

```
{
  "Response": "{\"ResReqId\":753690684504425618,\"output\":\"\\n--------------------------
 show hw-module slice all ---------------------------\\nSlice ID:               1\\nStatus:
                  Provisioned\\nClient Bitrate:          100\\nTrunk Bitrate:
100\\nDP FPGA Version:         H201 (NEED UPG)\\n\\nClient Port -   Trunk Port\\t
CoherentDSP0/0/0/12\\t CoherentDSP0/0/0/13\\nTraffic Split
Percentage\\n\\nHundredGigECtrlr0/0/0/7  \\t                 100
0\\nHundredGigECtrlr0/0/0/11 \\t                   0                 100\\n\\n\\n\"}",
  "FatalErrors": ""
}
```

Example—View the Optics Controller Configuration Using gRPC and Yang

Set-up:

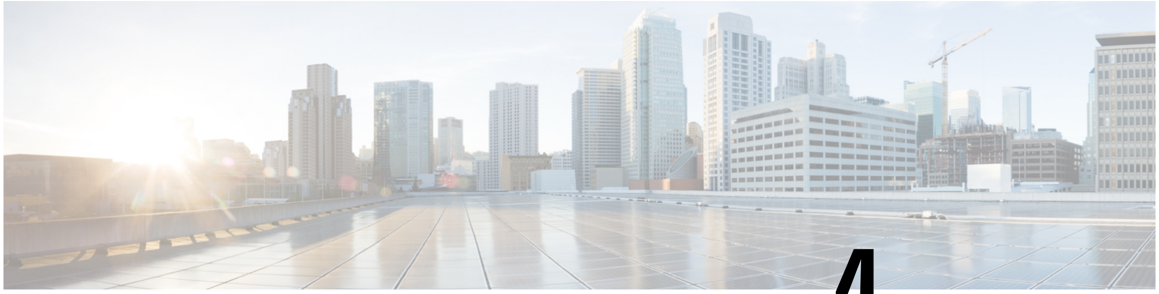
- Client—client_v3
- Client IP address and configured grpc port—192.0.2.198:57500
- Yang model—Cisco-IOS-XR-ifmgr-cfg

```
./client -server_addr=192.0.2.198:57500 -username=root -password=lab -oper=get-config
-yang_path='{"Cisco-IOS-XR-ifmgr-cfg:interface-configurations": [null]}'
```

The optics controller configuration is displayed.

```
{
 "Cisco-IOS-XR-ifmgr-cfg:interface-configurations": {
  "interface-configuration": [
   {
    "active": "act",
    "interface-name": "Optics0/0/0/5",
    "shutdown": [null]
   },
   {
    "active": "act",
    "interface-name": "Optics0/0/0/6",
 "Cisco-IOS-XR-controller-optics-cfg:optics": {
  "optics-dwdm-carrier": {
   "grid-type": "100mhz-grid",
   "param-type": "frequency",
   "param-value": 1927000
  }
 },
 "secondary-admin-state": "maintenance"
},
{
 "active": "act",
 "interface-name": "Optics0/0/0/12",
 "shutdown": [
  null
 ]
},
{
 "active": "act",
 "interface-name": "Optics0/0/0/13",
 "Cisco-IOS-XR-controller-optics-cfg:optics": {
  "optics-dwdm-carrier": {
   "grid-type": "100mhz-grid",
   "param-type": "frequency",
   "param-value": 1927000
  }
 },
 "secondary-admin-state": "maintenance"
},
{
 "active": "act",
 "interface-name": "Optics0/0/0/14",
 "Cisco-IOS-XR-controller-optics-cfg:optics": {
  "rx-thresholds": {
   "rx-threshold": [
    {
     "rx-threshold-type": "low",
     "rx-threshold": -120
    },
    {
     "rx-threshold-type": "high",
     "rx-threshold": 49
    }]}}}
 ,
{
 "active": "act",
 "interface-name": "Optics0/0/0/18",
 "Cisco-IOS-XR-controller-optics-cfg:optics": {
  "rx-thresholds": {
   "rx-threshold": [
    {
     "rx-threshold-type": "low",
     "rx-threshold": -120
    },
    {
     "rx-threshold-type": "high",
     "rx-threshold": 49
    }]}}}
   ,
{
 "active": "act",
 "interface-name": "Optics0/0/0/19",
 "shutdown": [
  null
 ],
 "Cisco-IOS-XR-controller-optics-cfg:optics": {
  "optics-dwdm-carrier": {
   "grid-type": "50g-hz-grid",
   "param-type": "frequency",
   "param-value": 19270
  }}}
,
{
 "active": "act",
 "interface-name": "Optics0/0/0/20",
 "Cisco-IOS-XR-controller-optics-cfg:optics": {
  "optics-dwdm-carrier": {
   "grid-type": "50g-hz-grid",
   "param-type": "frequency",
   "param-value": 19270
  },
  "rx-thresholds": {
   "rx-threshold": [
    {
     "rx-threshold-type": "low",
     "rx-threshold": -120
    },
    {
     "rx-threshold-type": "high",
     "rx-threshold": 49
    }]}}}
   ],
{
 "active": "act",
 "interface-name": "Optics0/0/0/26",
 "shutdown": [
  null
 ]
},
{
 "active": "act",
 "interface-name": "Optics0/0/0/27",
 "shutdown": [
  null
 ]
},
{
 "active": "act",
 "interface-name": "MgmtEth0/RP0/CPU0/0",
 "Cisco-IOS-XR-ipv4-io-cfg:ipv4-network": {
  "addresses": {
   "primary": {
    "address": "10.77.132.165",
    "netmask": "255.255.255.0"
   }}}}
  ,
{
 "active": "act",
 "interface-name": "TenGigECtrlr0/0/0/0/1",
 "Cisco-IOS-XR-pmengine-cfg:performance-management": {
  "ethernet-minute15": {
   "minute15-ether": {
    "minute15-ether-reports": {
     "minute15-ether-report": [
      {
       "ether-report": "report-fcs-err"
      }
     ]
    },
    "minute15-ether-thresholds": {
     "minute15-ether-threshold": [
      {
       "ether-threshold": "thresh-fcs-err",
       "ether-threshold-value": 1000
      }
     ]
    }
   }
  }
 }
},
{
 "active": "act",
 "interface-name": "TenGigECtrlr0/0/0/0/2",
 "Cisco-IOS-XR-pmengine-cfg:performance-management": {
  "ethernet-minute15": {
   "minute15-ether": {
    "minute15-ether-reports": {
     "minute15-ether-report": [
      {
       "ether-report": "report-fcs-err"
      }
     ]
    },
    "minute15-ether-thresholds": {
     "minute15-ether-threshold": [
      {
       "ether-threshold": "thresh-fcs-err",
       "ether-threshold-value": 1000
      }
     ]
    }
   }
  }
 }
},
{
 "active": "act",
 "interface-name": "TenGigECtrlr0/0/0/0/3",
 "Cisco-IOS-XR-pmengine-cfg:performance-management": {
  "ethernet-minute15": {
   "minute15-ether": {
    "minute15-ether-reports": {
     "minute15-ether-report": [
      {
       "ether-report": "report-fcs-err"
      }
     ]
    },
    "minute15-ether-thresholds": {
     "minute15-ether-threshold": [
      {
       "ether-threshold": "thresh-fcs-err",
       "ether-threshold-value": 1000
      }
     ]
    }
   }
  }
 }
},
{
 "active": "act",
 "interface-name": "TenGigECtrlr0/0/0/0/4",
 "Cisco-IOS-XR-pmengine-cfg:performance-management": {
  "ethernet-minute15": {
   "minute15-ether": {
    "minute15-ether-reports": {
     "minute15-ether-report": [
      {
       "ether-report": "report-fcs-err"
      }
     ]
    },
    "minute15-ether-thresholds": {
     "minute15-ether-threshold": [
      {
       "ether-threshold": "thresh-fcs-err",
       "ether-threshold-value": 1000
      }
     ]
    }
   }
  }
 }
},
{
 "active": "act",
 "interface-name": "TenGigECtrlr0/0/0/11/1",
 "Cisco-IOS-XR-pmengine-cfg:performance-management": {
  "ethernet-minute15": {
   "minute15-ether": {
    "minute15-ether-reports": {
     "minute15-ether-report": [
      {
       "ether-report": "report-fcs-err"
      }
     ]
    },
    "minute15-ether-thresholds": {
     "minute15-ether-threshold": [
      {
       "ether-threshold": "thresh-fcs-err",
       "ether-threshold-value": 1000
      }
     ]
    }
   }
  }
 }
},
{
 "active": "act",
 "interface-name": "TenGigECtrlr0/0/0/11/2",
 "Cisco-IOS-XR-pmengine-cfg:performance-management": {
  "ethernet-minute15": {
   "minute15-ether": {
    "minute15-ether-reports": {
     "minute15-ether-report": [
      {
       "ether-report": "report-fcs-err"
      }
     ]
    },
    "minute15-ether-thresholds": {
     "minute15-ether-threshold": [
      {
          "ether-threshold": "thresh-fcs-err",
          "ether-threshold-value": 1000
         }
        ]
       }
      }
     }
    }
   },
   {
    "active": "act",
    "interface-name": "TenGigECtrlr0/0/0/11/3",
    "Cisco-IOS-XR-pmengine-cfg:performance-management": {
     "ethernet-minute15": {
      "minute15-ether": {
       "minute15-ether-reports": {
        "minute15-ether-report": [
         {
          "ether-report": "report-fcs-err"
         }
        ]
       },
       "minute15-ether-thresholds": {
        "minute15-ether-threshold": [
         {
          "ether-threshold": "thresh-fcs-err",
          "ether-threshold-value": 1000
         }
        ]
       }
      }
     }
    }
   },
   {
    "active": "act",
    "interface-name": "TenGigECtrlr0/0/0/11/4",
    "Cisco-IOS-XR-pmengine-cfg:performance-management": {
     "ethernet-minute15": {
      "minute15-ether": {
       "minute15-ether-reports": {
        "minute15-ether-report": [
         {
          "ether-report": "report-fcs-err"
         }
        ]
       },
       "minute15-ether-thresholds": {
        "minute15-ether-threshold": [
         {
          "ether-threshold": "thresh-fcs-err",
          "ether-threshold-value": 1000
         }
        ]
       }
      }
     }
    }
   }
  ]
 }
}
emsGetConfig: ReqId 1, byteRecv: 7455
```

```
----------------- gRPC Summary ----------------------

Operation: get-config
Number of iterations: 1
Total bytes transferred: 7455
Number of bytes per second: 124482
Ave elapsed time in seconds: 0.059888
Min elapsed time in seconds: 0.059888
Max elapsed time in seconds: 0.059888

--------------- End gRPC Summary ---------------------
```



CHAPTER 4

Terminal-device Model

The Terminal-device model is a industry-wide standard YANG model from standard bodies, such as IETF and IEEE. The terminal-device model provides a unique way to provision the Cisco NCS 1002 using YANG models that are defined for configuration data and operational data.

The gRPC (google-defined Remote Procedure Calls) and NETCONF (Network Configuration Protocol) communication protocols are used to establish connection between the client and Cisco NCS1002.

Structure of YANG Models

YANG data models can be represented in a hierarchical, tree-based structure with nodes. This representation makes the models easy to understand.

There are two Terminal-device models for Cisco NCS 1004, such as:

- OpenConfig Terminal model
- OpenConfig Platform model

For more details on supported versions, see Supported YANG Models in NCS 1004, on page 4.

The following is the tree structure of the OpenConfig Terminal model:

```
module: openconfig-terminal-device
   +--rw terminal-device
      +--ro state
      +--rw logical-channels
      |  +--rw channel* [index]
      |     +--rw index
      |     +--rw config
      |     |  +--rw index?                 uint32
      |     |  +--rw description?           string
      |     |  +--rw admin-state?           oc-opt-types:admin-state-type
      |     |  +--rw rate-class?            identityref
      |     |  +--rw trib-protocol?         identityref
      |     |  +--rw logical-channel-type?   identityref
      |     |  +--rw loopback-mode?         oc-opt-types:loopback-mode-type
      |     +--ro state
      |     |  +--ro index?                   uint32
      |     |  +--ro description?             string
      |     |  +--ro admin-state?             oc-opt-types:admin-state-type
      |     |  +--ro rate-class?              identityref
      |     |  +--ro trib-protocol?           identityref
      |     |  +--ro logical-channel-type?    identityref
      |     |  +--ro loopback-mode?           oc-opt-types:loopback-mode-type
      |     |  +--ro link-state?              enumeration
      |     +--rw otn
      |     |  +--rw config
      |     |  |  +--rw tti-msg-transmit?    string
      |     |  |  +--rw tti-msg-expected?    string
      |     |  |  +--rw tti-msg-auto?        boolean
      |     |  +--ro state
      |     |     +--ro tti-msg-transmit?            string
      |     |     +--ro tti-msg-expected?            string
      |     |     +--ro tti-msg-recv?                string
      |     |     +--ro errored-seconds?             yang:counter64
      |     |     +--ro severely-errored-seconds?    yang:counter64
      |     |     +--ro unavailable-seconds?         yang:counter64
      |     |     +--ro fec-corrected-bits?          yang:counter64
      |     |     +--ro background-block-errors?     yang:counter64
                  +--ro fec-uncorrectable-words
      |     |     +--ro pre-fec-ber
      |     |     |  +--ro instant?    decimal64
      |     |     |  +--ro avg?        decimal64
      |     |     |  +--ro min?        decimal64
      |     |     |  +--ro max?        decimal64
      |     |     +--ro post-fec-ber
      |     |     |  +--ro instant?    decimal64
      |     |     |  +--ro avg?        decimal64
      |     |     |  +--ro min?        decimal64
      |     |     |  +--ro max?        decimal64
      |     +--rw ethernet
      |     |  +--rw config
      |     |  +--ro state
      |     |     +--ro in-mac-pause-frames?       yang:counter64
      |     |     +--ro in-oversize-frames?        yang:counter64
      |     |     +--ro in-jabber-frames?          yang:counter64
      |     |     +--ro in-fragment-frames?        yang:counter64
      |     |     +--ro in-crc-errors?             yang:counter64
      |     |     +--ro out-mac-pause-frames?      yang:counter64
      |     +--rw ingress
      |     |  +--rw config
      |     |  |  +--rw transceiver?         -> /oc-platform:components/component/name
      |     |  |  +--rw physical-channel*    ->
/oc-platform:components/component/oc-transceiver:transceiver/physical-channels/channel/index

      |     |  +--ro state
      |     |     +--ro transceiver?         -> /oc-platform:components/component/name
      |     |     +--ro physical-channel*    ->
/oc-platform:components/component/oc-transceiver:transceiver/physical-channels/channel/index

      |     +--rw logical-channel-assignments
      |        +--rw assignment* [index]
      |           +--rw index     -> ../config/index
      |           +--rw config
      |           |  +--rw index?              uint32
      |           |  +--rw description?        string
      |           |  +--rw assignment-type?    enumeration
      |           |  +--rw logical-channel?    ->
/terminal-device/logical-channels/channel/index
      |           |  +--rw optical-channel?    -> /oc-platform:components/component/name
      |           |  +--rw allocation?        decimal64
      |           +--ro state
      |              +--ro index?             uint32
      |              +--ro description?       string
      |              +--ro assignment-type?   enumeration
      |              +--ro logical-channel?   ->
/terminal-device/logical-channels/channel/index
      |              +--ro optical-channel?   -> /oc-platform:components/component/name
      |              +--ro allocation?        decimal64
      +--rw operational-modes
         +--ro mode* [mode-id]
            +--ro mode-id    -> ../state/mode-id
            +--ro config
            +--ro state
               +--ro mode-id?       uint16
               +--ro description?   string
               +--ro vendor-id?     string
```

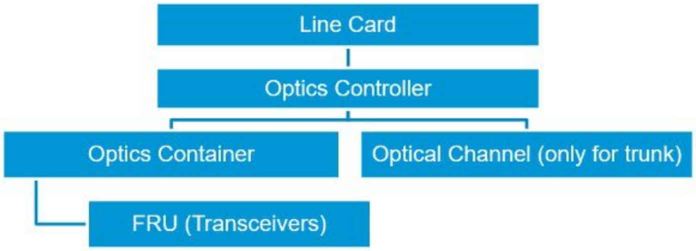
The following is the tree structure of the OpenConfig Platform model:

```
module: openconfig-platform
   +--rw components
      +--rw component* [name]
         +--rw name                           -> ../config/name
         +--rw config
         |  +--rw name?   string
         +--ro state
         |  +--ro name?          string
         |  +--ro type?          union
         |  +--ro id?            string
         |  +--ro description?   string
         |  +--ro mfg-name?      string
         |  +--ro version?       string
         |  +--ro serial-no?     string
         |  +--ro part-no?       string

         +--rw oc-transceiver:transceiver
         |  +--ro oc-transceiver:state
         |  |  +--ro oc-transceiver:form-factor?                 identityref
         |  |  +--ro oc-transceiver:present?                     enumeration
         |  |  +--ro oc-transceiver:connector-type?              identityref
         |  |  +--ro oc-transceiver:internal-temp?               int16
         |  |  +--ro oc-transceiver:vendor?                      string
         |  |  +--ro oc-transceiver:vendor-part?                 string
         |  |  +--ro oc-transceiver:vendor-rev?                  string
         |  |  +--ro oc-transceiver:ethernet-compliance-code?    identityref
         |  |  +--ro oc-transceiver:sonet-sdh-compliance-code?   identityref
         |  |  +--ro oc-transceiver:otn-compliance-code?         identityref
         |  |  +--ro oc-transceiver:serial-no?                   string
         |  |  +--ro oc-transceiver:date-code?                   yang:date-and-time
         |  |  +--ro oc-transceiver:fault-condition?             boolean
         |  +--rw oc-transceiver:physical-channels
         |     +--rw oc-transceiver:channel* [index]
         |        +--rw oc-transceiver:index     -> ../config/index
         |          +--ro oc-transceiver:output-frequency?      oc-opt-types:frequency-type

         |           +--ro oc-transceiver:output-power
         |           |  +--ro oc-transceiver:instant?   decimal64
         |           |  +--ro oc-transceiver:avg?       decimal64
         |           |  +--ro oc-transceiver:min?       decimal64
         |           |  +--ro oc-transceiver:max?       decimal64
         |           +--ro oc-transceiver:input-power
         |           |  +--ro oc-transceiver:instant?   decimal64
         |           |  +--ro oc-transceiver:avg?       decimal64
         |           |  +--ro oc-transceiver:min?       decimal64
         |           |  +--ro oc-transceiver:max?       decimal64
         |           +--ro oc-transceiver:laser-bias-current
         |              +--ro oc-transceiver:instant?   decimal64
         |              +--ro oc-transceiver:avg?       decimal64
         |              +--ro oc-transceiver:min?       decimal64
         |              +--ro oc-transceiver:max?       decimal64
         +--rw oc-opt-term:optical-channel
            +--rw oc-opt-term:config
            |  +--rw oc-opt-term:frequency?             oc-opt-types:frequency-type
            |  +--rw oc-opt-term:target-output-power?   decimal64
            |  +--rw oc-opt-term:operational-mode?      uint16
            |  +--rw oc-opt-term:line-port?             ->
/oc-platform:components/component/name
            +--ro oc-opt-term:state
               +--ro oc-opt-term:frequency?
oc-opt-types:frequency-type
               +--ro oc-opt-term:target-output-power?                         decimal64
               +--ro oc-opt-term:operational-mode?                            uint16
               +--ro oc-opt-term:line-port?                                   ->
/oc-platform:components/component/name
               +--ro oc-opt-term:chromatic-dispersion
               |  +--ro oc-opt-term:instant?   decimal64
               |  +--ro oc-opt-term:avg?       decimal64
               |  +--ro oc-opt-term:min?       decimal64
               |  +--ro oc-opt-term:max?       decimal64
               +--ro oc-opt-term:second-order-polarization-mode-dispersion
               |  +--ro oc-opt-term:instant?   decimal64
               |  +--ro oc-opt-term:avg?       decimal64
               |  +--ro oc-opt-term:min?       decimal64
               |  +--ro oc-opt-term:max?       decimal64
               +--ro oc-opt-term:polarization-dependent-loss
                  +--ro oc-opt-term:instant?   decimal64
                  +--ro oc-opt-term:avg?       decimal64
                  +--ro oc-opt-term:min?       decimal64
                  +--ro oc-opt-term:max?       decimal64
```

Inventory Details of Terminal-device Model

The hierarchy of Cisco NCS 1004 inventory is shown below:

Figure 3: Hierarchy of Cisco NCS 1004 Inventory



The inventory details and the naming convention of the components used in the Cisco NCS 1004 Terminal-device model are as follows:

Table 3: Inventory Details

Components	Naming Convention
Optics Controller	R/S-OpticsCtlrR/S/I/P
Optics Container	R/S-OpticsContainerR/S/I/P
Transceivers	R/S-OpticsR/S/I/P
Optical Channel Module	R/S-OpticalChannelR/S/I/P

The following table lists all the valid transceivers and optical channels that can be used for configuring Cisco NCS 1004 using Terminal-device model:

Table 4: Transceiver and Optical Channel Details

Components	Applicable Channels
Transceivers	0/0-Optics0/0/0/0 to 0/0-Optics0/0/0/13 0/1-Optics0/1/0/0 to 0/1-Optics0/1/0/13 0/2-Optics0/2/0/0 to 0/2-Optics0/2/0/13 0/3-Optics0/3/0/0 to 0/3-Optics0/3/0/13
Optical Channels	• 0/0-OpticalChannel0/0/0/0 • 0/0-OpticalChannel0/0/0/1 • 0/1-OpticalChannel0/1/0/0 • 0/1-OpticalChannel0/1/0/1 • 0/2-OpticalChannel0/2/0/0 • 0/2-OpticalChannel0/2/0/1 • 0/3-OpticalChannel0/3/0/0 • 0/3-OpticalChannel0/3/0/1

Note Only the optical channels of trunk ports must be mapped to the line ports. For more information about the port details, see Slice and Port Numbering.

Configuring Cisco NCS1004 Using Terminal-device Model

The following configurations are supported on the 1.2 Tbps line card. Client port operate at 100GE and OTU4 and map to trunk ports operating at 200G, 300G, 400G, 500G, or 600G.

You can configure the client port to OTU4 only in the muxponder configuration. LLDP drop, L1 encryption, and AINS are not supported on the OTU4 configuration.

The following table displays the client and trunk ports that are enabled for the muxponder configuration.

Trunk Data Rate	Client Data Rate (100GE/OTU4)	Trunk Ports	Client Ports
200	100GE/OTU4	0, 1	2,3, 4, 5
300	100GE/OTU4	0, 1	2, 3, 4, 5, 6, 7
400	100GE	0, 1	2, 3, 4, 5, 6, 7, 8, 9
500	100GE	0, 1	2, 3, 4, 5, 6, 7, 8, 9, 10, 11
600	100GE	0,1	2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13

The following table displays the client and trunk ports that are enabled for the muxponder slice 0 configuration.

Trunk Data Rate	Client Data Rate	Trunk Ports	Client Ports
200	100	0	2, 3
300	100	0	2, 3, 4
400	100	0	2, 3, 4, 5
500	100	0	2, 3, 4, 5, 6
600	100	0	2, 3, 4, 5, 6, 7

The following table displays the client and trunk ports that are enabled for the muxponder slice 1 configuration.

Trunk Data Rate	Client Data Rate	Trunk Ports	Client Ports
200	100	1	8, 9
300	100	1	8, 9, 10
400	100	1	8, 9, 10, 11
500	100	1	8, 9, 10, 11, 12
600	100	1	8, 9, 10, 11, 12, 13

All configurations can be accomplished using appropriate values for client bitrate and trunk bitrate parameters of the **hw-module** command.

The following table displays the trunk parameter ranges.

Trunk Payload	FEC	Min BPS	Max BPS	Min GBd	Max GBd
200G	27%	2	4.40625	31.51	69.43
300G	27%	2.8984375	6	34.7175497	71.8681352
400G	27%	3.8671875	6	46.2900663	71.8197392

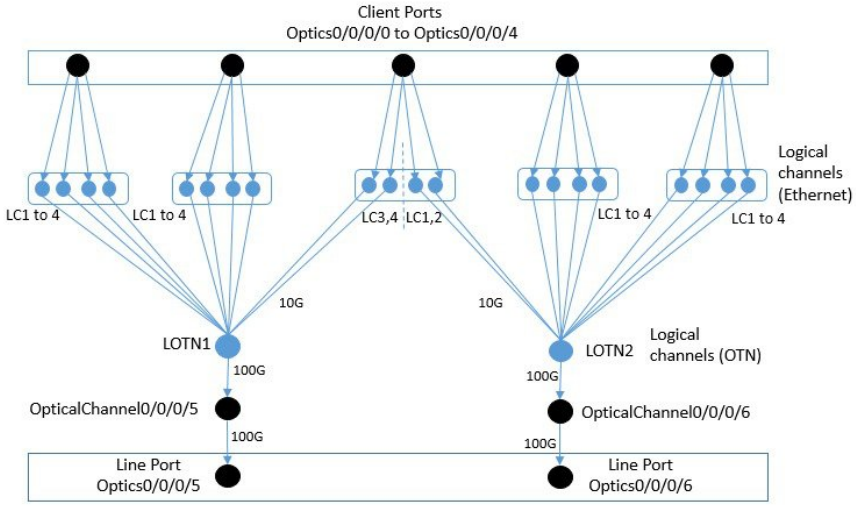
Trunk Payload	FEC	Min BPS	Max BPS	Min GBd	Max GBd
500G	27%	4.8281250	6	57.8625828	71.9068991
600G	15%	5.2578125	-	-	71.9552971

10Gx100G Configuration

For 10G to 100G configuration on slice 0, connect the client ports, Optics0/0/0/0 to Optics0/0/0/4 to the Logical OTN channels (LOTN1 and LOTN2). Each client port has four logical channels such as LC1, LC2, LC3, and LC4 and are identified by a unique identifier. These logical channels are the transceiver physical channels with which the logical channels are connected to the client ports. The LC1, LC2, LC3, and LC4 of client port Optics0/0/0/0 and client port Optics0/0/0/1 and the LC3, and LC4 of client port Optics0/0/0/2 are connected to the LOTN1. The LC1, LC2, LC3, and LC4 of client port Optics0/0/0/3 and client port Optics0/0/0/4 and the LC1, and LC2 of client port Optics0/0/0/2 are connected to the LOTN2. The LOTN1 is connected to the optical channel 5 and LOTN2 is connected to the Optical Channel0/0/0/6. Then the optical channel 5 is connected to the line port, Optics0/0/0/5 and the Optical Channel0/0/0/6 is connected to the line port, Optics0/0/0/6. The bandwidth allocation is 10G each for all the logical channels (LC) and the bandwidth allocation for LOTN1, LOTN2, Optical channel 5 and 6 is 100G.

The following figure explains the 10G to 100G configuration in slice 0.

Figure 4: 10Gx100G Configuration

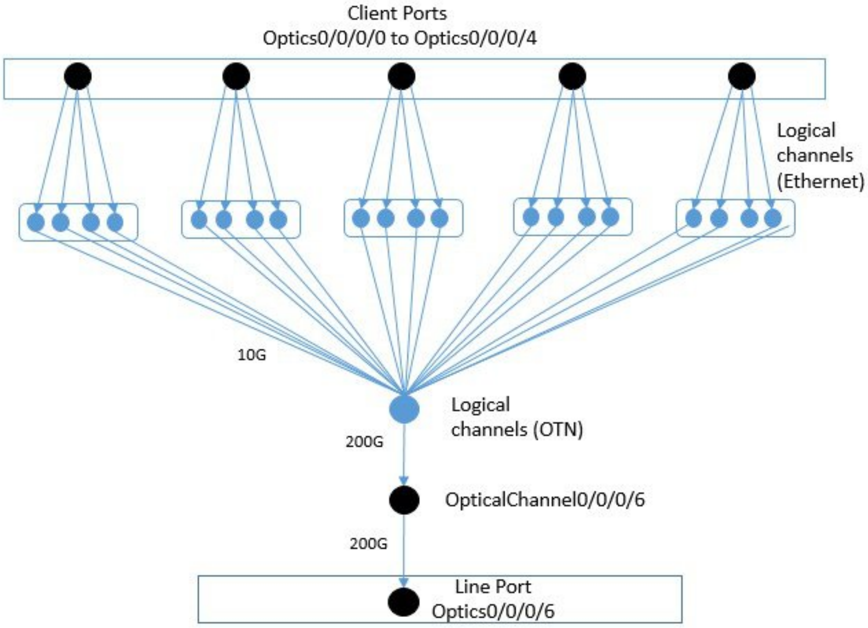


10Gx200G Configuration

For 10G to 200G configuration on slice 0, connect the 20 logical channels of client ports, Optics0/0/0/0 to Optics0/0/0/4 to the Logical OTN channel (LOTN). The Logical OTN channel is connected to the Optical Channel0/0/0/6. Then the Optical Channel0/0/0/6 is connected to the line port, Optics0/0/0/6. The bandwidth allocation is 10G each for all the logical channels and the bandwidth allocation for logical OTN channel, and Optical Channel0/0/0/6 is 200G.

The following figure explains 10G to 200G configuration in slice 0.

Figure 5: 10Gx200G Configuration

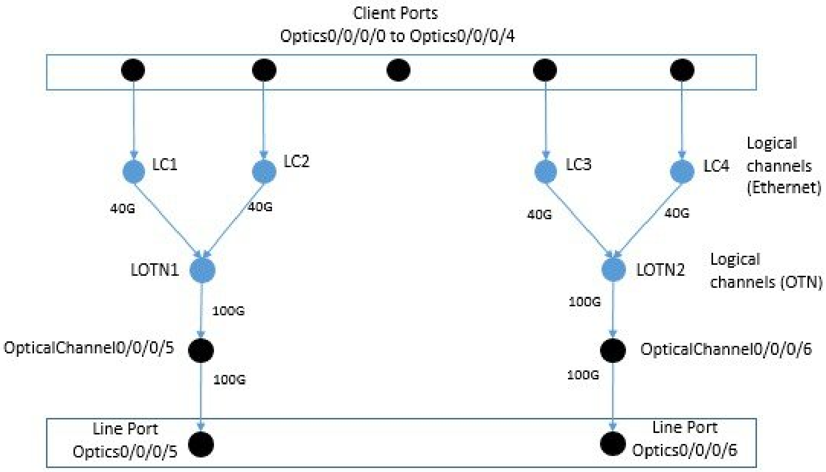


40Gx100G Configuration

For 40G to 100G configuration on slice 0, connect the client ports, Optics0/0/0/0, Optics0/0/0/1, Optics0/0/0/3, and Optics0/0/0/4 to the corresponding four Logical Channels, LC1, LC2, LC3 and LC4. The client port, Optics0/0/0/2 is not used. The LC1 and LC2 are connected to the first logical OTN channel (LOTN1) and the LC3 and LC4 are connected to the second logical OTN channel (LOTN2). The LOTN1 is connected to the optical channel 5 and LOTN2 is connected to the Optical Channel0/0/0/6. Then the optical channel 5 is connected to the line port, Optics0/0/0/5 and the Optical Channel0/0/0/6 is connected to the line port, Optics0/0/0/6. The bandwidth allocation is 40G each for all the logical channels (LC) and the bandwidth allocation for LOTN1, LOTN2, Optical channel 5 and 6 is 100G.

The following figure explains the 40G to 100G configuration in slice 0.

Figure 6: 40Gx100G Configuration

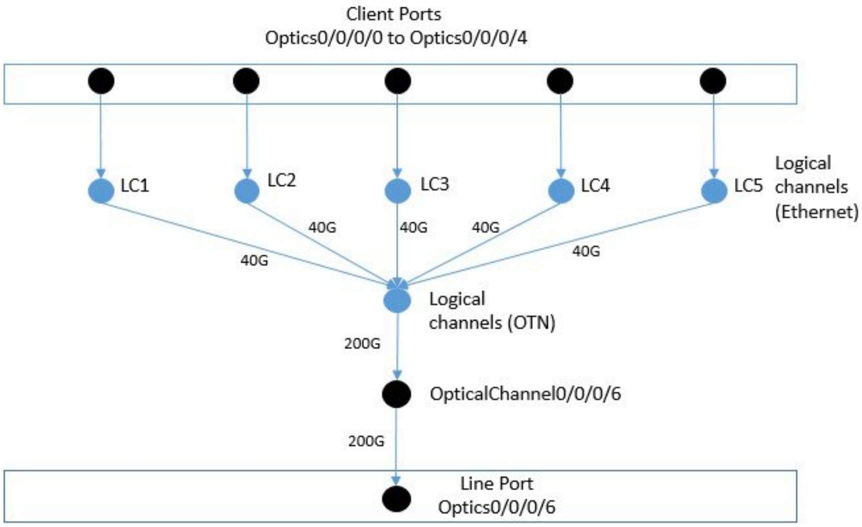


40Gx200G Configuration

For 40G to 200G configuration on slice 0, connect the Logical Channels, LC1, LC2, LC3, LC4 and LC5 of client ports, Optics0/0/0/0 to Optics0/0/0/4 to the Logical OTN channel (LOTN). The Logical OTN channel is connected to the Optical Channel0/0/0/6. Then the Optical Channel0/0/0/6 is connected to the line port, Optics0/0/0/6. The bandwidth allocation is 40G each for all the logical channels and the bandwidth allocation for logical OTN channel, and Optical Channel0/0/0/6 is 200G.

The following figure explains 40G to 200G configuration in slice 0.

Figure 7: 40Gx200G Configuration

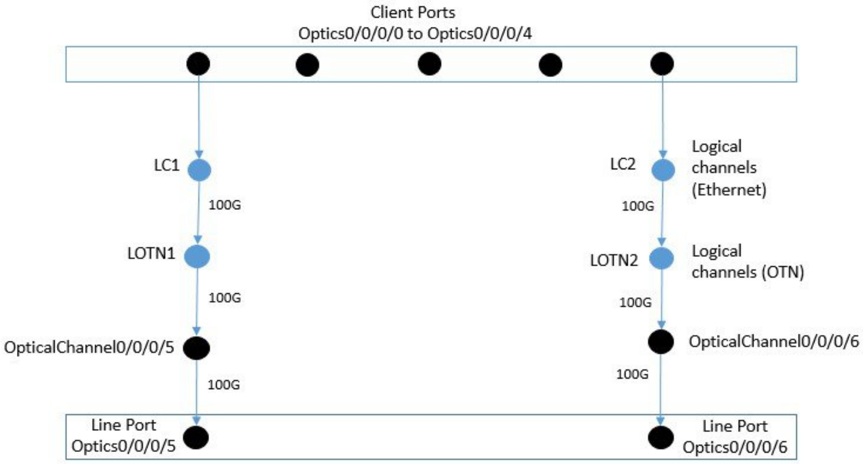


100Gx100G Configuration

For 100G to 100G configuration on slice 0, connect the logical channels LC1 and LC2 of the client ports, Optics0/0/0/0 and Optics0/0/0/4 to the Logical OTN channels, LOTN 1 and LOTN2. The client port Optics0/0/0/1, client port Optics0/0/0/2 and client port Optics0/0/0/3 are not used. The LOTN1 is connected to the optical channel 5 and LOTN2 is connected to the Optical Channel0/0/0/6. Then the optical channel 5 is connected to the line port Optics0/0/0/5 and the Optical Channel0/0/0/6 is connected to the line port Optics0/0/0/6. The bandwidth allocation is 100G each for the LC1, LC2, LOTN1, LOTN2, optical channel 5 and Optical Channel0/0/0/6.

The following figure explains 100G to 100G configuration in slice 0.

Figure 8: 100Gx100G Configuration

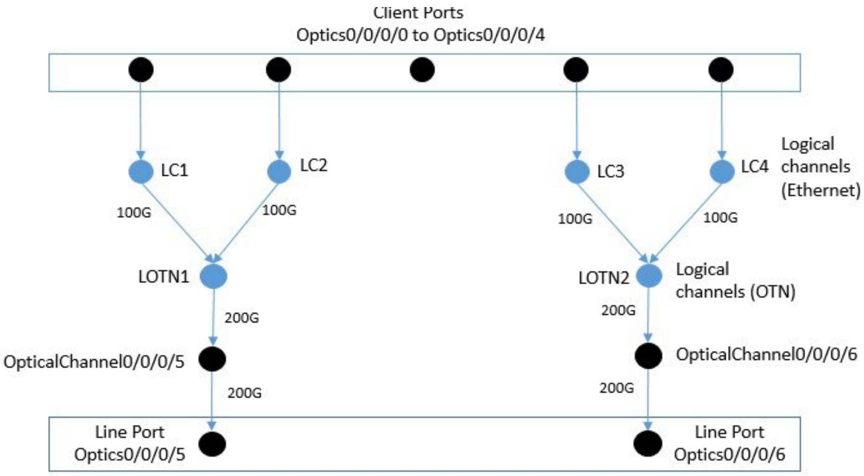


100Gx200G Configuration

For 100G to 200G configuration on slice 0, connect the client ports, Optics0/0/0/0, Optics0/0/0/1, Optics0/0/0/3, and Optics0/0/0/4 to the corresponding four Logical Channels, LC1, LC2, LC3 and LC4. The client port, Optics0/0/0/2 is not used. The LC1 and LC2 are connected to the first logical OTN channel (LOTN1) and the LC3 and LC4 are connected to the second logical OTN channel (LOTN2). The LOTN1 is connected to the optical channel 5 and LOTN2 is connected to the Optical Channel0/0/0/6. Then the optical channel 5 is connected to the line port, Optics0/0/0/5 and the Optical Channel0/0/0/6 is connected to the line port, Optics0/0/0/6. The bandwidth allocation is 100G each for all the logical channels and the bandwidth allocation for LOTN1, LOTN2, Optical channel 5 and 6 is 200G.

The following figure explains the 100G to 200G configuration in slice 0.

Figure 9: 100Gx200G Configuration

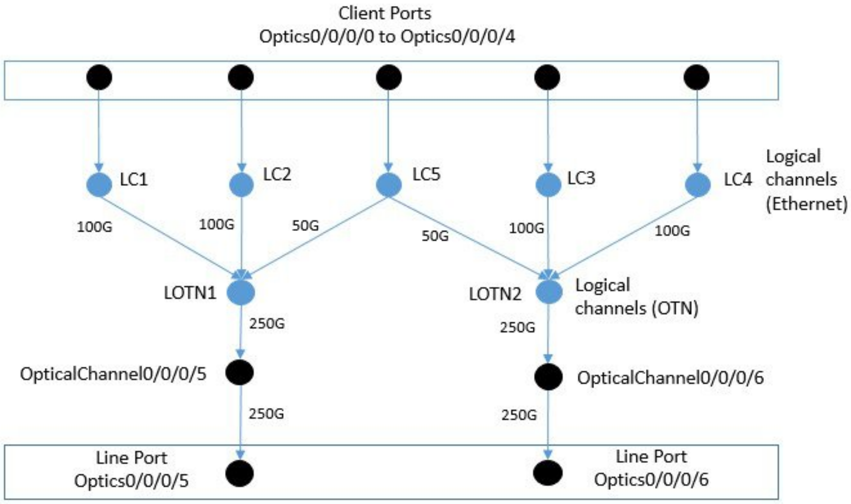


100Gx250G Configuration

For 100G to 250G configuration on slice 0, connect the logical channels, LC1, LC2, LC3, LC4 and LC5 of the client ports, Optics0/0/0/0, Optics0/0/0/1, Optics0/0/0/3, Optics0/0/0/4 and Optics0/0/0/2 to the Logical OTN channels (LOTN1 and LOTN2). The LC1 and LC2 are connected to the first logical OTN channel (LOTN1) and the LC3 and LC4 are connected to the second logical OTN channel (LOTN2). LC5 is divided and connected to LOTN1 and LOTN2. The LOTN1 is connected to the optical channel 5 and LOTN2 is connected to the Optical Channel0/0/0/6. Then the optical channel 5 is connected to the line port, Optics0/0/0/5 and the Optical Channel0/0/0/6 is connected to the line port, Optics0/0/0/6. The bandwidth allocation is 100G each for all the logical channels (LC) and the bandwidth allocation for LOTN1, LOTN2, Optical channel 5 and 6 is 250G.

The following figure explains the 100G to 250G configuration in slice 0.

Figure 10: 100Gx250G Configuration

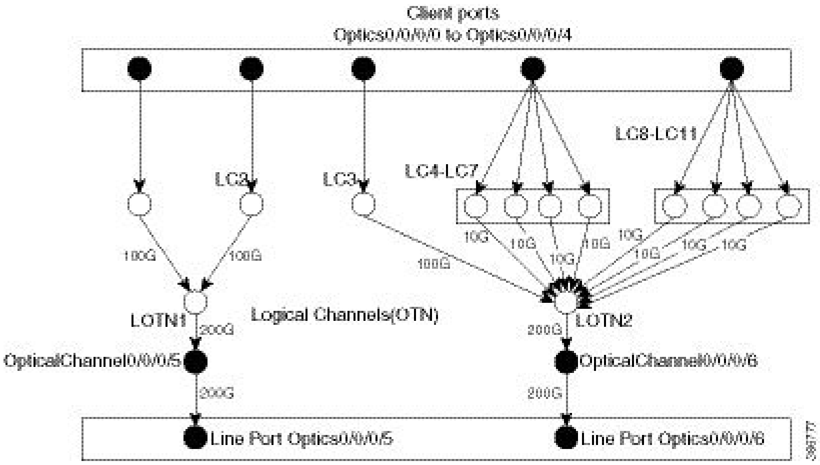


10G-100Gx200G Configuration (Mixed Mode Configuration)

For 10G-100G to 200G(Mixed mode) configuration on slice 0, connect the 11 logical channels(LC1 to LC11) of the client ports Optics0/0/0/0, Optics0/0/0/1, Optics0/0/0/2, Optics0/0/0/3 and Optics0/0/0/4 to the Logical OTN channels (LOTN1 and LOTN2). The LC1 and LC2 are connected to the first logical OTN channel (LOTN1) and the LC3 to LC11 are connected to the second logical OTN channel (LOTN2). The LOTN1 is connected to the Optical Channel 0/0/0/5 and LOTN2 is connected to the Optical Channel 0/0/0/6. Then the Optical Channel 0/0/0/5 is connected to the line port Optics0/0/0/5 and the Optical Channel 0/0/0/6is connected to the line port Optics0/0/0/6. The bandwidth allocation is 100G for the logical channels LC1 to LC3 and it is 10G for LC4 to LC11. The bandwidth allocation for LOTN1, LOTN2, Optical channels 0/0/0/5 and 0/0/0/6 is 200G.

The following figure explains the 10G-100G to 200G configuration in slice 0.

Figure 11: 10G-100G to 200G Configuration



Sample Configuration

The following is a sample in which slice 0 is configured to send the traffic with 100G client rate and 200G trunk rate in .json and .xml format.

Sample .xml file:

```
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<edit-config>
<target>
<candidate/>
</target>
<config xmlns:xc="urn:ietf:params:xml:ns:netconf:base:1.0">
<terminal-device xmlns="http://openconfig.net/yang/terminal-device">
   <logical-channels>
    <channel>
     <index>201</index>
     <config>
      <index>201</index>
      <rate-class>TRIB_RATE_100G</rate-class>
      <admin-state>ENABLED</admin-state>
      <description>Logical channel 201</description>
      <loopback-mode>NONE</loopback-mode>
      <trib-protocol>PROT_100G_MLG</trib-protocol>
      <logical-channel-type>PROT_ETHERNET</logical-channel-type>
```

```
 </config>
 <ingress>
  <config>
   <transceiver>0/0-Optics0/0/0/14</transceiver>
  </config>
 </ingress>
 <logical-channel-assignments>
  <assignment>
   <index>1</index>
   <config>
    <index>1</index>
    <allocation>100</allocation>
    <assignment-type>LOGICAL_CHANNEL</assignment-type>
    <description>Logical channel assigned 203</description>
    <logical-channel>203</logical-channel>
   </config>
  </assignment>
 </logical-channel-assignments>
</channel>
<channel>
 <index>202</index>
 <config>
  <index>202</index>
  <rate-class>TRIB_RATE_100G</rate-class>
  <admin-state>ENABLED</admin-state>
  <description>Logical channel 202</description>
  <loopback-mode>NONE</loopback-mode>
  <trib-protocol>PROT_100G_MLG</trib-protocol>
  <logical-channel-type>PROT_ETHERNET</logical-channel-type>
 </config>
 <ingress>
  <config>
   <transceiver>0/0-Optics0/0/0/15</transceiver>
  </config>
 </ingress>
 <logical-channel-assignments>
  <assignment>
   <index>1</index>
   <config>
    <index>1</index>
    <allocation>100</allocation>
    <assignment-type>LOGICAL_CHANNEL</assignment-type>
    <description>Logical channel assigned 203</description>
    <logical-channel>203</logical-channel>
   </config>
  </assignment>
 </logical-channel-assignments>
</channel>
<channel>
 <index>203</index>
 <config>
  <index>203</index>
  <admin-state>ENABLED</admin-state>
  <description>Logical channel 203</description>
  <loopback-mode>NONE</loopback-mode>
  <logical-channel-type>PROT_OTN</logical-channel-type>
 </config>
 <otn>
  <config>
   <tti-msg-expected>test1</tti-msg-expected>
   <tti-msg-transmit>test1</tti-msg-transmit>
  </config>
 </otn>
 <logical-channel-assignments>
```

```
  <assignment>
   <index>1</index>
   <config>
    <index>1</index>
    <allocation>200</allocation>
    <assignment-type>OPTICAL_CHANNEL</assignment-type>
    <description>Optical channel assigned a</description>
    <optical-channel>0/0-OpticalChannel0/0/0/19</optical-channel>
   </config>
  </assignment>
 </logical-channel-assignments>
</channel>
<channel>
 <index>204</index>
 <config>
  <index>204</index>
  <admin-state>ENABLED</admin-state>
  <description>Logical channel 204</description>
  <loopback-mode>NONE</loopback-mode>
  <logical-channel-type>PROT_OTN</logical-channel-type>
 </config>
 <otn>
  <config>
   <tti-msg-expected>test2</tti-msg-expected>
   <tti-msg-transmit>test2</tti-msg-transmit>
  </config>
 </otn>
 <logical-channel-assignments>
  <assignment>
   <index>1</index>
   <config>
    <index>1</index>
    <allocation>200</allocation>
    <assignment-type>OPTICAL_CHANNEL</assignment-type>
    <description>Optical channel assigned a</description>
    <optical-channel>0/0-OpticalChannel0/0/0/20</optical-channel>
   </config>
  </assignment>
 </logical-channel-assignments>
</channel>
<channel>
 <index>205</index>
 <config>
  <index>205</index>
  <rate-class>TRIB_RATE_100G</rate-class>
  <admin-state>ENABLED</admin-state>
  <description>Logical channel 205</description>
  <loopback-mode>NONE</loopback-mode>
  <trib-protocol>PROT_100G_MLG</trib-protocol>
  <logical-channel-type>PROT_ETHERNET</logical-channel-type>
 </config>
 <ingress>
  <config>
   <transceiver>0/0-Optics0/0/0/17</transceiver>
  </config>
 </ingress>
 <logical-channel-assignments>
  <assignment>
   <index>1</index>
   <config>
    <index>1</index>
    <allocation>100</allocation>
    <assignment-type>LOGICAL_CHANNEL</assignment-type>
    <description>Logical channel assigned 204</description>
```

```
      <logical-channel>204</logical-channel>
     </config>
    </assignment>
   </logical-channel-assignments>
  </channel>
  <channel>
   <index>206</index>
   <config>
    <index>206</index>
    <rate-class>TRIB_RATE_100G</rate-class>
    <admin-state>ENABLED</admin-state>
    <description>Logical channel 206</description>
    <loopback-mode>NONE</loopback-mode>
    <trib-protocol>PROT_100G_MLG</trib-protocol>
    <logical-channel-type>PROT_ETHERNET</logical-channel-type>
   </config>
   <ingress>
    <config>
     <transceiver>0/0-Optics0/0/0/18</transceiver>
    </config>
   </ingress>
   <logical-channel-assignments>
    <assignment>
     <index>1</index>
     <config>
      <index>1</index>
      <allocation>100</allocation>
      <assignment-type>LOGICAL_CHANNEL</assignment-type>
      <description>Logical channel assigned 204</description>
      <logical-channel>204</logical-channel>
     </config>
    </assignment>
   </logical-channel-assignments>
  </channel>
 </logical-channels>
</terminal-device>
  <components xmlns="http://openconfig.net/yang/platform">
 <component>
  <name>0/0-OpticalChannel0/0/0/19</name>
  <optical-channel xmlns="http://openconfig.net/yang/terminal-device">
   <config>
    <target-output-power>100</target-output-power>
    <frequency>191600000</frequency>
    <line-port>0/0-Optics0/0/0/19</line-port>
    <operational-mode>2</operational-mode>
   </config>
  </optical-channel>
 </component>
 <component>
  <name>0/0-OpticalChannel0/0/0/20</name>
  <optical-channel xmlns="http://openconfig.net/yang/terminal-device">
   <config>
    <target-output-power>100</target-output-power>
    <frequency>191600000</frequency>
    <line-port>0/0-Optics0/0/0/20</line-port>
    <operational-mode>2</operational-mode>
   </config>
  </optical-channel>
 </component>
 <component>
  <name>0/0-Optics0/0/0/20</name>
 </component>
 <component>
  <name>0/0-Optics0/0/0/19</name>
```

```
   </component>
  </components>
</config>
</edit-config>
</rpc>
```

Sample .json file:

```
{
"openconfig-interfaces:interfaces": {
  "interface": [
   {
    "name": "Optics0/0/0/5",
    "config": {
     "name": "Optics0/0/0/5",
     "type": "iana-if-type:opticalChannel",
     "enabled": false
    }
   },
   {
    "name": "Optics0/0/0/6",
    "config": {
     "name": "Optics0/0/0/6",
     "type": "iana-if-type:opticalChannel",
     "enabled": false
    }
   },
   {
    "name": "Optics0/0/0/12",
    "config": {
     "name": "Optics0/0/0/12",
     "type": "iana-if-type:opticalChannel",
     "enabled": false
    }
   },
  {
    "name": "Optics0/0/0/13",
    "config": {
     "name": "Optics0/0/0/13",
     "type": "iana-if-type:opticalChannel",
     "enabled": false
    }
   },
   {
    "name": "Optics0/0/0/19",
    "config": {
     "name": "Optics0/0/0/19",
     "type": "iana-if-type:opticalChannel",
     "enabled": false
    }
   },
   {
    "name": "Optics0/0/0/20",
    "config": {
     "name": "Optics0/0/0/20",
     "type": "iana-if-type:opticalChannel",
     "enabled": false
    }
   },
   {
    "name": "Optics0/0/0/26",
    "config": {
     "name": "Optics0/0/0/26",
     "type": "iana-if-type:opticalChannel",
     "enabled": false
    }
   },
{
    "name": "Optics0/0/0/27",
    "config": {
     "name": "Optics0/0/0/27",
     "type": "iana-if-type:opticalChannel",
     "enabled": false
    }
   }
]
},
"openconfig-terminal-device:terminal-device": {
   "logical-channels": {
    "channel": [
     {
      "index": 201,
      "config": {
       "rate-class": "TRIB_RATE_100G",
       "admin-state": "ENABLED",
       "description": "Logical channel 201",
       "trib-protocol": "PROT_100G_MLG",
       "logical-channel-type": "PROT_ETHERNET",
       "loopback-mode": "NONE"
      },
      "ingress": {
       "config": {
        "transceiver": "0/0-Optics0/0/0/14"
       }
      },
      "logical-channel-assignments": {
       "assignment": [
        {
         "index": 1,
         "config": {
          "allocation": 100,
          "assignment-type": "LOGICAL_CHANNEL",
          "description": "Logical channel assigned 203",
          "logical-channel": 203
         }
        }
       ]
      }
     },
     {
      "index": 202,
      "config": {
       "rate-class": "TRIB_RATE_100G",
       "admin-state": "ENABLED",
       "description": "Logical channel 202",
       "trib-protocol": "PROT_100G_MLG",
       "logical-channel-type": "PROT_ETHERNET",
       "loopback-mode": "NONE"
      },
      "ingress": {
       "config": {
        "transceiver": "0/0-Optics0/0/0/15"
       }
      },
      "logical-channel-assignments": {
       "assignment": [
        {
         "index": 1,
         "config": {
```

```
     "allocation": 100,
     "assignment-type": "LOGICAL_CHANNEL",
     "description": "Logical channel assigned 203",
     "logical-channel": 203
    }
   }
  ]
 }
},
{
 "index": 203,
 "config": {
  "admin-state": "ENABLED",
  "description": "Logical channel 203",
  "logical-channel-type": "PROT_OTN",
  "loopback-mode": "NONE"
 },

 "otn":{
 "config": {
 "tti-msg-transmit": "test1",
 "tti-msg-expected": "test1"
 }
 },

 "logical-channel-assignments": {
  "assignment": [
   {
    "index": 1,
    "config": {
     "allocation": 200,
     "assignment-type": "OPTICAL_CHANNEL",
     "description": "Optical channel assigned a",
     "optical-channel": "0/0-OpticalChannel0/0/0/19"
    }
   }
  ]
 }
},
{
 "index": 204,
 "config": {
  "admin-state": "ENABLED",
  "description": "Logical channel 204",
  "logical-channel-type": "PROT_OTN",
  "loopback-mode": "NONE"
 },
 "otn":{
 "config": {
 "tti-msg-transmit": "test2",
 "tti-msg-expected": "test2"
 }
 },

 "logical-channel-assignments": {
  "assignment": [
   {
    "index": 1,
    "config": {
     "allocation": 200,
     "assignment-type": "OPTICAL_CHANNEL",
     "description": "Optical channel assigned a",
     "optical-channel": "0/0-OpticalChannel0/0/0/20"
    }
```

```
     }
    ]
   }
  },
  {
   "index": 205,
   "config": {
    "rate-class": "TRIB_RATE_100G",
    "admin-state": "ENABLED",
    "description": "Logical channel 205",
    "trib-protocol": "PROT_100G_MLG",
    "logical-channel-type": "PROT_ETHERNET",
    "loopback-mode": "NONE"
   },
   "ingress": {
    "config": {
     "transceiver": "0/0-Optics0/0/0/17"
    }
   },
   "logical-channel-assignments": {
    "assignment": [
     {
      "index": 1,
      "config": {
       "allocation": 100,
       "assignment-type": "LOGICAL_CHANNEL",
       "description": "Logical channel assigned 204",
       "logical-channel": 204
      }
     }
    ]
   }
  },
  {
   "index": 206,
   "config": {
    "rate-class": "TRIB_RATE_100G",
    "admin-state": "ENABLED",
    "description": "Logical channel 206",
    "trib-protocol": "PROT_100G_MLG",
    "logical-channel-type": "PROT_ETHERNET",
    "loopback-mode": "NONE"
   },
   "ingress": {
    "config": {
     "transceiver": "0/0-Optics0/0/0/18"
    }
   },
   "logical-channel-assignments": {
    "assignment": [
     {
      "index": 1,
      "config": {
       "allocation": 100,
       "assignment-type": "LOGICAL_CHANNEL",
       "description": "Logical channel assigned 204",
       "logical-channel": 204
      }
     }
    ]
   }
  }
 ]
}
```

```
  },
  "openconfig-platform:components": {
   "component": [
    {
     "name": "0/0-OpticalChannel0/0/0/19",
     "openconfig-terminal-device:optical-channel": {
      "config": {
       "line-port": "0/0-Optics0/0/0/19",
       "operational-mode": 2,
       "target-output-power": 100,
       "frequency": 191600000
      }
     }
    },
    {
     "name": "0/0-OpticalChannel0/0/0/20",
     "openconfig-terminal-device:optical-channel": {
      "config": {
       "line-port": "0/0-Optics0/0/0/20",
       "operational-mode": 2,
       "target-output-power": 100,
       "frequency": 191600000
      }
     }
    }
   ]
  }
}
```

Sample configuration for mixed mode configuration

Sample .json file:

```
{
  "openconfig-interfaces:interfaces": {
   "interface": [
    {
     "name": "Optics0/0/0/5",
     "config": {
      "name": "Optics0/0/0/5",
      "type": "iana-if-type:opticalChannel",
      "enabled": true
     }
    },
    {
     "name": "Optics0/0/0/6",
     "config": {
      "name": "Optics0/0/0/6",
      "type": "iana-if-type:opticalChannel",
      "enabled": true
     }
    },
    {
     "name": "Optics0/0/0/12",
     "config": {
      "name": "Optics0/0/0/12",
      "type": "iana-if-type:opticalChannel",
      "enabled": true
     }
    },
    {
     "name": "Optics0/0/0/13",
     "config": {
      "name": "Optics0/0/0/13",
      "type": "iana-if-type:opticalChannel",
      "enabled": true
     }
    },
    {
     "name": "Optics0/0/0/19",
     "config": {
      "name": "Optics0/0/0/19",
      "type": "iana-if-type:opticalChannel",
      "enabled": true
     }
    },
    {
     "name": "Optics0/0/0/20",
     "config": {
      "name": "Optics0/0/0/20",
      "type": "iana-if-type:opticalChannel",
      "enabled": true
     }
    },
    {
     "name": "Optics0/0/0/26",
     "config": {
      "name": "Optics0/0/0/26",
      "type": "iana-if-type:opticalChannel",
      "enabled": true
     }
    },
    {
     "name": "Optics0/0/0/27",
     "config": {
      "name": "Optics0/0/0/27",
      "type": "iana-if-type:opticalChannel",
      "enabled": true
     }
    }
   ]
  },
  "openconfig-platform:components": {
   "component": [
    {
     "name": "0/0-OpticalChannel0/0/0/19",
     "openconfig-terminal-device:optical-channel": {
      "config": {
       "target-output-power": -100,
       "frequency": 192700000,
       "line-port": "0/0-Optics0/0/0/19",
       "operational-mode": 1
      }
     }
    },
    {
     "name": "0/0-OpticalChannel0/0/0/20",
     "openconfig-terminal-device:optical-channel": {
      "config": {
       "target-output-power": -100,
       "frequency": 192700000,
       "line-port": "0/0-Optics0/0/0/20",
       "operational-mode": 1
      }
     }
    }
   ]
  },
  "openconfig-terminal-device:terminal-device": {
```

```
"logical-channels": {
 "channel": [
  {
   "index": 1001,
   "config": {
    "rate-class": "TRIB_RATE_100G",
    "admin-state": "ENABLED",
    "description": "Logical channel 1001",
    "trib-protocol": "PROT_100G_MLG",
    "logical-channel-type": "PROT_ETHERNET"
   },
   "ingress": {
    "config": {
     "transceiver": "0/0-Optics0/0/0/14"
    }
   },
   "logical-channel-assignments": {
    "assignment": [
     {
      "index": 1,
      "config": {
       "allocation": 100,
       "assignment-type": "LOGICAL_CHANNEL",
       "description": "Logical channel assigned 1003",
       "logical-channel": 1003
      }
     }
    ]
   }
  },
  {
   "index": 1003,
   "config": {
    "admin-state": "ENABLED",
    "description": "Logical channel 1003",
    "logical-channel-type": "PROT_OTN"
   },
   "logical-channel-assignments": {
    "assignment": [
     {
      "index": 1,
      "config": {
       "allocation": 200,
       "assignment-type": "OPTICAL_CHANNEL",
       "description": "Optical channel assigned a",
       "optical-channel": "0/0-OpticalChannel0/0/0/19"
      }
     }
    ]
   }
  },
  {
   "index": 1004,
   "config": {
    "admin-state": "ENABLED",
    "description": "Logical channel 1004",
    "logical-channel-type": "PROT_OTN"
   },
   "otn":{
   "config": {
   "tti-msg-transmit": "abcd",
   "tti-msg-expected": "abcd"
   }
   },
   "logical-channel-assignments": {
    "assignment": [
     {
      "index": 1,
      "config": {
       "allocation": 200,
       "assignment-type": "OPTICAL_CHANNEL",
       "description": "Optical channel assigned a",
       "optical-channel": "0/0-OpticalChannel0/0/0/20"
      }
     }
    ]
   }
  },
  {
   "index": 1005,
   "config": {
    "rate-class": "TRIB_RATE_100G",
    "admin-state": "ENABLED",
    "description": "Logical channel 1005",
    "trib-protocol": "PROT_100G_MLG",
    "logical-channel-type": "PROT_ETHERNET"
   },
   "ingress": {
    "config": {
     "transceiver": "0/0-Optics0/0/0/15"
    }
   },
   "logical-channel-assignments": {
    "assignment": [
     {
      "index": 1,
      "config": {
       "allocation": 100,
       "assignment-type": "LOGICAL_CHANNEL",
       "description": "Logical channel assigned 1003",
       "logical-channel": 1003
      }
     }
    ]
   }
  },
   {
   "index": 1013,
   "config": {
    "rate-class": "TRIB_RATE_10G",
    "admin-state": "ENABLED",
    "description": "Logical channel 1013",
    "trib-protocol": "PROT_10GE_LAN",
    "logical-channel-type": "PROT_ETHERNET"
   },
   "ingress": {
    "config": {
     "transceiver": "0/0-Optics0/0/0/17",
     "physical-channel": [
      1
     ]
    }
   },
   "logical-channel-assignments": {
    "assignment": [
     {
      "index": 1,
      "config": {
       "allocation": 10,
       "assignment-type": "LOGICAL_CHANNEL",
       "description": "Logical channel assigned 1004",
       "logical-channel": 1004
      }
     }
    ]
   }
  },
  {
   "index": 1014,
   "config": {
    "rate-class": "TRIB_RATE_10G",
    "admin-state": "ENABLED",
    "description": "Logical channel 1014",
    "trib-protocol": "PROT_10GE_LAN",
    "logical-channel-type": "PROT_ETHERNET"
   },
   "ingress": {
    "config": {
     "transceiver": "0/0-Optics0/0/0/17",
     "physical-channel": [
      2
     ]
    }
   },
   "logical-channel-assignments": {
    "assignment": [
     {
      "index": 1,
      "config": {
       "allocation": 10,
       "assignment-type": "LOGICAL_CHANNEL",
       "description": "Logical channel assigned 1004",
       "logical-channel": 1004
      }
     }
    ]
   }
  },
  {
   "index": 1015,
   "config": {
    "rate-class": "TRIB_RATE_10G",
    "admin-state": "ENABLED",
    "description": "Logical channel 1015",
    "trib-protocol": "PROT_10GE_LAN",
    "logical-channel-type": "PROT_ETHERNET"
   },
   "ingress": {
    "config": {
     "transceiver": "0/0-Optics0/0/0/17",
     "physical-channel": [
      3
     ]
    }
   },
   "logical-channel-assignments": {
    "assignment": [
     {
      "index": 1,
      "config": {
       "allocation": 10,
       "assignment-type": "LOGICAL_CHANNEL",
       "description": "Logical channel assigned 1004",
       "logical-channel": 1004
      }
     }
    ]
   }
  },
  {
   "index": 1016,
   "config": {
    "rate-class": "TRIB_RATE_10G",
    "admin-state": "ENABLED",
    "description": "Logical channel 1016",
    "trib-protocol": "PROT_10GE_LAN",
    "logical-channel-type": "PROT_ETHERNET"
   },
   "ingress": {
    "config": {
     "transceiver": "0/0-Optics0/0/0/17",
     "physical-channel": [
      4
     ]
    }
   },
   "logical-channel-assignments": {
    "assignment": [
     {
      "index": 1,
      "config": {
       "allocation": 10,
       "assignment-type": "LOGICAL_CHANNEL",
       "description": "Logical channel assigned 1004",
       "logical-channel": 1004
      }
     }
    ]
   }
  },
  {
   "index": 1017,
   "config": {
    "rate-class": "TRIB_RATE_10G",
    "admin-state": "ENABLED",
    "description": "Logical channel 1017",
    "trib-protocol": "PROT_10GE_LAN",
    "logical-channel-type": "PROT_ETHERNET"
   },
   "ingress": {
    "config": {
     "transceiver": "0/0-Optics0/0/0/18",
     "physical-channel": [
      1
     ]
    }
   },
   "logical-channel-assignments": {
    "assignment": [
     {
      "index": 1,
      "config": {
       "allocation": 10,
       "assignment-type": "LOGICAL_CHANNEL",
       "description": "Logical channel assigned 1004",
       "logical-channel": 1004
      }
     }
    ]
   }
  },
  {
   "index": 1018,
   "config": {
    "rate-class": "TRIB_RATE_10G",
    "admin-state": "ENABLED",
    "description": "Logical channel 1018",
    "trib-protocol": "PROT_10GE_LAN",
    "logical-channel-type": "PROT_ETHERNET"
   },
   "ingress": {
    "config": {
     "transceiver": "0/0-Optics0/0/0/18",
     "physical-channel": [
      2
     ]
    }
   },
   "logical-channel-assignments": {
    "assignment": [
     {
      "index": 1,
      "config": {
       "allocation": 10,
       "assignment-type": "LOGICAL_CHANNEL",
       "description": "Logical channel assigned 1004",
       "logical-channel": 1004
      }
     }
    ]
   }
  },
  {
   "index": 1019,
   "config": {
    "rate-class": "TRIB_RATE_10G",
    "admin-state": "ENABLED",
    "description": "Logical channel 1019",
    "trib-protocol": "PROT_10GE_LAN",
    "logical-channel-type": "PROT_ETHERNET"
   },
   "ingress": {
    "config": {
     "transceiver": "0/0-Optics0/0/0/18",
     "physical-channel": [
      3
     ]
    }
   },
   "logical-channel-assignments": {
    "assignment": [
     {
      "index": 1,
      "config": {
       "allocation": 10,
       "assignment-type": "LOGICAL_CHANNEL",
       "description": "Logical channel assigned 1004",
       "logical-channel": 1004
      }
     }
    ]
   }
  },
  {
   "index": 1020,
   "config": {
    "rate-class": "TRIB_RATE_10G",
    "admin-state": "ENABLED",
    "description": "Logical channel 1020",
    "trib-protocol": "PROT_10GE_LAN",
    "logical-channel-type": "PROT_ETHERNET"
   },
   "ingress": {
    "config": {
     "transceiver": "0/0-Optics0/0/0/18",
     "physical-channel": [
      4
     ]
    }
   },
   "logical-channel-assignments": {
    "assignment": [
     {
      "index": 1,
      "config": {
       "allocation": 10,
       "assignment-type": "LOGICAL_CHANNEL",
       "description": "Logical channel assigned 1004",
       "logical-channel": 1004
      }
     }
    ]
   }
  },
  {
   "index": 1007,
   "config": {
    "rate-class": "TRIB_RATE_100G",
    "admin-state": "ENABLED",
    "description": "Logical channel 1006",
    "trib-protocol": "PROT_100G_MLG",
    "logical-channel-type": "PROT_ETHERNET"
   },
   "ingress": {
    "config": {
     "transceiver": "0/0-Optics0/0/0/16"
    }
   },
   "logical-channel-assignments": {
    "assignment": [
     {
      "index": 1,
      "config": {
       "allocation": 100,
       "assignment-type": "LOGICAL_CHANNEL",
       "description": "Logical channel assigned 1004",
       "logical-channel": 1004
      }
     }
    ]
   }
  }
 ]
```

```
   }
  }
}
```

Verifying Terminal-device Configuration

Use the following commands to verify that you have correctly configured terminal-device configuration for Cisco NCS1004.

To view the configuration layout of channels, use the following command:

```
Router#show terminal-device layout-all osa
Wed Aug 28 07:49:47.495 UTC

Output format: Channel-name [Channel-type, Bandwidth]

CP - Client port : LP - Line port : OP - Optics
OTN - OTN Logical channel : ETH - Ethernet Logical channel

Optics0/0/0/2 [CP, 100] 30002 [ETH, 100] 30020 [ODU4, 100] 30000 [OTU, 500]
0/0-OpticalChannel0/0/0/0 [OP, 500] Optics0/0/0/0 [LP, 500]
Optics0/0/0/3 [CP, 100] 30003 [ETH, 100] 30021 [ODU4, 100] 30000 [OTU, 500]
0/0-OpticalChannel0/0/0/0 [OP, 500] Optics0/0/0/0 [LP, 500]
Optics0/0/0/4 [CP, 100] 30004 [ETH, 100] 30022 [ODU4, 100] 30000 [OTU, 500]
0/0-OpticalChannel0/0/0/0 [OP, 500] Optics0/0/0/0 [LP, 500]
Optics0/0/0/5 [CP, 100] 30005 [ETH, 100] 30023 [ODU4, 100] 30000 [OTU, 500]
0/0-OpticalChannel0/0/0/0 [OP, 500] Optics0/0/0/0 [LP, 500]
Optics0/0/0/6 [CP, 100] 30006 [ETH, 100] 30024 [ODU4, 100] 30000 [OTU, 500]
0/0-OpticalChannel0/0/0/0 [OP, 500] Optics0/0/0/0 [LP, 500]
Optics0/0/0/8 [CP, 100] 30007 [ETH, 100] 30025 [ODU4, 100] 30001 [OTU, 500]
0/0-OpticalChannel0/0/0/1 [OP, 500] Optics0/0/0/1 [LP, 500]
Optics0/0/0/9 [CP, 100] 30008 [ETH, 100] 30026 [ODU4, 100] 30001 [OTU, 500]
0/0-OpticalChannel0/0/0/1 [OP, 500] Optics0/0/0/1 [LP, 500]
Optics0/0/0/10 [CP, 100] 30009 [ETH, 100] 30027 [ODU4, 100] 30001 [OTU, 500]
0/0-OpticalChannel0/0/0/1 [OP, 500] Optics0/0/0/1 [LP, 500]
Optics0/0/0/11 [CP, 100] 30010 [ETH, 100] 30028 [ODU4, 100] 30001 [OTU, 500]
0/0-OpticalChannel0/0/0/1 [OP, 500] Optics0/0/0/1 [LP, 500]
Optics0/0/0/12 [CP, 100] 30011 [ETH, 100] 30029 [ODU4, 100] 30001 [OTU, 500]
0/0-OpticalChannel0/0/0/1 [OP, 500] Optics0/0/0/1 [LP, 500]
Optics0/1/0/2 [CP, 100] 40002 [ETH, 100] 40020 [ODU4, 100] 40000 [OTU, 500]
0/1-OpticalChannel0/1/0/0 [OP, 500] Optics0/1/0/0 [LP, 500]
Optics0/1/0/3 [CP, 100] 40003 [ETH, 100] 40021 [ODU4, 100] 40000 [OTU, 500]
0/1-OpticalChannel0/1/0/0 [OP, 500] Optics0/1/0/0 [LP, 500]
Optics0/1/0/4 [CP, 100] 40004 [ETH, 100] 40022 [ODU4, 100] 40000 [OTU, 500]
0/1-OpticalChannel0/1/0/0 [OP, 500] Optics0/1/0/0 [LP, 500]
Optics0/1/0/5 [CP, 100] 40005 [ETH, 100] 40023 [ODU4, 100] 40000 [OTU, 500]
0/1-OpticalChannel0/1/0/0 [OP, 500] Optics0/1/0/0 [LP, 500]
Optics0/1/0/6 [CP, 100] 40006 [ETH, 100] 40024 [ODU4, 100] 40000 [OTU, 500]
0/1-OpticalChannel0/1/0/0 [OP, 500] Optics0/1/0/0 [LP, 500]
Optics0/1/0/8 [CP, 100] 40007 [ETH, 100] 40025 [ODU4, 100] 40001 [OTU, 500]
0/1-OpticalChannel0/1/0/1 [OP, 500] Optics0/1/0/1 [LP, 500]
Optics0/1/0/9 [CP, 100] 40008 [ETH, 100] 40026 [ODU4, 100] 40001 [OTU, 500]
0/1-OpticalChannel0/1/0/1 [OP, 500] Optics0/1/0/1 [LP, 500]
Optics0/1/0/10 [CP, 100] 40009 [ETH, 100] 40027 [ODU4, 100] 40001 [OTU, 500]
0/1-OpticalChannel0/1/0/1 [OP, 500] Optics0/1/0/1 [LP, 500]
Optics0/1/0/11 [CP, 100] 40010 [ETH, 100] 40028 [ODU4, 100] 40001 [OTU, 500]
0/1-OpticalChannel0/1/0/1 [OP, 500] Optics0/1/0/1 [LP, 500]
Optics0/1/0/12 [CP, 100] 40011 [ETH, 100] 40029 [ODU4, 100] 40001 [OTU, 500]
0/1-OpticalChannel0/1/0/1 [OP, 500] Optics0/1/0/1 [LP, 500]
Optics0/2/0/2 [CP, 100] 50002 [ETH, 100] 50020 [ODU4, 100] 50000 [OTU, 500]
0/2-OpticalChannel0/2/0/0 [OP, 500] Optics0/2/0/0 [LP, 500]
Optics0/2/0/3 [CP, 100] 50003 [ETH, 100] 50021 [ODU4, 100] 50000 [OTU, 500]
0/2-OpticalChannel0/2/0/0 [OP, 500] Optics0/2/0/0 [LP, 500]
```

```
Optics0/2/0/4 [CP, 100] 50004 [ETH, 100] 50022 [ODU4, 100] 50000 [OTU, 500]
0/2-OpticalChannel0/2/0/0 [OP, 500] Optics0/2/0/0 [LP, 500]
Optics0/2/0/5 [CP, 100] 50005 [ETH, 100] 50023 [ODU4, 100] 50000 [OTU, 500]
0/2-OpticalChannel0/2/0/0 [OP, 500] Optics0/2/0/0 [LP, 500]
Optics0/2/0/6 [CP, 100] 50006 [ETH, 100] 50024 [ODU4, 100] 50000 [OTU, 500]
0/2-OpticalChannel0/2/0/0 [OP, 500] Optics0/2/0/0 [LP, 500]
Optics0/2/0/8 [CP, 100] 50007 [ETH, 100] 50025 [ODU4, 100] 50001 [OTU, 500]
0/2-OpticalChannel0/2/0/1 [OP, 500] Optics0/2/0/1 [LP, 500]
Optics0/2/0/9 [CP, 100] 50008 [ETH, 100] 50026 [ODU4, 100] 50001 [OTU, 500]
0/2-OpticalChannel0/2/0/1 [OP, 500] Optics0/2/0/1 [LP, 500]
Optics0/2/0/10 [CP, 100] 50009 [ETH, 100] 50027 [ODU4, 100] 50001 [OTU, 500]
0/2-OpticalChannel0/2/0/1 [OP, 500] Optics0/2/0/1 [LP, 500]
Optics0/2/0/11 [CP, 100] 50010 [ETH, 100] 50028 [ODU4, 100] 50001 [OTU, 500]
0/2-OpticalChannel0/2/0/1 [OP, 500] Optics0/2/0/1 [LP, 500]
Optics0/2/0/12 [CP, 100] 50011 [ETH, 100] 50029 [ODU4, 100] 50001 [OTU, 500]
0/2-OpticalChannel0/2/0/1 [OP, 500] Optics0/2/0/1 [LP, 500]
Optics0/3/0/2 [CP, 100] 60002 [ETH, 100] 60020 [ODU4, 100] 60000 [OTU, 500]
0/3-OpticalChannel0/3/0/0 [OP, 500] Optics0/3/0/0 [LP, 500]
Optics0/3/0/3 [CP, 100] 60003 [ETH, 100] 60021 [ODU4, 100] 60000 [OTU, 500]
0/3-OpticalChannel0/3/0/0 [OP, 500] Optics0/3/0/0 [LP, 500]
Optics0/3/0/4 [CP, 100] 60004 [ETH, 100] 60022 [ODU4, 100] 60000 [OTU, 500]
0/3-OpticalChannel0/3/0/0 [OP, 500] Optics0/3/0/0 [LP, 500]
Optics0/3/0/5 [CP, 100] 60005 [ETH, 100] 60023 [ODU4, 100] 60000 [OTU, 500]
0/3-OpticalChannel0/3/0/0 [OP, 500] Optics0/3/0/0 [LP, 500]
Optics0/3/0/6 [CP, 100] 60006 [ETH, 100] 60024 [ODU4, 100] 60000 [OTU, 500]
0/3-OpticalChannel0/3/0/0 [OP, 500] Optics0/3/0/0 [LP, 500]
Optics0/3/0/8 [CP, 100] 60007 [ETH, 100] 60025 [ODU4, 100] 60001 [OTU, 500]
0/3-OpticalChannel0/3/0/1 [OP, 500] Optics0/3/0/1 [LP, 500]
Optics0/3/0/9 [CP, 100] 60008 [ETH, 100] 60026 [ODU4, 100] 60001 [OTU, 500]
0/3-OpticalChannel0/3/0/1 [OP, 500] Optics0/3/0/1 [LP, 500]
Optics0/3/0/10 [CP, 100] 60009 [ETH, 100] 60027 [ODU4, 100] 60001 [OTU, 500]
0/3-OpticalChannel0/3/0/1 [OP, 500] Optics0/3/0/1 [LP, 500]
Optics0/3/0/11 [CP, 100] 60010 [ETH, 100] 60028 [ODU4, 100] 60001 [OTU, 500]
0/3-OpticalChannel0/3/0/1 [OP, 500] Optics0/3/0/1 [LP, 500]
Optics0/3/0/12 [CP, 100] 60011 [ETH, 100] 60029 [ODU4, 100] 60001 [OTU, 500]
0/3-OpticalChannel0/3/0/1 [OP, 500] Optics0/3/0/1 [LP, 500]
```

To view specific logical channel details, use the following command:

```
Router#show terminal-device logical-channel osa number 30000
Wed Aug 28 12:25:27.054 UTC
Logical Channel Index:      30000
Name:                       CoherentDSP0/0/0/0
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              Not Configured
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            NA
Optical Channel:            0_0-OpticalChannel0_0_0_0
Allocation:                 500G
Assignment Type:            Optical
RP/0/RP0/CPU0:BH1_P2A4#
```

To view all the logical channels details, use the following command:

```
Router#show terminal-device logical-channel osa all
Wed Aug 28 12:23:25.611 UTC
```

```
Logical Channel Index:      30000
Name:                       CoherentDSP0/0/0/0
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              Not Configured
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            NA
Optical Channel:            0_0-OpticalChannel0_0_0_0
Allocation:                 500G
Assignment Type:            Optical

Logical Channel Index:      30001
Name:                       CoherentDSP0/0/0/1
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              Not Configured
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            NA
Optical Channel:            0_0-OpticalChannel0_0_0_1
Allocation:                 500G
Assignment Type:            Optical

Logical Channel Index:      30002
Name:                       HundredGigECtrlr0/0/0/2
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/0/0/2
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            30020
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      30003
Name:                       HundredGigECtrlr0/0/0/3
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/0/0/3
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:                NA
Logical Assignment Index:    1
Logical Assignment Name:     NA
Logical Channel:             30021
Optical Channel:             NA
Allocation:                  100G
Assignment Type:             Logical

Logical Channel Index:       30004
Name:                        HundredGigECtrlr0/0/0/4
Admin-State:                 Enable
Loopback-Mode:               None
Rate-Class:                  100G tributary signal rate
Trib-Protocol:               100G MLG protocol
Protocol-Type:               Ethernet protocol framing
Ingress Client Port:         Optics0/0/0/4
Ingress Physical Channel:    0
TTI Transmit:                NA
TTI Expected:                NA
Logical Assignment Index:    1
Logical Assignment Name:     NA
Logical Channel:             30022
Optical Channel:             NA
Allocation:                  100G
Assignment Type:             Logical

Logical Channel Index:       30005
Name:                        HundredGigECtrlr0/0/0/5
Admin-State:                 Enable
Loopback-Mode:               None
Rate-Class:                  100G tributary signal rate
Trib-Protocol:               100G MLG protocol
Protocol-Type:               Ethernet protocol framing
Ingress Client Port:         Optics0/0/0/5
Ingress Physical Channel:    0
TTI Transmit:                NA
TTI Expected:                NA
Logical Assignment Index:    1
Logical Assignment Name:     NA
Logical Channel:             30023
Optical Channel:             NA
Allocation:                  100G
Assignment Type:             Logical

Logical Channel Index:       30006
Name:                        HundredGigECtrlr0/0/0/6
Admin-State:                 Enable
Loopback-Mode:               None
Rate-Class:                  100G tributary signal rate
Trib-Protocol:               100G MLG protocol
Protocol-Type:               Ethernet protocol framing
Ingress Client Port:         Optics0/0/0/6
Ingress Physical Channel:    0
TTI Transmit:                NA
TTI Expected:                NA
Logical Assignment Index:    1
Logical Assignment Name:     NA
Logical Channel:             30024
Optical Channel:             NA
Allocation:                  100G
Assignment Type:             Logical

Logical Channel Index:       30007
Name:                        HundredGigECtrlr0/0/0/8
```

```
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/0/0/8
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            30025
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      30008
Name:                       HundredGigECtrlr0/0/0/9
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/0/0/9
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            30026
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      30009
Name:                       HundredGigECtrlr0/0/0/10
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/0/0/10
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            30027
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      30010
Name:                       HundredGigECtrlr0/0/0/11
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/0/0/11
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            30028
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      30011
Name:                       HundredGigECtrlr0/0/0/12
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/0/0/12
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            30029
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      30020
Name:                       ODU40/0/0/0/1
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            30000
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      30021
Name:                       ODU40/0/0/0/2
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            30000
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      30022
Name:                       ODU40/0/0/0/3
Admin-State:                Enable
Loopback-Mode:              None
```

```
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            30000
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      30023
Name:                       ODU40/0/0/0/4
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            30000
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      30024
Name:                       ODU40/0/0/0/5
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            30000
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      30025
Name:                       ODU40/0/0/1/1
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            30001
```

```
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      30026
Name:                       ODU40/0/0/1/2
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            30001
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      30027
Name:                       ODU40/0/0/1/3
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            30001
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      30028
Name:                       ODU40/0/0/1/4
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            30001
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      30029
Name:                       ODU40/0/0/1/5
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
```

```
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            30001
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      40000
Name:                       CoherentDSP0/1/0/0
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              Not Configured
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            NA
Optical Channel:            0_1-OpticalChannel0_1_0_0
Allocation:                 500G
Assignment Type:            Optical

Logical Channel Index:      40001
Name:                       CoherentDSP0/1/0/1
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              Not Configured
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            NA
Optical Channel:            0_1-OpticalChannel0_1_0_1
Allocation:                 500G
Assignment Type:            Optical

Logical Channel Index:      40002
Name:                       HundredGigECtrlr0/1/0/2
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/1/0/2
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            40020
Optical Channel:            NA
Allocation:                 100G
```

```
Assignment Type:            Logical

Logical Channel Index:      40003
Name:                       HundredGigECtrlr0/1/0/3
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/1/0/3
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            40021
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      40004
Name:                       HundredGigECtrlr0/1/0/4
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/1/0/4
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            40022
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      40005
Name:                       HundredGigECtrlr0/1/0/5
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/1/0/5
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            40023
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      40006
Name:                       HundredGigECtrlr0/1/0/6
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/1/0/6
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            40024
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      40007
Name:                       HundredGigECtrlr0/1/0/8
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/1/0/8
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            40025
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      40008
Name:                       HundredGigECtrlr0/1/0/9
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/1/0/9
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            40026
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      40009
Name:                       HundredGigECtrlr0/1/0/10
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/1/0/10
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            40027
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical
```

```
Logical Channel Index:      40010
Name:                       HundredGigECtrlr0/1/0/11
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/1/0/11
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            40028
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      40011
Name:                       HundredGigECtrlr0/1/0/12
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/1/0/12
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            40029
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      40020
Name:                       ODU40/1/0/0/1
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            40000
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      40021
Name:                       ODU40/1/0/0/2
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:                 NA
Logical Assignment Index:     1
Logical Assignment Name:      NA
Logical Channel:              40000
Optical Channel:              NA
Allocation:                   100G
Assignment Type:              Logical

Logical Channel Index:        40022
Name:                         ODU40/1/0/0/3
Admin-State:                  Enable
Loopback-Mode:                None
Rate-Class:                   Not Configured
Trib-Protocol:                ODU 4 protocol
Protocol-Type:                OTN protocol framing
Ingress Client Port:          NA
Ingress Physical Channel:     0
TTI Transmit:                 NA
TTI Expected:                 NA
Logical Assignment Index:     1
Logical Assignment Name:      NA
Logical Channel:              40000
Optical Channel:              NA
Allocation:                   100G
Assignment Type:              Logical

Logical Channel Index:        40023
Name:                         ODU40/1/0/0/4
Admin-State:                  Enable
Loopback-Mode:                None
Rate-Class:                   Not Configured
Trib-Protocol:                ODU 4 protocol
Protocol-Type:                OTN protocol framing
Ingress Client Port:          NA
Ingress Physical Channel:     0
TTI Transmit:                 NA
TTI Expected:                 NA
Logical Assignment Index:     1
Logical Assignment Name:      NA
Logical Channel:              40000
Optical Channel:              NA
Allocation:                   100G
Assignment Type:              Logical

Logical Channel Index:        40024
Name:                         ODU40/1/0/0/5
Admin-State:                  Enable
Loopback-Mode:                None
Rate-Class:                   Not Configured
Trib-Protocol:                ODU 4 protocol
Protocol-Type:                OTN protocol framing
Ingress Client Port:          NA
Ingress Physical Channel:     0
TTI Transmit:                 NA
TTI Expected:                 NA
Logical Assignment Index:     1
Logical Assignment Name:      NA
Logical Channel:              40000
Optical Channel:              NA
Allocation:                   100G
Assignment Type:              Logical

Logical Channel Index:        40025
Name:                         ODU40/1/0/1/1
```

```
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            40001
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      40026
Name:                       ODU40/1/0/1/2
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            40001
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      40027
Name:                       ODU40/1/0/1/3
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            40001
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      40028
Name:                       ODU40/1/0/1/4
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            40001
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      40029
Name:                       ODU40/1/0/1/5
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            40001
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      50000
Name:                       CoherentDSP0/2/0/0
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              Not Configured
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            NA
Optical Channel:            0_2-OpticalChannel0_2_0_0
Allocation:                 500G
Assignment Type:            Optical

Logical Channel Index:      50001
Name:                       CoherentDSP0/2/0/1
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              Not Configured
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            NA
Optical Channel:            0_2-OpticalChannel0_2_0_1
Allocation:                 500G
Assignment Type:            Optical

Logical Channel Index:      50002
Name:                       HundredGigECtrlr0/2/0/2
Admin-State:                Enable
Loopback-Mode:              None
```

```
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/2/0/2
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            50020
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      50003
Name:                       HundredGigECtrlr0/2/0/3
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/2/0/3
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            50021
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      50004
Name:                       HundredGigECtrlr0/2/0/4
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/2/0/4
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            50022
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      50005
Name:                       HundredGigECtrlr0/2/0/5
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/2/0/5
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            50023
```

```
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      50006
Name:                       HundredGigECtrlr0/2/0/6
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/2/0/6
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            50024
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      50007
Name:                       HundredGigECtrlr0/2/0/8
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/2/0/8
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            50025
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      50008
Name:                       HundredGigECtrlr0/2/0/9
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/2/0/9
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            50026
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      50009
Name:                       HundredGigECtrlr0/2/0/10
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
```

```
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/2/0/10
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            50027
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      50010
Name:                       HundredGigECtrlr0/2/0/11
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/2/0/11
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            50028
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      50011
Name:                       HundredGigECtrlr0/2/0/12
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/2/0/12
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            50029
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      50020
Name:                       ODU40/2/0/0/1
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            50000
Optical Channel:            NA
Allocation:                 100G
```

```
Assignment Type:            Logical

Logical Channel Index:      50021
Name:                       ODU40/2/0/0/2
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            50000
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      50022
Name:                       ODU40/2/0/0/3
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            50000
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      50023
Name:                       ODU40/2/0/0/4
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            50000
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      50024
Name:                       ODU40/2/0/0/5
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            50000
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      50025
Name:                       ODU40/2/0/1/1
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            50001
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      50026
Name:                       ODU40/2/0/1/2
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            50001
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      50027
Name:                       ODU40/2/0/1/3
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            50001
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical
```

```
Logical Channel Index:      50028
Name:                       ODU40/2/0/1/4
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            50001
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      50029
Name:                       ODU40/2/0/1/5
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            50001
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      60000
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              Not Configured
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            NA
Optical Channel:            0_3-OpticalChannel0_3_0_0
Allocation:                 500G
Assignment Type:            Optical

Logical Channel Index:      60001
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              Not Configured
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            NA
Optical Channel:            0_3-OpticalChannel0_3_0_1
Allocation:                 500G
Assignment Type:            Optical

Logical Channel Index:      60002
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/3/0/2
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            60020
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      60003
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/3/0/3
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            60021
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      60004
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/3/0/4
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            60022
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      60005
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/3/0/5
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            60023
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      60006
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/3/0/6
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            60024
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      60007
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/3/0/8
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            60025
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      60008
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/3/0/9
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            60026
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      60009
Admin-State:                Enable
Loopback-Mode:              None
```

```
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/3/0/10
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            60027
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      60010
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/3/0/11
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            60028
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      60011
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 100G tributary signal rate
Trib-Protocol:              100G MLG protocol
Protocol-Type:              Ethernet protocol framing
Ingress Client Port:        Optics0/3/0/12
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            60029
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      60020
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            60000
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical
```

```
Logical Channel Index:      60021
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            60000
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      60022
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            60000
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      60023
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            60000
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      60024
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            60000
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      60025
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            60001
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      60026
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            60001
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      60027
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
TTI Transmit:               NA
TTI Expected:               NA
Logical Assignment Index:   1
Logical Assignment Name:    NA
Logical Channel:            60001
Optical Channel:            NA
Allocation:                 100G
Assignment Type:            Logical

Logical Channel Index:      60028
Admin-State:                Enable
Loopback-Mode:              None
Rate-Class:                 Not Configured
Trib-Protocol:              ODU 4 protocol
Protocol-Type:              OTN protocol framing
Ingress Client Port:        NA
Ingress Physical Channel:   0
```

```
TTI Transmit:                NA
TTI Expected:                NA
Logical Assignment Index:    1
Logical Assignment Name:     NA
Logical Channel:             60001
Optical Channel:             NA
Allocation:                  100G
Assignment Type:             Logical

Logical Channel Index:       60029
Admin-State:                 Enable
Loopback-Mode:               None
Rate-Class:                  Not Configured
Trib-Protocol:               ODU 4 protocol
Protocol-Type:               OTN protocol framing
Ingress Client Port:         NA
Ingress Physical Channel:    0
TTI Transmit:                NA
TTI Expected:                NA
Logical Assignment Index:    1
Logical Assignment Name:     NA
Logical Channel:             60001
Optical Channel:             NA
Allocation:                  100G
Assignment Type:             Logical
```

To view the supported operational modes, use the following command:

```
Router#show terminal-device operational-modes
Wed Aug 28 12:24:58.115 UTC
Mode  FEC       Baud-Rate     Description
3     SD_15     0.000000      SoftDecision_FEC15:Baud_Default
4     SD_27     0.000000      SoftDecision_FEC27:Baud_Default
101   SD_27     24.020791     SoftDecision_FEC27:Baud_24.02079115
102   SD_27     24.053296     SoftDecision_FEC27:Baud_24.05329561
103   SD_27     24.064150     SoftDecision_FEC27:Baud_24.06414998
104   SD_27     24.085888     SoftDecision_FEC27:Baud_24.08588815
105   SD_27     24.107666     SoftDecision_FEC27:Baud_24.10766562
106   SD_27     24.118569     SoftDecision_FEC27:Baud_24.11856914
107   SD_27     24.151339     SoftDecision_FEC27:Baud_24.15133893
108   SD_27     24.184198     SoftDecision_FEC27:Baud_24.1841979
109   SD_27     24.195171     SoftDecision_FEC27:Baud_24.19517076
110   SD_27     24.217146     SoftDecision_FEC27:Baud_24.21714639
111   SD_27     24.239162     SoftDecision_FEC27:Baud_24.23916198
112   SD_27     24.250185     SoftDecision_FEC27:Baud_24.25018479
113   SD_27     24.283313     SoftDecision_FEC27:Baud_24.28331346
114   SD_27     24.316533     SoftDecision_FEC27:Baud_24.31653277
115   SD_27     24.327626     SoftDecision_FEC27:Baud_24.32762608
116   SD_27     24.349843     SoftDecision_FEC27:Baud_24.34984309
117   SD_27     24.372101     SoftDecision_FEC27:Baud_24.37210071
118   SD_27     24.383245     SoftDecision_FEC27:Baud_24.38324479
119   SD_27     24.416738     SoftDecision_FEC27:Baud_24.41673826
120   SD_27     24.450324     SoftDecision_FEC27:Baud_24.45032387
121   SD_27     24.461540     SoftDecision_FEC27:Baud_24.46153962
122   SD_27     24.484002     SoftDecision_FEC27:Baud_24.484002
123   SD_27     24.506506     SoftDecision_FEC27:Baud_24.50650568
124   SD_27     24.517773     SoftDecision_FEC27:Baud_24.51777304
125   SD_27     24.551637     SoftDecision_FEC27:Baud_24.55163737
126   SD_27     24.585595     SoftDecision_FEC27:Baud_24.58559537
127   SD_27     24.596936     SoftDecision_FEC27:Baud_24.59693559
128   SD_27     24.619647     SoftDecision_FEC27:Baud_24.61964744
129   SD_27     24.642401     SoftDecision_FEC27:Baud_24.64240128
130   SD_27     24.653794     SoftDecision_FEC27:Baud_24.65379397
131   SD_27     24.688035     SoftDecision_FEC27:Baud_24.68803535
132   SD_27     24.722372     SoftDecision_FEC27:Baud_24.72237198
```

```
133    SD_27      24.733839     SoftDecision_FEC27:Baud_24.73383876
```

Migrating CLI to Terminal-device Configuration

Cisco NCS 1004 supports migration from CLI to OC configuration only, vice-versa is not supported. The transition from CLI to terminal-device must be done via merge-config operation in gRPC.

To migrate from CLI configuration to the terminal-device configuration, perform the following:

Step 1 You must configure a slice using the CLI configuration command. For more details, see Configure the Slice.

Note Do not use all keyword to configure all slices, instead you must configure each slice individually.

Note Configure the trunk port frequencies with 100MHz spacing as after the migration to OC Models only 100MHz spacing is supported.

Note Ignore this step if you are migrating a configured slice. You cannot change the slice configuration while performing migration. For example, if you have configured 100G to 200 G traffic on a slice using CLI, then you can perform OC configuration for the same 100G to 200G slice configuration.

Step 2 Apply OC configuration using Netconf or gRPC. For more details, see Configuring Cisco NCS1004 Using Terminal-device Model, on page 49.

To disable the transition from CLI configuration to the terminal-device configuration, use the following command:

terminal-device transition cli-to-yang disable

OpenConfig Terminal Device Revision

Table 5: Feature History

Feature Name	Release	Description
OC (Open Configuration) Terminal Device Revision	Cisco IOS XR Release 7.3.1	The Open Configuration terminal device revision to 1.7.2 allows you to provide LLDP support on the client optics. This feature allows you to learn LLDP neighbors and the topology of the devices for Operations, Administration, and Maintenance (OAM) purposes.

The client-side LLDP is enabled by default. The LLDP state data is collected over gNMI telemetry.

Limitations:

- There is no support on configuration, since LLDP is enabled by default.
- There is no support for LLDP counters.
- There is no support for leafs age and last update in LLDP neighbor discovery.

Sample gNMI telemetry output for LLDP:

```
{
  "openconfig-terminal-device": {
    "terminal-device": {
      "logical-channels": {
        "channel": {
          "10005": {
            "ethernet": {
              "lldp": {
                "neighbors": {
                  "neighbor": {
                    "nncs5500_node1#HundredGigE0/0/0/30": {
                      "state": {
                        "chassis-id": "008a.96cd.34df",
                        "chassis-id-type": "MAC_ADDRESS",
                        "id": "nncs5500_node1#HundredGigE0/0/0/30",
                        "management-address": "10.127.60.23",
                        "management-address-type": "ipv4",
                        "port-id": "HundredGigE0/0/0/30",
                        "port-id-type": "INTERFACE_NAME",
                        "system-description": " 7.2.1.36I, NCS-5500",
                        "system-name": "nncs5500_node1"
                      }
                    }
                  }
               },
                "state": {
                  "enabled": true,
                  "snooping": true
```

```
                                }
                            }
                        }
                    }
                }
            }
        }
    }
}
```