



Transport Slice Provisioning

This section explains the following topics:

- [Overview, on page 1](#)
- [Scenario: Implement an Any-To-Any L3 eMBB Slice, on page 7](#)
- [Step 1 Create a Slice Template Catalog Entry, on page 10](#)
- [Step 2 Add Service Assurance into the Slice Template Catalog using the NSO CLI \(optional\), on page 12](#)
- [Step 3 Create the Transport Slice Instance, on page 13](#)
- [Step 4 Deploy a Slice using NSO CLI \(optional method\), on page 22](#)
- [Step 5 Visualize and Validate the New Slice Deployment , on page 24](#)
- [Summary and Conclusion, on page 27](#)

Overview

Objective

Simplify Transport Service provisioning by focusing on the service's SLA intents (the "what" instead of the "how"). This implies a service-oriented view, leveraging the concepts of software-defined networking (SDN).

Challenge

Service providers face ever-growing demands from end users for highly customized, flexible network services with very different, sometimes contradictory, service level requirements: support for highly mobile smart cars, ultralow-latency AR and mobile gaming applications, secure machine-to-machine communication in logistics and manufacturing, and so on. Modern software-defined network (SDN) traffic engineering technologists have responded with a host of innovative protocols and features that offer many ways to engineer network traffic to meet these special needs. Crosswork support for these approaches, such as SR-TE services, Tree-SID and Local Congestion Mitigation, are featured elsewhere in this Guide.

The advent of 5G mobile networking has accelerated this trend, resulting in a new approach to network architecture: network slicing. This still-emerging standard enables network engineers to slice the 5G network's bandwidth into tranches that prioritize some services over others, instead of treating it as a single, monolithic network. The network engineer can design each network slice around the needs and intents of its users, allocating speed, latency, throughput and other resources to each slice as required. CNC delivers a rich and customizable tool set to make deploying these slices easier. When combined with Service Health, it provides the added ability to easily monitor the health of these services. The provider organization can then offer the slice itself as a service, helping to broaden the range of service offerings.

But how to make these services easy to provision? The design and coding of the sophisticated traffic engineering services that underlie network slicing require the skills of experienced network architects and deep knowledge of each provider's existing network infrastructure. Without some form of automation that allows line operators to instantiate the designed slices quickly and easily, network slices might remain a type of custom configuration, achievable only for a small set of important users, instead of a scalable commodity providers can monetize.

This is an evolving standard. At present, the Crosswork solution addresses the Transport-level Network Slice Management Functions (NSMF) only.

Solution

Cisco Crosswork Network Controller offers direct support for network slicing at the OSI transport layer. Using this solution, network engineering experts can design slices around customer intents and then add them to a catalog. Network line operators can then simply pick the slice intent that best meets the customer's needs, specify the slice endpoints, and (where needed) set any custom constraints or options built into the chosen slice.

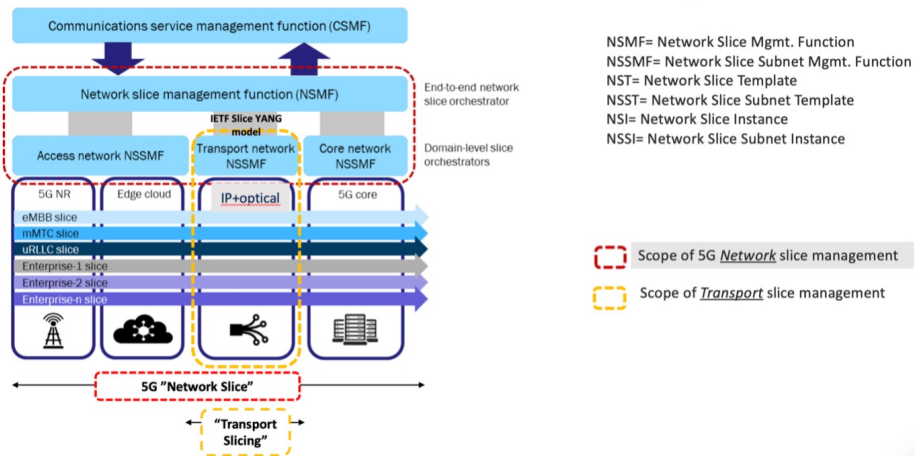
This is Cisco's initial offering in the network slicing arena, chosen because of our company's strengths at the transport layer. At present, the Crosswork solution provides a large catalog of slice template examples and an extensive customization for each template. This document offers a scenario that you can follow to create and (optionally) monitor a network slice.

How Does Transport Slicing Work?

It's important at the outset to understand the difference between 5G network slicing and generalized transport slicing. When operational, a 5G network slice is an end-to-end solution crossing multiple sub-domains. The 5G network authority 3rd Generation Partnership Project (3GPP) refers to each end-to-end network slice operating on the network as the Network Slice Instance (NSI). Each NSI is a unique entity, provisioned in the network with a set of Service Level Requirements chosen from a set of pre-created Network Slice Templates (NST).

All NSIs must be orchestrated by a controller called the Network Slice Management Function (NSFM). The NSMF in turn communicates with sub-domain controllers, referred to as Network Slice Subnet Management Functions (NSSMF). Each NSSMF in turn provisions the corresponding domain-specific slice instance across its own sub-domain boundaries (called a Network Slice Subnet Instance or NSSI). For the Transport domain, the IETF has defined the NSSI as an "IETF Network Slice" in order to differentiate slices in the transport domain from slices bridging other domains. The space occupied by transport slicing in this hierarchy is shown in the illustration below, where the CNC solution will provide the NSSMF functionality for the Transport domain. It is important to highlight that Cisco's Transport Slicing solution can be used independently from 5G use cases, as it's a generic solution for implementing any transport service based on intents.

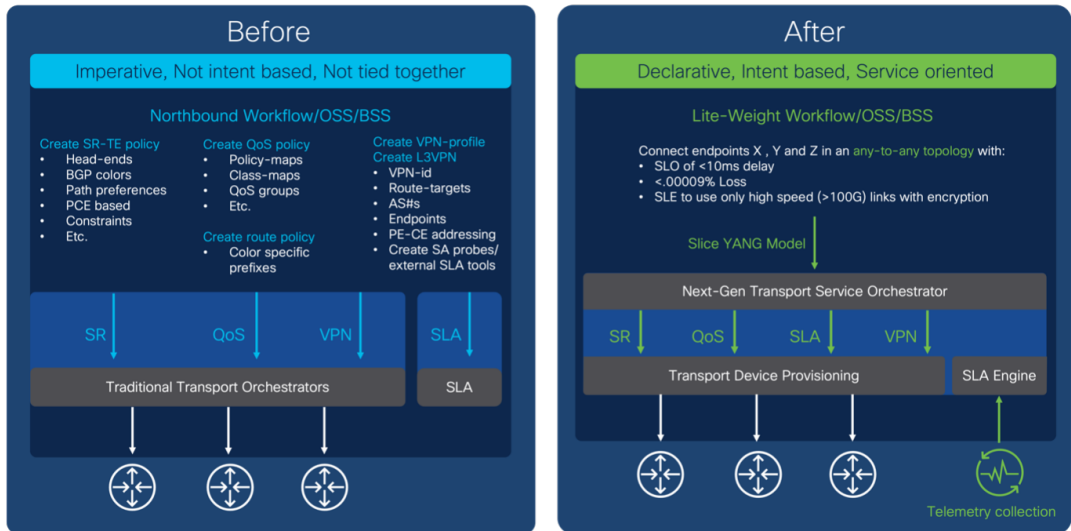
Defining Transport Slicing Scope: 3GPP reference architecture for 5G network slicing



Simplification and ease of use are key principles in transport slicing. The operator wants to start very simply, by requesting from a controller a service based on a desired service intent or outcome (such as supplying low latency to an AR application). He then wants the controller to build the service.

The controller must also monitor the built service to ensure it honors the operator's intent. Above all, the operator wants to avoid exposure to the many configuration parameters required to actually deploy the service at the device layer. Realizing that vision requires the creation of intent-based modularity for value-added transport services supporting the slice, using well-abstracted and declarative service-layer APIs. These service APIs must be maintained and exposed by a controller that can act in a declarative and outcome-based way, as shown in the following figure.

Abstracting the Service Intent



Monitoring the slice's fidelity to the intent involves a Service Level Agreement (SLA) between the customer and the slice provider. To ensure that this SLA both captures the slice intent and has concrete, actionable terms, it can be further defined as either an SLO or SLE:

- **Service Level Objective (SLO):** A desired, achievable target value or range of values for the measurements returned by observation of a Service Level Indicator (SLI). For example, an SLO may be expressed as "SLI <= target", or "lower bound <= SLI <= upper bound".
- **Service Level Expectation (SLE):** The expression of an unmeasurable service-related request that a customer makes of the provider. An SLE is distinct from an SLO because the customer may have little or no way of determining whether the SLE is being met, but will still contract with the provider for a service that meets the expectation (see the following table of sample SLEs).

Table 1: Sample Service Level Expectations

SLE	Description
Encrypted Link Services	Traffic must transit encrypted links only.
Disjoint Path Services	The network has multiple forwarding planes with no common nodes or links.
High speed links only	Traffic must transit high-speed links only. Links offering speeds greater than or equal to 100Gbps are typical for "elephant flows".
Lowest Latency	Always take the lowest latency path. No SLO would be specified in this case.
Regional Avoidance	Do not use nodes or links in specific regions or countries.
Trusted Nodes	Only use trusted nodes ("trusted" meaning verified and not in the common carrier space).
L4-L7 Services	Redirect to "in-line" L4-L7 service on traffic (typically used for security services).
Reliable Links	Use only transit links that have optical protection and L1 diversity.
"Circuit-Style" Services	Provide L1 circuit-like connectivity.
Gaming Services	Use network segments optimized for network gamers (low latency, high bandwidth)
Connected Car	Use network segments optimized for network-connected cars (low latency, close proximity)
Cloud Provider-Specific	Connect me to the secure "walled garden" for a cloud provider (such as AWS or Azure).

The SLA therefore sets key goals and measures to be applied for a given connectivity construct between a sending endpoint and the set of receiving endpoints. It may also describe the extent to which divergence from individual SLOs and SLEs can be tolerated, and specific consequences for violating these SLOs and SLEs.

What Makes Up a Cisco Transport Slice?

To build and deploy these highly abstract intents, Crosswork Network Controller must translate them into actual device configurations. Governing bodies like the IETF and 3GPP leave these decisions to vendors. Cisco can leverage a complete toolkit, built over many years of innovation, as shown in the following figure.

Review: Cisco Toolset for transport level slicing

- QoS and H-QoS: Core and edge
- Forwarding Planes: Shortest Path / SR policies (SRv6 / SR-TE / Flex-algo / Circuit-Style (future))
- SR underlay performance management tools (SR-PM)

Creating and managing the forwarding plane (underlay)

Combining these offer different levels of transport slice separation

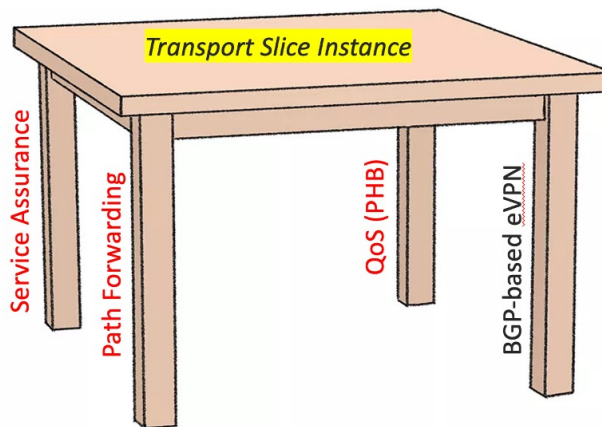
- Virtual Private networks : L2 / L3 VPNs
- ODN and Automated traffic Steering (AS)
- VPN performance management tools (Y1731)

Endpoint selection, Slice isolation and mapping to slice forwarding planes. (overlay)

For a Cisco Transport Slice Instance, we categorize the features in the preceding illustration as follows: Service Assurance, Path Forwarding, QoS (PHB), and BGP-based EVPN. The configurations in these categories are what support the slice instance, as shown in the illustration below.

What is a Cisco Transport Slice Instance?

The four legs of the table that make up a Transport Slice Service Instance



RED= Defined in Slice Catalog (intents)
Black= Defined in Slice Instance (endpoints)

Important: A Transport Slice Instance (or Service) is the combination of all these components.

Scale goals:

Slice "types" defined in catalog = ~10-20?

Slice "instances" (differentiated by VPNs/endpoints) = ~1000s

The first three of these features (shown in red) are defined in the slice template catalog (this catalog is equivalent to what 3GPP calls the NSST). The slice catalog contains slice templates, each of which is defined once by a slice designer. Slice *templates* are just blueprints and are not instantiated in the network in any way. Slice *instances* are the instantiated services after they are deployed in the network. The end-user really doesn't need to know the details of what is inside the templates, just what the overall intent (or SLA) is for each slice template. The slice template catalog is thus a catalog of slice intents.

The fourth category – BGP-based VPNs – that makes up a Cisco Transport Slice Instance is the selection of endpoints and service types (L2 or L3 forwarding). Operators define these when deploying the Transport Slice Instance.

The benefit of this approach is to fully abstract the underlying configuration details of the various machinery components required to realize a Transport Slice Instance (aka the IETF Network Slice, or, in 3GPP parlance, the Network Subnet Slice Instance or NSSI).

To deploy a new slice instance, the operator executes the following steps:

1. Select a Slice Intent from the available Templates in the Slice Catalog.
2. Select slice endpoints and connectivity details, which drive the VPN configuration. Once committed, Crosswork Network Controller will then provision:
 - The forwarding plane policy details which drive the segment routing traffic engineering (SR-TE) configurations and BGP prefix coloring for ODN/AS.
 - The QoS profile details, which drive the ingress marking (for PHB treatment) and the egress scheduling.
 - The SLA details, which will drive the needed Service Assurance configurations.
 - The BGP based VPN connectivity requirements to provide endpoint connectivity.

The following illustration provides more detail on the parts of the slice template that automate slice instantiation.

So what is automated when deploying a Slice Instance?

1. **QoS:** The Slicing CFP can apply input and output QoS policy maps on all slice endpoint interfaces (policy-maps pre-deployed). Both L2 & L3 QoS supported.
2. **Path Forwarding:** The Slicing CFP can deploy SR-TE ODN templates on all headends (metrics= latency, igp, TE, BWoD, FA, etc). Additionally, it will set BGP color community accordingly on all slice advertised prefixes.
3. **Service Assurance:** The Slicing CFP can setup:
 - CNC Heuristic packages for CNC Automated Assurance/Service Health
 - Configure Y1731 probing for P2P L2 slices
 - Configure SR-PM probing for delay and liveness on all slice SR-TE tunnels
4. **Connectivity:** The Slicing CFP will use the L2/L3VPN IETF NM to setup L3 or L2 connectivity automatically across defined slice endpoints. All VPN parameters inferred and abstracted.
 - Setup eVPN VPWS for P2P L2 slices
 - Setup eVPN any-to-any or hub-spoke for L2 multi-point or L3 multipoint slices.
 - Setup up “extranet” connectivity between dedicated and shared slice types. (more on this later).
 - Setup PE-CE eBGP for L3 based slices

© 2022 Cisco and/or its affiliates. All rights reserved. Cisco Confidential

Transport Slice High Level Workflow

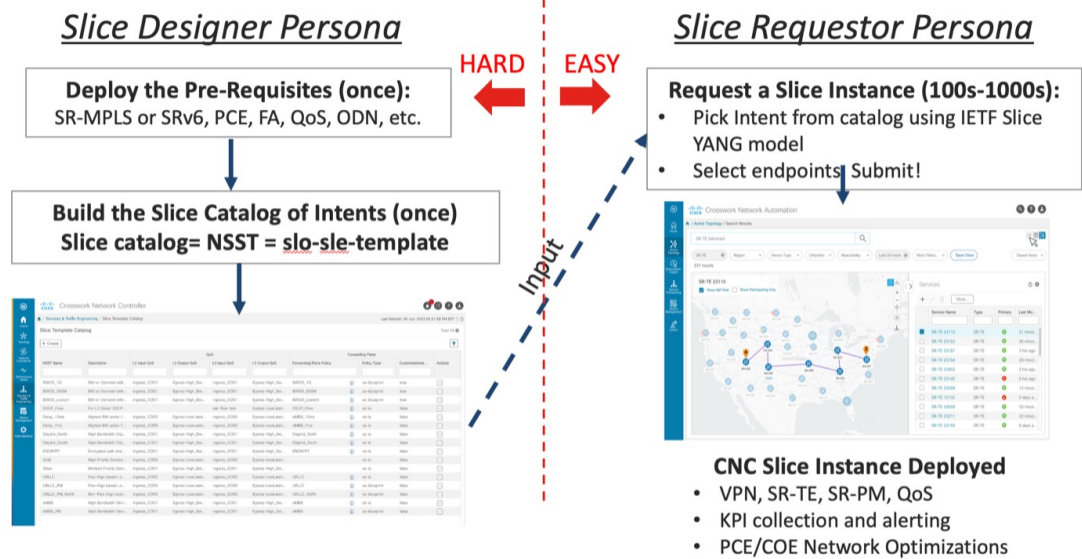
Transport slicing in Crosswork Network Controller is designed around two main user personas:

Slice Designer: The Designer understands the service requirements the provider organization want to offer to customer and is very familiar with the provider network’s underlying capabilities. This person has authorization to do one-time setup operations within the network and has a networking engineering background. They will set up the needed network pre-requisites and then build the slice template catalog, which offers a listing of available slice service offerings for network operators.

Slice Requestor: The Requestor requests new slice instances using the intent-based and simplified CNC user interface. They pick their desired slice type from the pre-built slice catalog, select their endpoints and transport options, and then click submit.

Cisco's objective in the Cisco Crosswork Network Controller Transport Slice solution is to make the user experience as simple as possible for the Requestor. This is the only slice deployment operation driving network service provisioning, and as it must be done constantly for a large SP network, it is a major contributor to provider OPEX. The Slice template catalog creation is done once by highly skilled designer personnel. While the design step is not automated, this approach leverages those skilled resources in a way that maximizes their value to the provider organization at a scale that cannot be realized if the designer must instantiate every slice by hand. The catalog creation requires a good understanding of the network and its capabilities, and requires pre-requisite configurations as shown in the figure below. Slice Designers must be familiar with all the pre-requisite configuration types listed in the illustration for this approach to work.

Slice Automation High Level Workflow



Scenario: Implement an Any-To-Any L3 eMBB Slice

In this scenario, you require a transport slice which has Layer3 any-to-any connectivity across three endpoints, using the intent defined in the catalog for Mobile Broadband (eMBB). The eMBB intent will provide the highest bandwidth available path (including proper QoS marking/scheduling treatment), along with some basic service assurance capabilities such as endpoint interface status and PE-CE route health. The eMBB intent will also enable you to specify:

- The highest bandwidth available path.
- Some basic service assurance capabilities.
- eBGP peer routing connectivity details to CE devices.

Assumptions and Prerequisites

This scenario assumes the network has already built out the required network capabilities for this intent. However, they will be briefly reviewed here for this scenario. For more detailed explanations, see the Cisco Transport Slice Automation Design Guide.

Slice Service Package Prerequisites

There are a few optional prerequisites used by the NSO slice services package that need to be bootstrapped into NSO. The need for these prerequisites are dependent on the types of slices and intents required.

First, if you plan on using any “as-blueprint” forwarding-plane-policy-types in your template catalog, then you need to create an NSO sr-color resource pool so that the slicing service can assign colors for dynamically create ODN policies. This pool should then be referenced by the slicing service.

Second, if creating point-to-point L2 slice service types, the route-policy map assigned to the BGP session for the route reflector is required to be identified to the slicing package. This policy map will be modified by the slicing package with new policies as needed for L2 services, which is a standard approach for VPWS.

In this scenario, these items are not required since you are using neither of these functions:

```
resource-pools id-pool sr-color-pool
range start 1000
range end 2000
!
network-slice-services cfp-configurations color-pool-name sr-color-pool
network-slice-services global-settings parent-rr-route-policy SET_COLOR_EVPN_VPWS
!
```

Path Forwarding Prerequisites

The following settings have been preconfigured with the NSO T-SDN SR-TE CFP for the eMBB ODN path-forwarding intent with these properties:

- Use Color 100 to identify the intent.
- PCE is responsible for dynamic path computation
- The dynamic path computation will be based on the IGP metric.

On NSO, this set of properties will look like the below example. At this stage, the ODN policy has not been pushed to the devices.

```
admin@ncs# show running-config cisco-sr-te-cfp:sr-te odn odn-template eMBB
cisco-sr-te-cfp:sr-te odn odn-templatee eMBB
color 100
dynamic pce
dynamic metric-type igp
!
```

QoS Prerequisites

As described, you (the Slice Designer) should have a good understanding of the network’s settings and device capabilities. You should have a well designed and implemented QoS design throughout your network. In the case of QoS treatment for the high BW business services and for example illustration purposes only, you have chosen to deploy these services with the network’s existing “Class of Service 1” traffic policy (called “ingress_COS1”).

The details of this policy are again provider specific, but in this example the policy will not examine or modify the ingress traffic’s IP DSCP setting, but simply mark all the traffic with an MPLS experimental bit (EXP) of 1 so that downstream core scheduling can provide the proper BW treatment. On egress from the provider

network, you have chosen a policy called “Egress-High_BW-Apps” which will assign 50% of the bandwidth to Class of Service 1 (COS1) marked traffic.

It is assumed these QoS policies are already deployed on all edge PE devices (but not yet on the customer facing interfaces, but have been built out and ready for use). Yet, you still need to identify that these QoS policies are available to the Slice Template catalog for use for Slice services. You will need to provide that mapping and will need to identify which policies are available for either Layer 2 or Layer 3 slice services, or both. Since QoS policies can be tailored specifically to Layer 3 or Layer 2 traffic (for example matching on L3 DSCP vs L2 ToS bits) the system allows you to specify the usage. In the case of the example above, since all traffic is being marked with EXP=1 regardless of DSCP or ToS, these policies are applicable to both L2 or L3 services.

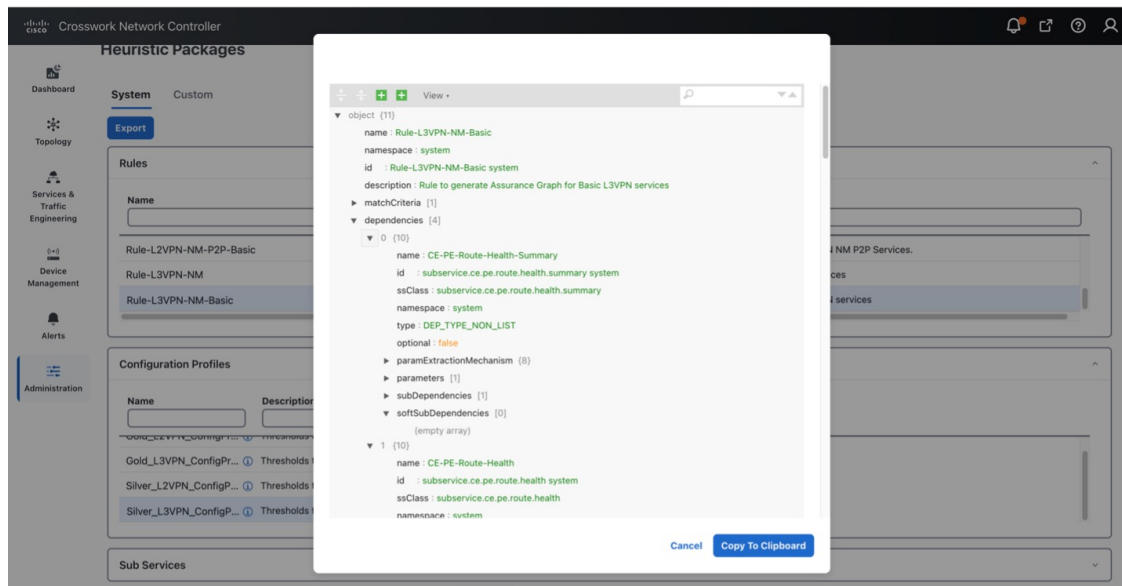
```
admin@ncs# show running-config network-slice-services slq-sle-templates qos-catalog
network-slice-services slq-sle-templates qos-catalog L2 output-qos-policy Egress-High_Bw_Apps
description "High BW egress"
!
network-slice-services slq-sle-templates qos-catalog L2 input-qos-policy ingress_COS1
description "Treat all as Business Data"
!
network-slice-services slq-sle-templates qos-catalog L3 output-qos-policy Egress-High_Bw_Apps
description "High BW egress"
!
network-slice-services slq-sle-templates qos-catalog L3 input-qos-policy ingress_COS1
description "Treat all as Business Data"
!
```

Slice Service Assurance Settings

In this scenario you only have basic service assurance requirements which are based on passive state monitoring (no active probing). You will be using Crosswork Network Controller's Service Health capability and the Crosswork Network Controller system’s pre-built heuristic packages (ConfigProfiles and Rules) which define the objects to be monitored. In the scenario you want to monitor basic device health and PE-CE route health which are included in the basic system package. When you build the slice catalog, you can define which packages to use for L2 point-to-point, L2 multipoint and/or L3 services.

The scenario above is requiring L3 services, but when you create your catalog for the eMBB intent (next step), you also need to consider future slices instances for eMBB services that are L2 service types, thus you can include all these pre-built system heuristic packages in the catalog entry for eMBB. Since these are all pre-built system packages, no prerequisite configurations are required. The system heuristic packages can be viewed via the CNC UI by selecting **Administration > Heuristic Packages** (see below figure).

System Heuristic Profile and Rule names used for eMBB	Usage
Silver_L3PN_ConfigProfile system	L3 profile-name
Rule-L3VPN-NM-Basic system	L3 rule-name
Silver_L2PN_ConfigProfile system	L2 multipoint profile-name
Rule-L2VPN-MP-Basic system	L2 multipoint rule-name
Silver_L2PN_ConfigProfile system	L2 point-to-point profile-name
Rule-L2VPN-NM-Basic system	L2 point-to-point rule-name



Step 1 Create a Slice Template Catalog Entry

You will now build out the slice catalog entry for eMBB intent-based slice services. This operation is done once and will use the above components as input and can cover both L2 and L3 slice instance requests. Once complete, you can move to deploying the slice service instance for this scenario and this entry will now be available for future slice instances requiring eMBB intent services (for example, all of the above prerequisite steps will not be required).

This step, performed by the Slice Designer, builds a slice catalog of intents, or slice types, that will be referred to when creating the actual slice instance. This catalog (along with the slice instances themselves) can be built in multiple ways:

- Using the Crosswork Network Controller UI
- Using the Crosswork Network Controller or NSO Slicing API
- Using the NSO CLI, including load merge from a text file

This scenario has a Service Assurance option that can only be created using the NSO CLI or the Crosswork Network Controller API. It will be shown in the section "Add Service Assurance into the Slice Template Catalog using the NSO CLI".

```
admin@ncs# show running-config network-slice-services slo-sle-templates slo-sle-template eMBB
network-slice-services slo-sle-templates slo-sle-template eMBB
  template-description "High Bandwidth Service with basic SLA monitoring"
  qos-policy L2 input-policy ingress_COS1
  qos-policy L2 output-policy Egress-High_Bw_Apps
  qos-policy L3 input-policy ingress_COS1
  qos-policy L3 output-policy Egress-High_Bw_Apps
  odn forwarding-plane-policy eMBB
  odn forwarding-plane-policy-type as-is
  service-assurance heuristics monitoring-state enable
  service-assurance heuristics L2 point-to-point profile-name "Silver_L2VPN_ConfigProfile system"
  service-assurance heuristics L2 point-to-point rule-name "Rule-L2VPN-NM-Basic system"
  service-assurance heuristics L2 multipoint profile-name "Silver_L2VPN_ConfigProfile system"
  service-assurance heuristics L2 multipoint rule-name "Rule-L2VPN-MP-Basic system"
  service-assurance heuristics L3 profile-name "Silver_L3VPN_ConfigProfile system"
  service-assurance heuristics L3 rule-name "Rule-L3VPN-NM-Basic system"
!
```

To create a slice template catalog entry using the Crosswork Network Controller UI, do the following:

Step 1 Go to **Services & Traffic Engineering > Slice Template Catalog**.

The Slice Template Catalog screen appears.

Note For the purpose of this scenario, the templates that appear in the image below have already been created on NSO.

NSST Name	Description	QoS				Forwarding Plane			Actions
		L2 Input QoS	L2 Output QoS	L3 Input QoS	L3 Output QoS	Forwarding Plane Policy	Policy Type	Customizati...	
BWOD_1G	BW on Demand wit...	ingress_COS1	Egress-High_Bw...	ingress_COS1	Egress-High_Bw...	BWOD_1G	as-blueprint	true	...
BWOD_500M	BW on Demand wit...	ingress_COS1	Egress-High_Bw...	ingress_COS1	Egress-High_Bw...	BWOD_500M	as-blueprint	true	...
BWOD_custom	BW on Demand wit...	ingress_COS1	Egress-High_Bw...	ingress_COS1	Egress-High_Bw...	BWOD_custom	as-blueprint	true	...
DSCP_Flow	For L3 Slices: DSC...			per-flow-test	Egress-LowLate...	DSCP_Flow	as-is	false	...
Delay_10ms	Highest BW under ...	ingress_COS5	Egress-LowLate...	ingress_COS5	Egress-LowLate...	eMBB_10ms	as-is	false	...
Delay_7ms	Highest BW under ...	ingress_COS5	Egress-LowLate...	ingress_COS5	Egress-LowLate...	eMBB_7ms	as-is	false	...
Disjoint_North	High Bandwidth Di...	ingress_COS1	Egress-High_Bw...	ingress_COS1	Egress-High_Bw...	Disjoint_North	as-is	false	...
Disjoint_South	High Bandwidth Di...	ingress_COS1	Egress-High_Bw...	ingress_COS1	Egress-High_Bw...	Disjoint_South	as-is	false	...
ENCRYPT	Encrypted path onl...	ingress_COS1	Egress-High_Bw...	ingress_COS1	Egress-High_Bw...	ENCRYPT	as-is	false	...
Gold	High Priority Servi...	ingress_COS5	Egress-LowLate...	ingress_COS5	Egress-LowLate...			false	...
Silver	Medium Priority Se...	ingress_COS1	Egress-High_Bw...	ingress_COS1	Egress-High_Bw...			false	...
URLLC	Flex-Algo based Lo...	ingress_COS5	Egress-LowLate...	ingress_COS5	Egress-LowLate...	URLLC	as-is	false	...
URLLC_PM	Flex-Algo based Lo...	ingress_COS5	Egress-LowLate...	ingress_COS5	Egress-LowLate...	URLLC	as-blueprint	false	...

Step 2 Click **+ Create** to create a new slice catalog entry. The New Slice Template screen appears.

Step 3 For Network Subnet Slice Template (NSST), type the new slice template name: **eMBB**. In addition, in the Description field, type a short description of the slice template’s intent: **High Bandwidth Service**.

Step 4 Assign the QoS ingress and egress policies. This depends on the slice instance Service Type (L2 or L3 policy) you will define later when creating a new slice instance.

Note For the purpose of this scenario, the five fields at the bottom of the screen (L2 Input QoS, L2 Output QoS, L3 Input QoS, L3 Output QoS, Forwarding Plane Policy Template) have already been provisioned in NSO and automatically appear as an option in each list.

Note You may also refer to the table built earlier and found under QoS in the prerequisites.

Step 2 Add Service Assurance into the Slice Template Catalog using the NSO CLI (optional)

- For L2 Input QoS, select **ingress_COS1**
- For L2 Output QoS, select **Egress-High_Bw_Apps**
- For L3 Input QoS, select **ingress_COS1**
- For L3 Output QoS, select **Egress-High_Bw_Apps**

Step 5 For Forwarding Plane Policy Template, select the ODN template policy created earlier (see prerequisites and assumptions section) that complements the forwarding plane intent. For the purpose of this scenario, select **eMBB**.

Step 6 For Policy Type, determine if this template is to be used **as-is** or **as-blueprint**. For the purpose of this scenario, select **as-is**.

The **as-is** forwarding templates increase overall scalability as the SR-TE tunnels can be shared across multiple slice templates and instances. However, these ODN templates will not be dynamically modified by the slice package with additional functionality, including dynamic support for Performance Measurement or BWoD reservations.

Note When **as-blueprint** is selected, determine if you also want to allow further per-slice instance customizations. These settings determine the SR-TE infrastructure re-use and the scale. Once **as-blueprint** is selected for Policy Type, the **Allow Customizations** check box becomes available.

The screenshot shows the 'New Slice Template' configuration page in the Crosswork Network Controller. The form is titled 'New Slice Template' and is part of the 'Slice Template Catalog'. The left sidebar contains navigation options: Dashboard, Topology, Services & Traffic Engineering (selected), Device Management, Alerts, and Administration. The main form area contains the following fields:

- NSST**: eMBB
- Description**: High Bandwidth Service
- L2 Input QoS**: ingress_COS1
- L2 Output QoS**: Egress-High_Bw_Apps
- L3 Input QoS**: ingress_COS1
- L3 Output QoS**: Egress-High_Bw_Apps
- Forwarding Plane Policy Template**: eMBB
- Policy Type**: as-is
- Allow Customizations

Annotations on the right side of the form provide additional context:

- Enter string data: A unique transport Network Slice Template Name (NSST) and Description
- Enter Desired QoS polices, depending on Slice Instance Service Type the slicing package will select the proper L2 or L3 policy.
- Forwarding Plane Intent, select the desired Forwarding Plane Policy Template (from pre-created ODN templates). Determine if this forwarding template should be used 'as-is' or 'as-blueprint'. If 'as-blueprint' determine if you would like to allow further per-slice instance customizations (i.e., BWoD). These settings will determine the SR-TE infra re-use and ultimately the scale.

Step 7 Click Save.

Step 2 Add Service Assurance into the Slice Template Catalog using the NSO CLI (optional)

If Service Assurance is required for the Slice instance, then the Slice Designer can add the necessary Service Assurance functionality into the Slice Template using NSO CLI or API. Below is a template in NSO CLI with Service Assurance parameters to be used for this scenario. It can be added directly into NSO CLI or with the API.

There are three Service Assurance sections to the template settings:

- Step 1** Reference pointers to Crosswork Network Controller Service Health Heuristic packages to be used and monitoring state. This monitoring state cannot be changed at the Slice instance level at this time, it is set universally for all slice instances referencing this Slice template. Since different connectivity-types (pt-2-pt or multi-point) can be selected when provisioning a slice instance and different service types (L2 or L3), multiple heuristic package options are available, and the system will select the proper package depending on the slice instance requirements.
- Step 2** If the Slice Instance is a L2 service type with pt-2-pt connectivity, then Y1731 probe monitoring can be enabled. The settings required are shown in the below example. Slice SLA alarming and alerting can be configured if the proper settings are selected in the L2 Heuristic package for Service Health.
- Step 3** This scenario does not require Performance Measurement, so it is not included in the below template. But if desired, it can be enabled in the template and SR-PM will be dynamically configured on the SR-TE tunnel if the Slice Forwarding policy-type is set for **as-blueprint**. Slice SLA alarming and alerting can be configured if the proper settings are selected in the Heuristic package for Service Health.

```
admin@ncs# show running-config network-slice-services s1o-s1e-templates s1o-s1e-template eMBB
network-slice-services s1o-s1e-templates s1o-s1e-template eMBB
template-description "High Bandwidth Service with basic SLA monitoring"
qos-policy L2 input-policy Ingress_COS1
qos-policy L2 output-policy Egress-High_Bw_Apps
qos-policy L3 input-policy Ingress_COS1
qos-policy L3 output-policy Egress-High_Bw_Apps
odn forwarding-plane-policy eMBB
odn forwarding-plane-policy-type as-is
service-assurance heuristics monitoring-state enable
service-assurance heuristics L2 point-to-point profile-name "Silver_L2VPN_ConfigProfile system"
service-assurance heuristics L2 point-to-point rule-name "Rule-L2VPN-NM-Basic system"
service-assurance heuristics L2 multipoint profile-name "Silver_L2VPN_ConfigProfile system"
service-assurance heuristics L2 multipoint rule-name "Rule-L2VPN-MP-Basic system"
service-assurance heuristics L3 profile-name "Silver_L3VPN_ConfigProfile system"
service-assurance heuristics L3 rule-name "Rule-L3VPN-NM-Basic system"
service-assurance ethernet-service-oam md-name foo
service-assurance ethernet-service-oam md-level 4
service-assurance ethernet-service-oam y-1731 profile-delay Profile-Delay-1
!
```

See previous explanations from UI figure

CNC Heuristic packages to be used for Service health. Slicing package will pick proper package depending on Slice instance service type (L2 or L3) and L2 connectivity model (point-to-point or multi-point). User can also associate custom packages.

Y1731 probing specifications: For L2 point-to-point slice service types only. Ignored for other slice types

As previously highlighted, Slice Templates with Service Assurance parameters can only created using the NSO CLI or Crosswork Network Controller/NSO API at this time. This also means that these additional parameters will not be visible when viewing the Slice Template in the Crosswork Network Controller UI.

Step 3 Create the Transport Slice Instance

Once the slice type catalog has been created, we can now deploy the transport slice instance. The below table outlines the user data required to deploy this slice. The mandatory data consists of a series of string-data names (user defined), selection of the service type (L2 or L3), catalog intent selection and then defining the Service Demarcation Points (SDPs) which are the PE endpoints facing the customer. These PE endpoints will require IP information since this is a L3 slice and optionally since eBGP was desired for the PE-CE peering protocol, the CE eBGP information is required.

For the purpose of this scenario, use the sample data below:

Table 2: Required Parameter Values:

Parameter	User Value	Mandatory	Notes
slice-service-name	a_L3_A2A_ded	Y	String. Maximum 17 characters. Must be unique.

Parameter	User Value	Mandatory	Notes
description	“any string data”	N	Any string
customer	ACME	N	String meta data- user defined
service-tag	L3	Y	L2 or L3 forwarding
nssai	123459876	N	String meta data- could match 5G nssai assignment if provider desires
slo-sle-template	eMBB	Y	Selection from pre-built slice catalog
isolation	dedicated	N	The default is dedicated- the other option is shared
First SDP endpoint name	1	Y	String- unique within slice instance- At least one SDP must be created, the rest optional
Node-Name	Node-4	Y	PE Node-Name as defined in CNC topology
Attachment-circuit name	ac1	Y	String- unique within slice instance
Interface-ID	TenGigE0/0/0/10	Y	Customer facing PE Interface
VLAN ID	401	N	VLAN ID if using vlan sub-interfaces
Interface IP	172.16.2.1	Y	PE Interface IP address (since L3 service)
Interface IP Mask	29	Y	Interface prefix length (i.e. /29)
Peering protocol	BGP	N	PE-CE peering protocol (BGP or none)
BGP Neighbor ASN	65102	Y	Since bgp selected, peer ASN
BGP Neighbor Address	172.16.2.2	Y	Since bgp was selected, peer IP address

Parameter	User Value	Mandatory	Notes
Second SDP endpoint name	2	N	Additional SDPs are optional, but in this scenario we have three endpoints
Node-Name	Node-5		PE Node-Name as defined in CNC topology
Attachment-circuit name	Ac2		String- unique within slice instance
Interface-ID	TenGigE0/0/0/2		Customer facing PE Interface
VLAN ID	301		VLAN ID if using vlan sub-interfaces
Interface IP	172.16.1.1		PE Interface IP address (since L3 service)
Interface prefix length	29		Interface prefix length (i.e. /29)
Peering protocol	bgp		PE-CE peering protocol (bgp or none)
BGP Neighbor Address	172.16.1.2		Since bgp was selected, peer IP address
BGP Neighbor ASN	65101		Since bgp selected, peer ASN
Third SDP endpoint name	3	N	Additional SDPs are optional, but in this scenario, we have three endpoints
Node-Name	Node-2		PE Node-Name as defined in CNC topology
Attachment-circuit name	Ac3		String- unique within slice instance
Interface-ID	TenGigE0/0/0/2		Customer facing PE Interface
VLAN ID	601		VLAN ID if using vlan sub-interfaces
Interface IP	172.16.3.1		PE Interface IP address (since L3 service)

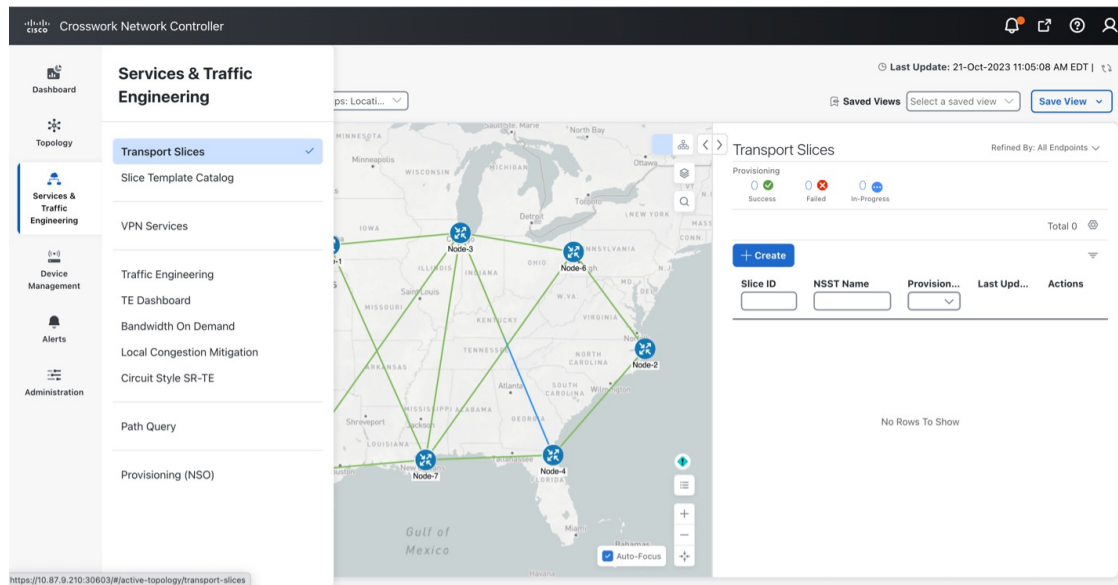
Step 3 Create the Transport Slice Instance

Parameter	User Value	Mandatory	Notes
Interface prefix length	29		Interface prefix length (i.e. /29)
Peering protocol	bgp		PE-CE peering protocol (bgp or none)
BGP Neighbor Address	172.16.3.2		Since bgp was selected, peer IP address
BGP Neighbor ASN	65103		Since bgp selected, peer ASN

The Slice Instance may be created using the Crosswork Network Controller UI, Crosswork Network Controller /NSO API, or NSO CLI. The below example demonstrates the UI steps and explains various fields (both required and optional).

Step 1 Go to **Services & Traffic Engineering > Transport Slices**.

The Transport Slices panel appears.



Step 2 Click **+ Create** to create a new slice.

The New Slice panel appears. At the top, four steps are displayed that tracks the creation of a new slice. The first step requires Basic Details of the new slice.



Step 3 Type the string data into the Slice ID, Customer, and Description fields. For example"

- Slice ID: **a_L3_A2A_ded**
- Customer: **ACME**
- Description: **L3 any-2-any dedicated slice**

Step 4 Select the Service Type: either Layer 2 (**L2**) or Layer 3 (**L3**) connectivity services. In this instance, select **L3**.

Step 5 Optionally, add a string-based Single-Network Slice Selection Assistance Information (S-NSSAI) for 5G mobility customers. This mobility slice-ID information is only used as meta-data by the orchestration system. For example, type **123459876**.

Step 6 Click **Next**.

The screenshot shows the 'New Slice' configuration interface. At the top, there is a progress bar with four steps: 'Basic Details' (selected), 'NSST', 'Connectivity', and 'SDP'. Below the progress bar, the 'Basic Details' section contains the following fields:

- Slice ID *** (Required Field): a_L3_A2A_ded
- Customer**: ACME
- Description**: L3 any-2-any dedicated slice
- Service Type**: L2 (unselected), L3 (selected)
- S-NSSAI**: 123459876

At the bottom of the form, there are two buttons: 'Cancel' and 'Next'.

Step 7 The second step requires Network Subnet Slice Template (NSST) details.

This information specifies which Slice Template to use from the pre-created Template catalog. So to match the 3GPP 5G naming convention, it is named the Network Slice Subnet Template (NSST). The description on these templates describe the intent specified by the Slice Designer. Depending on the Slice Service Type selected in the previous step (which was **L3**), the system pulls the appropriate L3 (or L2, if specified) based functionality referenced in the template (for example, QoS settings).

Step 8 Select the desired intent from the pre-created Slice Catalog: **eMBB**.

Step 3 Create the Transport Slice Instance

New Slice * Required Field

Basic Details **NSST** Connectivity SDP

Network Subnet Slice Template (NSST) * ⓘ

<input type="radio"/> BWOD_500M	L3 Input QoS ingress_COS1
<input type="radio"/> BWOD_custom	L3 Output QoS Egress-High_Bw_Apps
<input type="radio"/> DSCP_Flow	Forwarding Plane Policy Template eMBB
<input type="radio"/> Delay_10ms	Policy Type as-is
<input type="radio"/> Delay_7ms	Customization false
<input type="radio"/> Disjoint_North	
<input type="radio"/> Disjoint_South	
<input type="radio"/> ENCRYPT	
<input type="radio"/> Gold	
<input type="radio"/> Silver	
<input type="radio"/> URLLC	
<input type="radio"/> URLLC_PM	
<input type="radio"/> URLLC_PM_NoFA	
<input checked="" type="radio"/> eMBB	

Cancel Previous Next

Note The Slice Catalog names, descriptions, and parameters are set by Slice Designer during the catalog creation phase

Step 9 Click **Next**.

Step 10 The third step requires Connectivity details.

This information builds the connectivity details for the slice by defining if the slice is dedicated or shared. If this is a dedicated slice, it can optionally connect to pre-created shared slices (if it is not a L2 P2P slice). Single Sided Control will allow for uniform bi-directional policies when connecting to the shared slice (i.e., the dedicated slice policies are used when connecting to shared slice endpoints).

Step 11 For Connectivity Group, the field will automatically show Default and cannot be changed.

Note The IETF Slice YANG model has the concept of Connectivity Groups with the idea that multiple Connectivity Groups can be built under a single Slice ID. Currently, only one Connectivity Group is supported.

Step 12 Determine the slice Isolation behavior by either selecting **Dedicated** or **Shared** for Isolation. In this instance, select **Dedicated**.

Note Unique to Crosswork Network Controller, **Dedicated** slices can connect to shared slices (for example, providing a VPN extranet connectivity model).

Note If **Shared** is selected, the remaining selections in the Connectivity step default to system details (for example, the Connectivity Type is set to **Any To Any**).

Step 13 For Connectivity Type, select **Any To Any**.

When selecting the connectivity requirements, choosing L2 or L3 services will determine the available Connectivity Type options.

- L3 Services: **Any To Any, Hub and Spoke.**
- L2 Services: **Any To Any, Hub and Spoke, Point To Point.**

Note If you select **Hub and Spoke**, the endpoint role is selected in a later step.

Step 14 In this instance, skip both Connectivity Shared Slices and Bandwidth Reservation.

Note Connectivity Shared Slices – If **Dedicated** was selected, the option to connect to an existing shared slice instance becomes available.

Note Bandwidth Reservation – If **Allow Customizations** is selected during the catalog entry (and the Policy Type selected is **as-blueprint**), thus having a customizable NSST, you can select Bandwidth Reservation per slice instance or enter a different value.

Step 15 For Single Sided Control, leave as the default, **True**.

Note If **True** is selected, it will force a dedicated slice path forwarding behavior towards shared slice endpoints (overriding shared slice path forwarding intent and will ensure the same bi-directional path forwarding).

Step 16 Select **Show advanced settings** to edit optional parameters.

Advanced settings are only available for non-L2 P2P based slices. This will allow for custom Route Target (RT) and Route Distinguisher (RD) settings. By default, these are set to auto and thus not required to configure. If connectivity type is any-to-any and if manual RT is selected then a box allowing for manual entry of the RT is presented, with this value being used uniformly across all sites to import/export. If the connectivity type is hub-spoke, then there will be two RTs required, one for hub and one for spoke.

Step 17 For Route Target Type, select **Auto**.

Step 18 For Route Distinguisher Type, select **Auto**.

Step 19 Click **Next**.

New Slice * Required Field

Progress: Basic Details | NSST | **Connectivity** | SDP

Connectivity Group ⓘ
Default

Isolation ⓘ
 Dedicated Shared

Connectivity Type ⓘ
Any To Any

Connectivity Shared Slices ⓘ
Select One or More

Single Sided Control ⓘ
 True False

Bandwidth Reservation ⓘ : None Selected

1 G 5 G 10 G 50 G 100 G

OR
Enter a value Gbps

[Hide advanced settings](#)

Route Target Type ⓘ
 Auto Manual

Route Distinguisher Type ⓘ
 Auto Manual

Step 20 The fourth step requires Slice Demarcation Point (SDP) details.

Here, the Slice Requester provides PE endpoint interface details. These endpoints are called SDPs per IETF Slicing standards.

Step 21 Enter the SDP ID and Attachment Circuit ID to configure the string data. Both entries must be unique within the slice service instance.

- For SDP ID, type **1**.
- For Attachment Circuit ID, type **1**.

Step 22 In Node ID, select a node from the list: **Node-4 [192.168.255.20]**.

This Node ID uniquely identifies an edge node of the SDP.

Step 23 Select the Interface Type, **TenGigE**, and type the Interface ID, **0/0/0/10**.

Step 24 For VLAN ID (optional), type **401**.

Step 25 For Interface IP, type **172.16..2.1** and type **29** for the mask. This defines the IP address of the attachment circuit.

Step 26 Since this is an L3 slice service, the following field is required. For Peering Protocol, select **BGP** as the SDP peering protocol to CE.

Step 27 Since the Peering Protocol was defined as BGP, the following fields are required.

- a. For Remote-AS, type **65102**.
- b. For Neighbor Address, type **172.16.2.2**.

Step 28

Click **+ Add Another** to add a second (Node-5) and third (Node-2) SDP endpoints. See the Slice Instance Required Data table for parameter values.

Slice Demarcation Point

Node ID	SDP ID	AC ID
Node-4	1	1

SDP ID: 1

Attachment Circuit ID: 1

Node ID: Node-4 [192.168.255.20]

Interface Type: TenGigE

Interface ID: 0/0/0/10

VLAN ID: 401

Interface IP: 172.16.2.1 / 29 (Range: 1 to 128)

Peering Protocol: BGP

Remote-AS: 65102

Neighbour Address: 172.16.2.2

[Show advanced settings](#)

[+Add Another](#)

[Cancel](#) [Previous](#) [Commit Changes](#)

Step 29

Click **Commit Changes**.

The new slice service is deployed.

Step 4 Deploy a Slice using NSO CLI (optional method)

The option to deploy a slice using the NSO CLI is also available. The below payload shows the details of deploying a slice using load merge when using the NSO CLI. The defaults are not displayed.


```

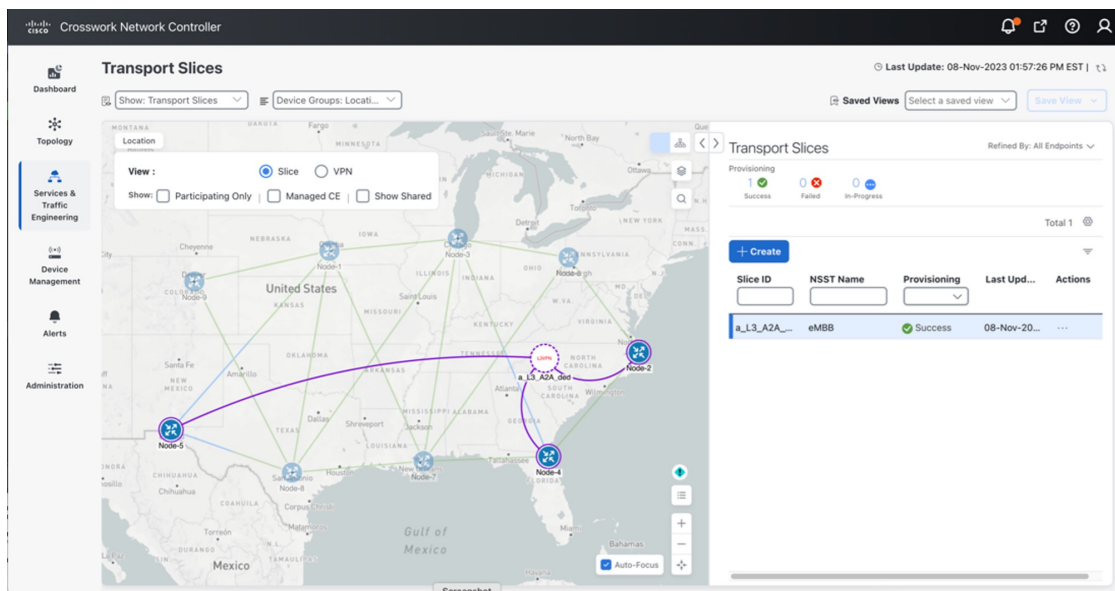
network-slice-services slice-service a_L3_A2A_ded
service-description "L3 any-2-any dedicated slice- a_L3_A2A_dedicated_eMBB.cli"
service-tags tag-type service-tag-customer
  value [ ACME_1 ]
!
service-tags tag-type service-tag-service
  value [ L3_1 ]
!
service-tags tag-opaque DSSAI
  value [ 123459876_1 ]
!
slo-sle-template eMBB
sdp sdp 1
  node-id Node-4
  service-match-criteria match-criterion 1
    target-connection-group-id group1
  !
  attachment-circuits attachment-circuit ac1
    ac-tp-id TenGigE0/0/0/10
    ac-ip-address 172.16.2.1
    ac-ip-prefix-length 29
    ac-tags ac-tags attachment-circuit-tag-vlan-id
      value [ 401_1 ]
    !
    sdp-peering protocol peering-protocol-bgp
      bgp-attributes neighbor [ 172.16.2.2_1 ]
      bgp-attributes remote-as 65102
    !
  !
sdp sdp 2
  node-id Node-5
  service-match-criteria match-criterion 1
    target-connection-group-id group1
  !
  attachment-circuits attachment-circuit ac2
    ac-tp-id TenGigE0/0/0/2
    ac-ip-address 172.16.1.1
    ac-ip-prefix-length 29
    ac-tags ac-tags attachment-circuit-tag-vlan-id
      value [ 301_1 ]
    !
    sdp-peering protocol peering-protocol-bgp
      bgp-attributes neighbor [ 172.16.1.2_1 ]
      bgp-attributes remote-as 65101
    !
  !
sdp sdp 3
  node-id Node-2
  service-match-criteria match-criterion 1
    target-connection-group-id group1
  !
  attachment-circuits attachment-circuit ac3
    ac-tp-id TenGigE0/0/0/2
    ac-ip-address 172.16.3.1
    ac-ip-prefix-length 29
    ac-tags ac-tags attachment-circuit-tag-vlan-id
      value [ 601_1 ]
    !
    sdp-peering protocol peering-protocol-bgp
      bgp-attributes neighbor [ 172.16.3.2_1 ]
      bgp-attributes remote-as 65103
    !
  !
!
connection-groups connection-group group1
connectivity-type any-to-any
!
!

```

Step 5 Visualize and Validate the New Slice Deployment

Step 1 Go to **Services & Traffic Engineering > Transport Slices**.

The Transport Slices panel appears with the new slice displayed. The Provisioning state should show as **Success**.



Step 2 Optionally, the slice service state can be verified, from the NSO CLI, that all stages were successfully provisioned with all plan states **reached**.


```
admin@ncs# show network-slice-services slice-service-plan_a_L3_A2A_ded
```

TYPE	NAME	BACK TRACK	GOAL	STATUS CODE	NODE	STATE	STATUS	WHEN	POST ACTION ref	STATUS
self	self	false	-	-	-	init ready	reached	2023-10-21T16:06:21	-	-
sdr	1	false	-	-	Node-4	init ready	reached	2023-10-21T20:51:31	-	-
sdr	2	false	-	-	Node-5	init ready	reached	2023-10-21T16:06:21	-	-
sdr	3	false	-	-	Node-2	init ready	reached	2023-10-21T20:51:31	-	-
sdr	3	false	-	-	Node-2	init ready	reached	2023-10-21T16:06:21	-	-
sdr	3	false	-	-	Node-2	init ready	reached	2023-10-21T16:06:21	-	-
sdr	3	false	-	-	Node-2	init ready	reached	2023-10-21T20:51:31	-	-

```

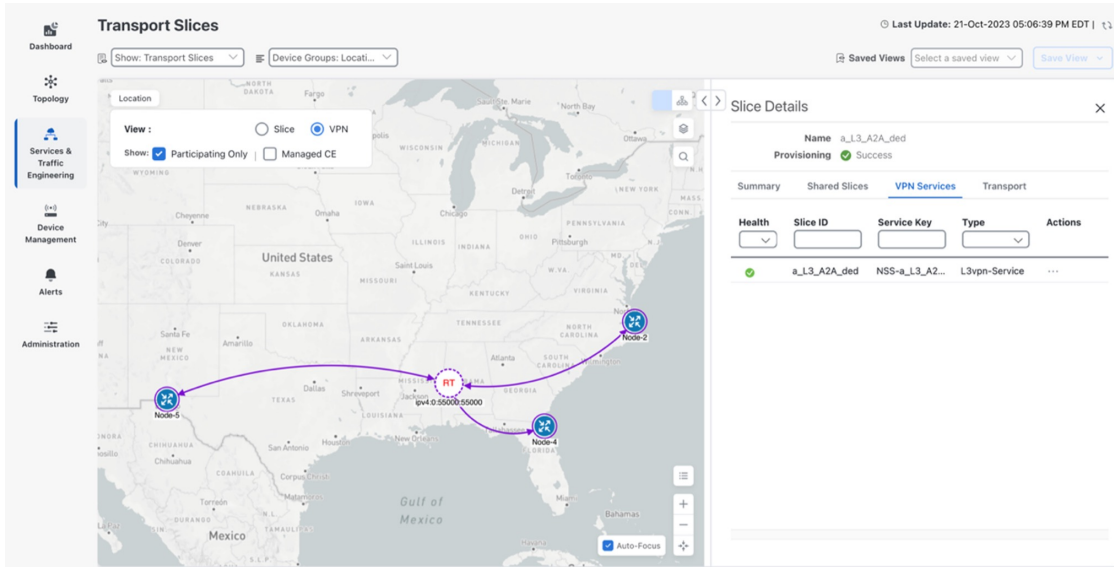
plan status color-allocation-data color 100
plan status service-tag-service [ L3 ]
plan status forwarding-plane-policy AMBB
plan status rt-allocation-data hub-rt 0:55000:55000

```

Step 3 From the Transport Slices screen in the UI, click  in the Actions column for the newly created slice, **a_L3_A2A_ded**, and select **View Details**.

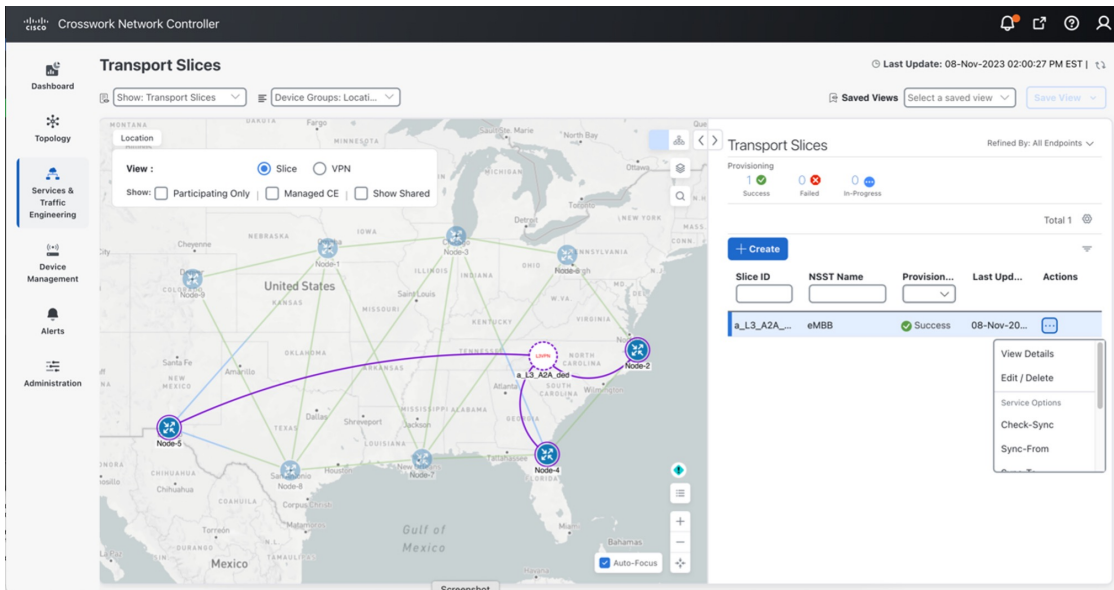
The Slice Details panel appears while the topology map updates to show the new slice.

Step 4 In the Slice Details panel, select the **VPN Services** tab, and in the topology map select **VPN** as the View. The information updates so you can see that the slice provisioned with auto-RTs of 55000:55000 and the service is healthy.



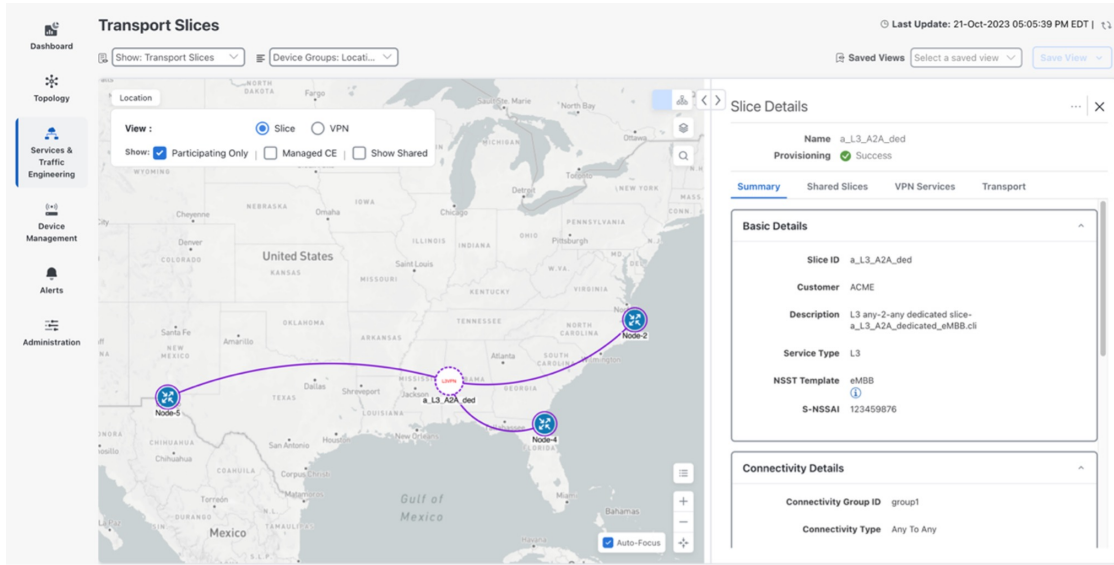
Step 5 In the Slice Details panel, click **X** to close the VPN Services tab and Slice Details panel.

Step 6 In the Transport Slices panel, click **⋮** in the Actions column for the newly created slice, **a_L3_A2A_ded**, and select **View Details**.



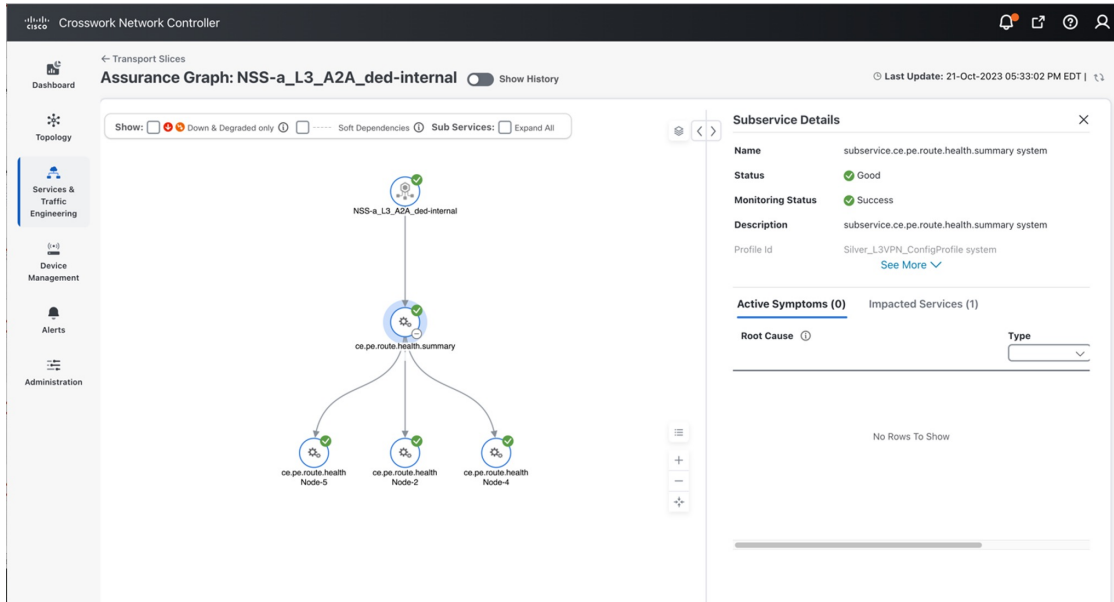
Step 7 In the Slice Details panel, select the **Summary** tab to view the slice details: Basic Details, Connectivity Details, Service Demarcation point (SDP).

Step 5 Visualize and Validate the New Slice Deployment



Step 8

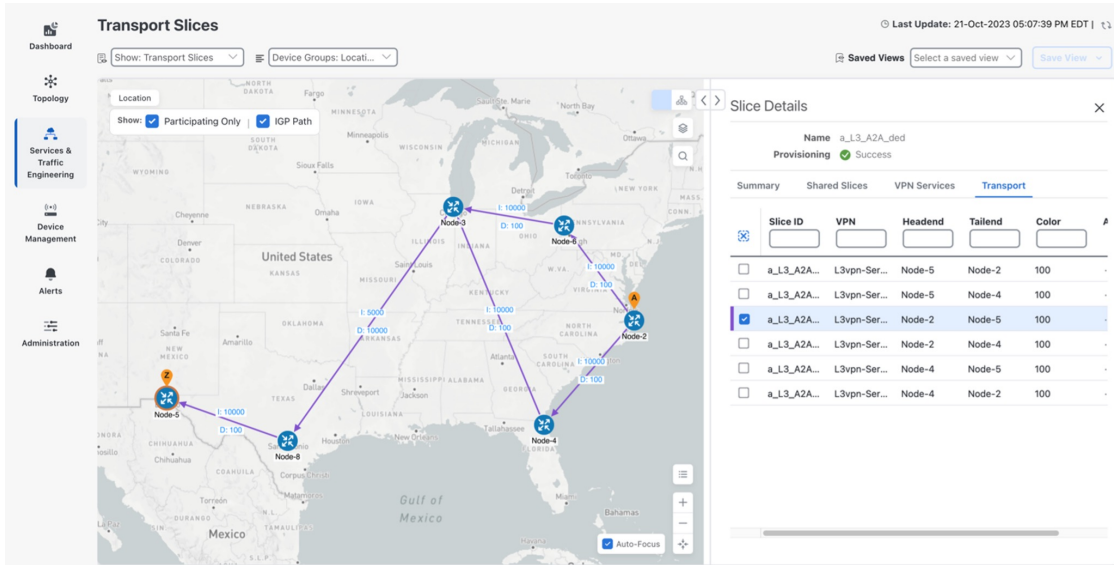
Again, select the VPN Services tab. Click ... in the Actions column for the newly created slice, **a_L3_A2A_ded**, and select **Assurance Graph** so to view the Monitoring Status and the status of the Service Health components defined in the Service Assurance Heuristics Package that was included in the Slice eMBB intent.



Step 9

As additional slices are added, you can visualize and validate further details, such as forwarding path. For example, here is a sample forwarding path for slice traffic from Node-2 to Node-5. A few observations.

- The high BW link between Node-3 and Node-8 (IGP=5k) is used (which was the desired intent), but this link also has a delay of high delay of 10ms (in each direction). Since latency was not an intent objective, this is fine.
- Also notice that Equal Cost Multi-Pathing is used when available (multiple ECMP paths between Node-2 and Node-3).



Step 10

Using the example above, once additional slices are added, if external Accedian probes were also installed at the CPE sites connected to the Slice endpoints at Node-2 and Node-5, an ~11ms latency (each way) between the two sites can be seen (example below). This is accurate because the link between Node-3 and Node-8 has a latency of ~10ms in each direction.



Summary and Conclusion

As we observed in this example, users can utilize Cisco Crosswork Network Controller to create a transport slice which has Layer3 any-to-any connectivity across three endpoints, using the enhanced Mobile Broadband (eMBB) catalog intent. The eMBB intent provides the highest bandwidth available path (including proper QoS marking/scheduling treatment), along with some basic service assurance capabilities such as endpoint interface status and PE-CE route health.

