

# Inter-Fabric L3Out for ACI Fabrics, 4.2.1

# Table of Contents


New and changed information . . . 1
Inter-Fabric L3Out Overview . . . 2
Inter-Fabric L3Out Guidelines and Limitations . . . 3
Configuring External TEP Pool . . . 4
Configuring External EPG to Use Inter-Fabric L3Out . . . 6
Creating a Contract for Inter-Fabric L3Out . . . 8
Use Cases . . . 9
- Configuring Nexus Dashboard using the PATCH API for static routes and ports . . . 9
  - Obtain the policy UUIDs . . . 9
  - PATCH API guidelines and limitations . . . 12
- Inter-Fabric L3Out for Application EPGs (Intra-VRF) . . . 12
- Shared Services with Inter-Fabric L3Out for Application EPGs (Inter-VRF) . . . 15
- Inter-Fabric Transit Routing . . . 18

Copyright . . . 21

# New and changed information

The following table provides an overview of the significant changes up to this current release. The table does not provide an exhaustive list of all changes or of the new features up to this release.

| Release Version | Feature | Description |
| --- | --- | --- |
| Nexus Dashboard 4.2.1 | There were no major changes from the previous release. | |

# Inter-Fabric L3Out Overview

Nexus Dashboard enables a number of scenarios in which endpoints located in one fabric are able to establish connectivity with entities, such as external network, mainframe, or service nodes, reachable through a remote L3Out.

These include the following:

- L3Out across fabrics-endpoints in an application EPG in one fabric using an L3Out in another fabric (both part of the same VRF).
- Inter-fabric transit routing-establishing communication between entities (such as endpoints, network devices, service nodes) connected behind L3Outs deployed in different fabrics (both L3Outs part of the same VRF).
- Shared services for inter-fabric L3Out-application EPG to remote L3Out or inter-fabric transit routing across VRFs.

The following sections are divided into the generic GUI procedures you can follow to create the objects required to implement inter-fabric L3Out use cases followed by overview and workflows specific to each supported use case scenario.



The term "inter-fabric L3Out" refers to the functionality allowing communication to external resources reachable via the L3Out connection of a remote fabric. However, in this document, the term may also be used to indicate the specific remote L3Out object.

The following sections describe how to configure an inter-fabric L3Out for EPG-to-L3Out use cases without Policy-Based Redirect (PBR).

- If you want to insert service chaining in the contract between an EPG and a remote L3Out to enable PBR, see Inter-Fabric L3Out with PBR instead.
- If you want to enable PBR between L3Outs in different fabrics (transit routing with PBR), see Inter-Fabric Transit Routing with PBR.

# Inter-Fabric L3Out Guidelines and Limitations

When configuring inter-fabric L3Out, you must consider the following:

- The steps described in the following sections assume you have L3Out connectivity already configured for your fabrics.

  This includes creating the L3Out template, creating the L3Out object and defining its configuration, and deploying the configuration to the fabric(s). Detailed information on configuring L3Outs is available in the External Connectivity (L3Out) chapter.

- Inter-fabric L3Out is supported for IPv4 and IPv6.
- With inter-fabric L3Out, in addition to the BGP eVPN sessions that are always established between fabrics in Multi-Fabric topology, MP BGP VPNv4 (or VPNv6) sessions are created to support the inter-fabric L3Out feature.
- You can now associate a bridge domain in one fabric with the L3Out in another fabric, however they must both be in the same VRF.

  This association is performed at the fabric-local level and is required to advertise the BD subnet out of the remote L3Out and ensure that inbound traffic to the BD can be maintained even if the local L3Out failed.

  However, instead of associating a bridge domain, we recommend that you define outbound route-maps for the L3Outs, including the BD prefixes that need to be advertised out.

- The Policy Control Enforcement direction for the VRF associated to the inter-fabric L3Out must be kept configured in the default ingress mode.
- The following scenarios are not supported with inter-fabric L3Out and remote leaf (RL):
  - Transit routing between L3Outs deployed on RL pairs associated to separate fabrics
  - Endpoints connected to a RL pair associated to a fabric communicating with the L3Out deployed on the RL pair associated to a remote fabric
  - Endpoints connected to the local fabric communicating with the L3Out deployed on the RL pair associated to a remote fabric
  - Endpoints connected to a RL pair associated to a fabric communicating with the L3Out deployed on a remote fabric
- The following other features are not supported with inter-fabric L3Out in Multi-Fabric:
  - Multicast receivers in a fabric receiving multicast from an external source via another fabric L3Out. Multicast received in a fabric from an external source is never sent to other fabrics. When a receiver in a fabric receives multicast from an external source it must be received on a local L3Out.
  - An internal multicast source sending multicast to an external receiver with PIM-SM any source multicast (ASM). An internal multicast source must be able to reach an external Rendezvous Point (RP) from a local L3Out
  - GOLF
  - Preferred Groups for External EPG

# Configuring External TEP Pool

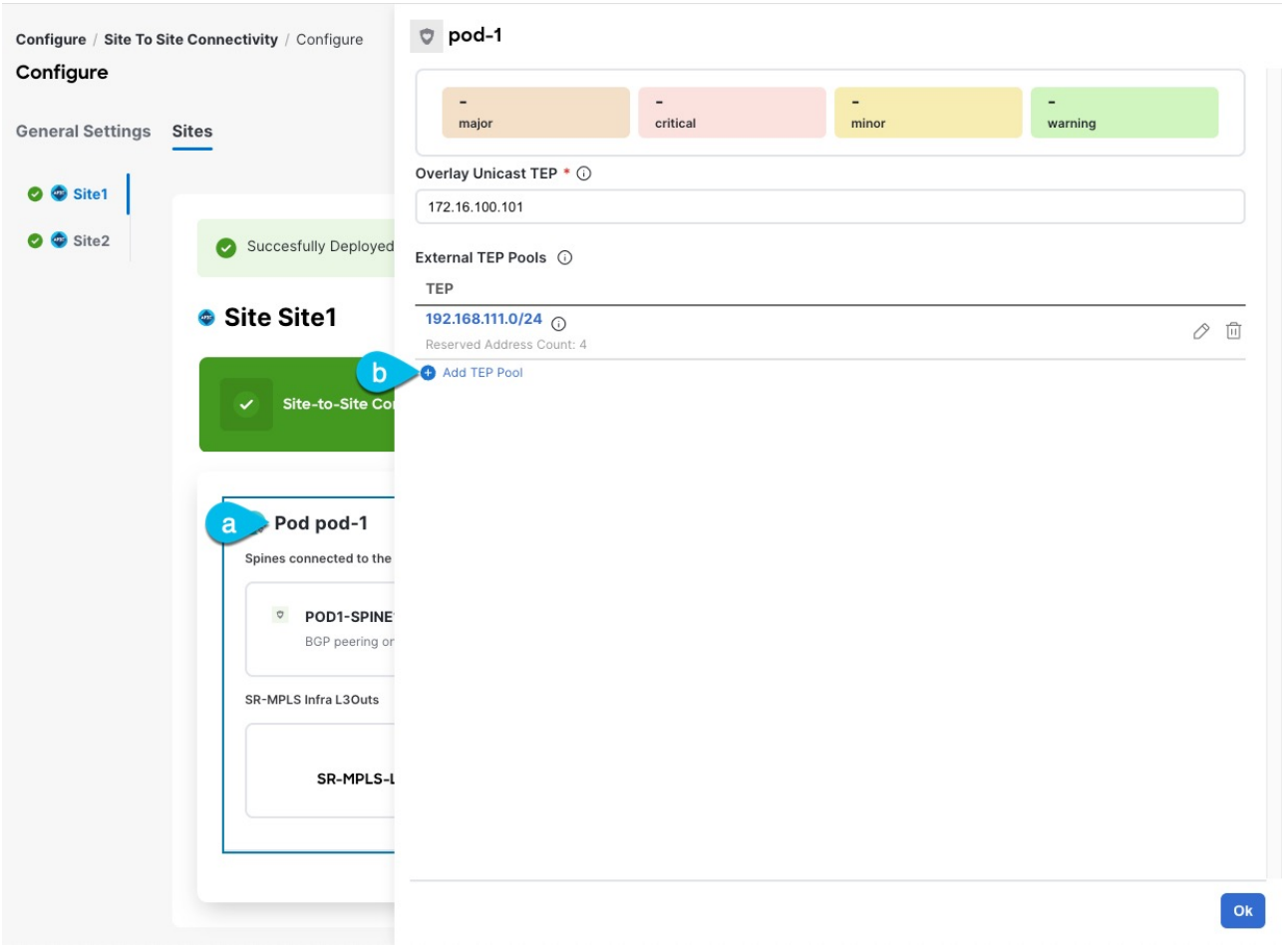
Inter-fabric L3Out requires an external TEP address for the border leaf switches in each pod. If you already have an external TEP pool that is configured, for example for another feature such as Remote leaf switch, the same pool can be used. The existing TEP pool will be inherited by the Cisco Nexus Dashboard and shown in the GUI as part of the infra configuration. Otherwise, you can add a TEP pool in the GUI, as described in this section.

Every pod must be assigned a unique TEP pool and it must not overlap with any other TEP pool in the fabric.

1. Navigate to the **Orchestration** window.

   **Manage > Orchestration**

2. Click **Inter-Fabric Connectivity**.
3. Click **Next** in the **Fabrics** page.
4. In the **Fabrics Details** page, locate the appropriate fabric in the **Fabric Settings** area and click **Edit**.
5. In the Pod Settings area, click on the name of the pod that you want to configure and then click **+Add TEP Pool**.




6. In the **Add TEP Pool** window, specify the **External TEP Pool** that you want to configure for that fabric and the **Reserved Address Count**.

For the TEP pool, provide the subnet and the subnet mask, for example 192.168.111.0/24.



You must ensure that the TEP pool you are adding does not overlap with any other TEP pools or fabric addresses.
Multiple disjointed TEP pools can be configured , so you are not required to specify a large TEP pool from the beginning.

7. Repeat the process for each fabric and pod where you plan to use inter-fabric L3Outs.

# Configuring External EPG to Use Inter-Fabric L3Out

*Before you begin:*

- Ensure that you have read and completed the requirements described in Inter-Fabric L3Out Guidelines and Limitations.
- You must have created and deployed the L3Outs in each fabric as part of configuring the fabric's external connectivity.

  Detailed information on configuring L3Outs is available in the External Connectivity (L3Out) chapter.

This section describes how to create an external EPG that will be associated to the inter-fabric L3Out. You can then use this external EPG and contracts to configure specific use cases for endpoints in one fabric to use an L3Out in another fabric or to configure L3Out-to-L3Out transit routing.

1. Select the schema and template where you want to create the external EPG.

   If you create the external EPG in a template that is associated to multiple fabrics, the external EPG will be stretched across all of those fabrics. This is recommended when the L3Outs defined in those fabrics provide access to a set of common external resources, for example the WAN.

   If you create the external EPG in a template that is associated with a single fabric, the external EPG will be created in that fabric only. This is recommended when the L3Out defined in that fabric provides access to external resources accessible only from that fabric.

2. Create an external EPG.
   a. In the main pane, select **+Create Object > External EPG**.
   b. Provide the **Display Name** for the external EPG.

      For example, eepg-inter-fabric-l3out

   c. From the **Virtual Routing & Forwarding** dropdown, select the same VRF you associated with the L3Out.
3. Map the external EPG to the L3Out.
   a. In the template view, select the tab for the fabric where the external EPG is deployed.
   b. Select the external EPG you created in the previous step.
   c. In the **<external-epg-name> on <fabric-name>** properties sidebar, choose the L3Out you created from the **L3Out** dropdown.

      Note that both the APIC-managed and the Orchestrator-managed L3Outs are available for selection. You can select either the L3Out you have created and deployed from Nexus Dashboard or pick an L3Out that exists in the fabric's APIC.

4. Configure one or more subnets for the external EPG.
   a. Select the external EPG.
   b. In the right sidebar, click **+Add Subnet**.

c. In the **Add Subnet** window's **Subnet** field, provide the subnet's network prefix.

d. (Optional) Provide a descriptive **Name** for the subnet.

e. Provide any required options for this subnet.

The prefixes and options you configure depend on the specific use cases:

- To classify the inbound traffic as belonging to the external EPG, select the **External Subnets for External EPG** flag for the specified prefix. Depending on the specific use case, this allows you to apply a contract with an internal EPG or with the external network domain reachable via a different L3Out.
- To advertise the external prefixes learned from another L3Out (in the same fabric or in a remote fabric) out of this L3Out, select the **Export Route Control** flag for the specified prefix. When specifying the 0.0.0.0/0 prefix, the **Aggregate Export** flag can be selected to advertise all prefixes out of the L3Out; if the **Aggregate Export** flag is not enabled, only the default route 0.0.0.0/0 would be advertised, if present in the routing table of the border leaf nodes.



However, we recommend that you use the Outbound Route-Map associated to the L3Out to match the external prefixes (in addition to the BDs' subnets) to be advertised out instead.

- To filter out specific routes received from the external network, select the **Import Route Control** flag for the specified prefix. If specifying the 0.0.0.0/0, you can also choose the **Aggregate Import** option.

  Note that this is possible only when peering BGP with the external routers.



However, similar to the previous bullet point, we recommend that you use the Outbound Route-Map associated to the L3Out to match the external prefixes (in addition to the BDs' subnets) to be advertised out instead.

- To leak routes to different VRFs, select the **Shared Route Control** and the associated **Aggregate Shared Routes** flags, as well as the **Shared Security Import** flag. These options are required for the specific use case of inter-VRF shared L3Out and inter-VRF inter-fabric transit routing.

5. (Optional) Enable **Include in Preferred Group** if you want the external EPG to be part of the EPG preferred group.
6. (Optional) From the **QoS Level** dropdown, select the QoS level for this external EPG.

   For additional information about QoS in ACI fabrics, see *Cisco APIC and QoS*.

   For additional information about configuring QoS Levels in your Nexus Dashboard, see QoS Preservation Across IPN.

# Creating a Contract for Inter-Fabric L3Out

This section describes how to create a filter and a contract you will use to enable communication between an application EPG deployed in a fabric and the external EPG associated to an L3Out in a different fabric (inter-fabric L3Out functionality).

1. Select the template where you want to create contract and filter.

   You can use the same schema and template where you created the VRF and the external EPG or you can choose a different schema and template.

   You must define the contracts and the filters explicitly on all fabrics where they will be used.

2. Create a filter.
   a. In the main pane, select **+Create Object > Filter**.
   b. Provide the **Display Name** for the filter.
   c. Click **+ Entry** and provide the filter entry information specific to the kind of traffic you want to allow.
   d. Click **Ok** to save the filter.
3. Create a contract
   a. In the main pane, select **+Create Object > Contract**.
   b. In the right pane, provide the **Display Name** for the contract
   c. Select the appropriate **Scope** for the contract.
      - If both inter-fabric L3Out and application EPG are in the same VRF, set the scope to vrf.
      - If the inter-fabric L3Out and application EPG are in different VRFs but the VRFs are in the same tenant, set the scope to tenant.
      - If the inter-fabric L3Out and application EPG are in different VRFs and the VRFs are in different tenants, set the scope to global.

   d. Ensure that the **Apply both directions** option is enabled if you want the same filter to apply for both consumer-to-provider and provider-to-consumer directions.

      With this option enabled, you need to provide the filters only once and they will apply for traffic in both directions.

   e. In the **Filter Chain** area, click **Create Filter** and choose the filter you created in the previous step.
   f. Click **Ok** to save the contract.

# Use Cases

## Configuring Nexus Dashboard using the PATCH API for static routes and ports

*Before you begin:*

Configure these items.

- L3Out and EPG configuration.
  - Ensure that the L3Outs and EPGs you intend to modify using the PATCH API are already created and deployed in your Nexus Dashboard environment.
  - Verify that the L3Outs have basic connectivity configured and the EPGs are associated with the appropriate bridge domains and VRFs.
- Use these IDs for the objects you plan to modify.
  - Schema ID: The unique identifier for Nexus Dashboard schema containing the template.
  - Template ID: The unique identifier for the template containing the L3Out or EPG.
  - Site ID: The unique identifier for the site where the template is deployed.

This section describes the PATCH API that allows you to make bulk configuration changes to L3Outs and EPGs efficiently. By using the PATCH API, you can significantly improve the performance of large-scale configurations. Because, when configuring many static routes in L3Outs or static ports in EPGs using Nexus Dashboard, making changes through the user interface or using the scripts may be slow and inefficient.

### Obtain the policy UUIDs

Nexus Dashboard generates Universally Unique Identifier (UUID) for each policy object when it is configured in a Nexus Dashboard schema or template and saved. The **trackPolicyRef** or **monitoringPolicyRef** fields in the PATCH payload require the **UUIDs** of existing IPSLA track-list and monitoring policies configured in your ACI fabric. Nexus Dashboard automatically generates these **UUIDs** when the track-list and monitoring policies are configured and saved.



You must perform a GET request to find the 'policyRefs' before using them in the PATCH method.

To obtain these **UUIDs**:

1. Access Nexus Dashboard API: Use a REST API to perform a GET request of the templates that contains the IPLSA track-list and monitoring policies.
   - URL: https://{{mso-target}}/mso/api/v1/schemas/{schemaId} or https://{{mso-target}}/mso/api/v1/templates/{templateId}
   - Replace {schemaId} and {templateId} with the appropriate IDs.
2. Examine the Response: The GET request returns a JSON response containing the complete schema or template definition.
3. Locate the Policy Definitions: Search the JSON response for the definitions of your IPSLA track-

list and monitoring policies.

4. Extract the **UUIDs**: Within the policy definitions, locate the **UUID** field. This field contains the **UUID** you need to use in your PATCH payload's **trackPolicyRef** or **monitoringPolicyRef** fields.

An example snippet from a GET response:

```
"bfdProfiles": [
  {
    "uuid": "65cf00c1-9b52-4163-a1d2-c1f6d0660a51",  // This is the *UUID* you need!
    "name": "My-BFD-Profile",
    "description": "BFD profile for static route tracking",
    // ... other profile details ...
  }
]
```

**Use Case 1: Adding static routes to L3Outs using the PATCH API**

This feature is supported in Nexus Dashboard 3.2.1x, Nexus Dashboard Orchestrator 4.4.1x and later. These steps describe the configuration of how to use the PATCH API to add static routes to L3Outs in an L3Out template.

1. Identify the L3Out template: Determine the schema ID and template ID of the L3Out template you want to modify.
2. Construct the PATCH request: Use the PATCH method with the appropriate URL and payload.
   - URL: https://{{mso-target}}/mso/api/v1/templates/{templateId}
   - Replace {templateId} with the actual template ID.

```
[
  {
    "op": "add",
    "path": "/l3outTemplate/l3outs/0/nodes/0/staticRoutes/-",
    "value": {
      "prefix": "6.6.6.6/24",
      "fallbackPref": 1,
      "enableBFDTracking": false,
      "trackPolicyRef": "65cf00c1-9b52-4163-a1d2-c1f6d0660a51",
      "nullNextHop": false,
      "nextHops": [
        {
          "nextHopIP": "6.6.6.6",
          "preference": 0,
          "monitoringPolicyRef": "4fa7f418-7ac4-4a1b-a612-672c72a436dd"
        }
      ]
    }
```

```
  },
  {
    "op" : "add",
    "path" : "/l3outTemplate/l3outs/0/nodes/0/staticRoutes/-",
    "value" : {
      "prefix" : "7.7.7.7/24",
      "fallbackPref" : 1,
      "enableBFDTracking" : false,
      "trackPolicyRef" : "65cf00c1-9b52-4163-a1d2-c1f6d0660a51",
      "nullNextHop" : false,
      "nextHops" : [
        {
          "nextHopIP" : "8.8.8.8",
          "preference" : 0,
          "monitoringPolicyRef" : "4fa7f418-7ac4-4a1b-a612-672c72a436dd"
        }
      ]
    }
  }
]
```

3. Send the PATCH request: Use a REST client to send the PATCH request to Nexus Dashboard.
4. Verify the configuration: After a successful PATCH request (response code '204 No Content'), verify that the static routes have been added to the L3Out template using the user interface or from the GET API.

**Use Case 2: Adding static ports to EPGs using the PATCH API**

This feature is supported in Nexus Dashboard Orchestrator 4.2.1d and later versions. It is also supported in Nexus Dashboard 3.2.1x and Nexus Dashboard Orchestrator 4.4.1x. These steps describe the configuration of how to use the PATCH API to add static ports to EPGs.

1. Identify the Schema, Template, and Site: Determine the schema ID, template name, and site ID of the EPGs you want to modify.
2. Construct the PATCH request: Use the PATCH method with the appropriate URL and payload.
   - URL: https://{{mso-target}}/mso/api/v1/schemas/{schemaId}
   - Replace {schemaId} with the actual schema ID.
   - Payload: Construct a JSON payload with this structure:
     - In this example the
       - **siteId** is '67c8039a031ee45a8bcd9bdb' and the **TemplateName** is 'T1'

```
[
  {
    "op" : "add",
```

```
    " path" : " /sites/67c8039a031ee45a8bcd9bdb-
T1/anps/AP1/epgs/EPG1/staticPorts/-" ,
    " value" : {
      " type" : " port" ,
      " path" : " topology/pod-1/paths-101/pathep-[eth1/30]" ,
      " portEncapVlan" : 30,
      " deploymentImmediacy" : " lazy" ,
      " mode" : " regular"
    }
  },
  {
    " op" : " add" ,
    " path" : " /sites/67c8039a031ee45a8bcd9bdb-
T1/anps/AP1/epgs/EPG1/staticPorts/-" ,
    " value" : {
      " type" : " port" ,
      " path" : " topology/pod-1/paths-101/pathep-[eth1/40]" ,
      " portEncapVlan" : 40,
      " deploymentImmediacy" : " lazy" ,
      " mode" : " regular"
    }
  }
]
```


3. Send the PATCH request: Use a REST client to send the PATCH request to Nexus Dashboard.
4. Verify the configuration: After a successful PATCH request (response code '204 No Content'), verify that the static ports have been added to the EPGs.

### PATCH API guidelines and limitations

- For large deployments, you must limit the number of bindings per patch operation to avoid potential payload size limitations and provisioning errors. Limit the payload to 500 bindings per operation.
- The patch operation also validates duplicate entries and any data entry errors.
- The L3Out template PATCH API supports index-based patching only.

## Inter-Fabric L3Out for Application EPGs (Intra-VRF)

*Before you begin:*

You need to have the following already configured:

- External connectivity (L3Out) in each fabric, as described in External Connectivity (L3Out).

  In this use case, a separate L3Out will be imported or created in each fabric-specific template.

- A schema with four templates.

Create a template for each fabric (for example, template-fabric1 and template-fabric2) where you configure the objects unique to those fabrics, such as the application EPG and the L3Outs.

In addition, create two more stretched templates: one for the stretched EPGs, External EPGs, and BDs and the second one for the VRFs, contracts, and filters.

- The external EPG for the inter-fabric L3Out, as described in Configuring External EPG to Use Inter-Fabric L3Out.

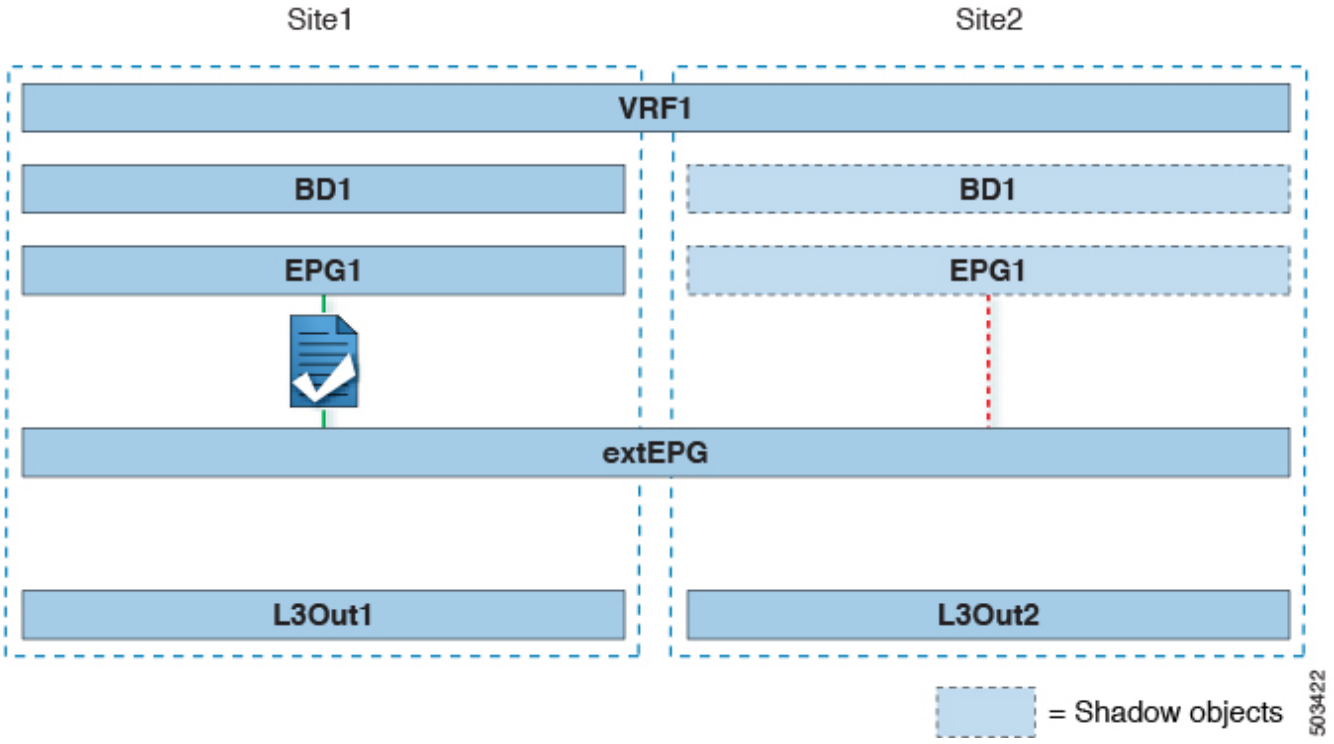
  In this use case, the external EPG is configured as a stretched object that is defined in one of the stretched template (for example, template-stretched-ext-epg). Assuming that the external EPG provides access to the entire external address space, we recommend configuring a 0.0.0.0/0 prefix for classification to avoid specifying a long list of more specific prefixes.

- The contract that you will use between the application EPG and the L3Out external EPG, as described in Creating a Contract for Inter-Fabric L3Out.

  We recommend creating the contract and the filter in the second stretched template (for example, template-stretched-vrf-contract), which also contains the VRF.

This section describes the configuration that is required to allow endpoints that are part of an application EPG to communicate with the external network domain reachable through an L3Out deployed in another fabric but within the same VRF (intra-VRF).

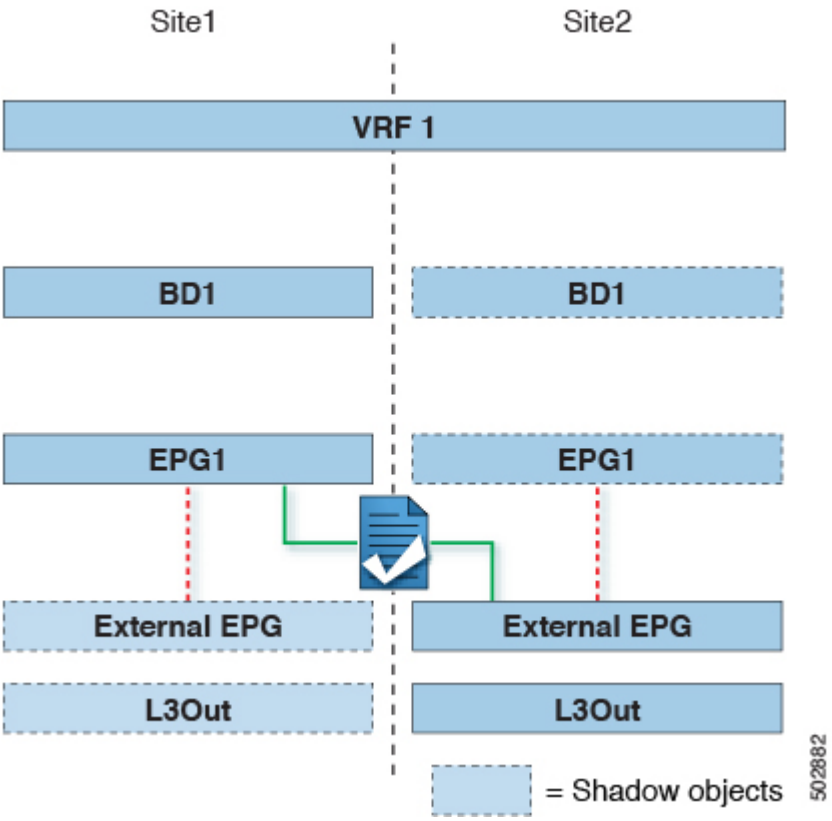
The first figure below shows a stretched external EPG and the associated L3Outs which will be created in both fabrics. An application EPG (EPG1) is created in Fabric 1 and has a contract with the external EPG. This use case is recommended when the L3Outs in the separate fabrics provide access to a common set of external resources. It simplifies the policy definition and external traffic classification, while still allowing you to apply route-map policies separately on each L3Out for the independent APIC domains.




*Stretched External EPG*

The second figure below shows a similar use case but with the external EPG being deployed to only

the fabric where the physical L3Out is located. The application EPG and the contract are configured in the same exact way to allow the traffic flow between the EPG in one fabric and the physical L3Out in the other.




*Non-Stretched (Fabric-Local) External EPG*

The following steps describe the configuration that is required to implement the use case shown in Figure 1, which represents the most common scenario. If you want to deploy the use case shown in Figure 2, you can adapt the procedure with minor changes.

1. Navigate to the **Orchestration** window.

   **Manage > Orchestration**

2. Choose **Tenant Templates > Applications**.
3. Select the schema and template for the application EPG and bridge domain.

   In this use case, you associate the template to Fabric1.

4. Configure an application EPG and its bridge domain belonging to the same VRF as the L3Out.

   If you already have an EPG that will use the inter-fabric L3Out, you can skip this step.

   You can create a new or import an existing EPG and bridge domain as you typically would.

5. Assign the contract to the application EPG.
   a. Select the EPG.
   b. In the right sidebar, click **+Contract**.
   c. Select the contract that you created in previous section and its type.

      You can choose whether the application EPG is the consumer or the provider.

6. Assign the contract to the external EPG mapped to the remote L3Out.
    a. Select the template-stretched where the external EPG is located.
    b. Select the external EPG.
    c. In the right sidebar, click **+Contract**.
    d. Select the contract that you created in previous section and its type.

        If you chose the application EPG to be the consumer, choose provider for the external EPG. Otherwise, choose consumer for the external EPG.

7. Associate the application EPG's bridge domain with the L3Out.

    This enables the BD subnet to be advertised out of the L3Out toward the external network domain. Note that one or more subnets associated to the BD must be configured with the **Advertised Externally** option to be advertised out of the L3Out

    a. In the left sidebar, under **Fabrics**, select the application EPG's template.
    b. Select the bridge domain associated with the application EPG.
    c. In the right sidebar, click **+L3Out**.
    d. Select the inter-fabric L3Out you created.

        For the use case shown in Figure 1, associate the BD to both the L3Outs defined in Fabric 1 and Fabric 2 to ensure that the external network can have access to the EPG from both paths. Specific policies can be associated to the L3Out or to the external routers to ensure that a specific L3Out path is normally preferred for inbound traffic. We recommend this when the EPG and BD are local to a fabric (as in the specific example) to avoid suboptimal inbound traffic path through the remote fabric's L3Out.

8. Deploy the schema.

# Shared Services with Inter-Fabric L3Out for Application EPGs (Inter-VRF)

*Before you begin:*

You must have the following already configured:

- External connectivity (L3Out) in each fabric, as described in External Connectivity (L3Out).

  In this use case, a separate L3Out will be imported or created in each fabric-specific template.

- A schema with four templates.

  Create a template for each fabric (for example, template-fabric1 and template-fabric2) where you configure the objects unique to those fabrics, such as the application EPG and the L3Outs.

  In addition, create two more stretched templates: one for the stretched External EPG and the second one for the contracts, and filters.

- The external EPG for the inter-fabric L3Out, as described in Configuring External EPG to Use Inter-Fabric L3Out.

In this use case, the external EPG is configured as a stretched object that is defined in the stretched template (template-stretched). Assuming that the external EPG provides access to the entire external address space, we recommend configuring a 0.0.0.0/0 prefix for classification to avoid specifying a long list of more specific prefixes.

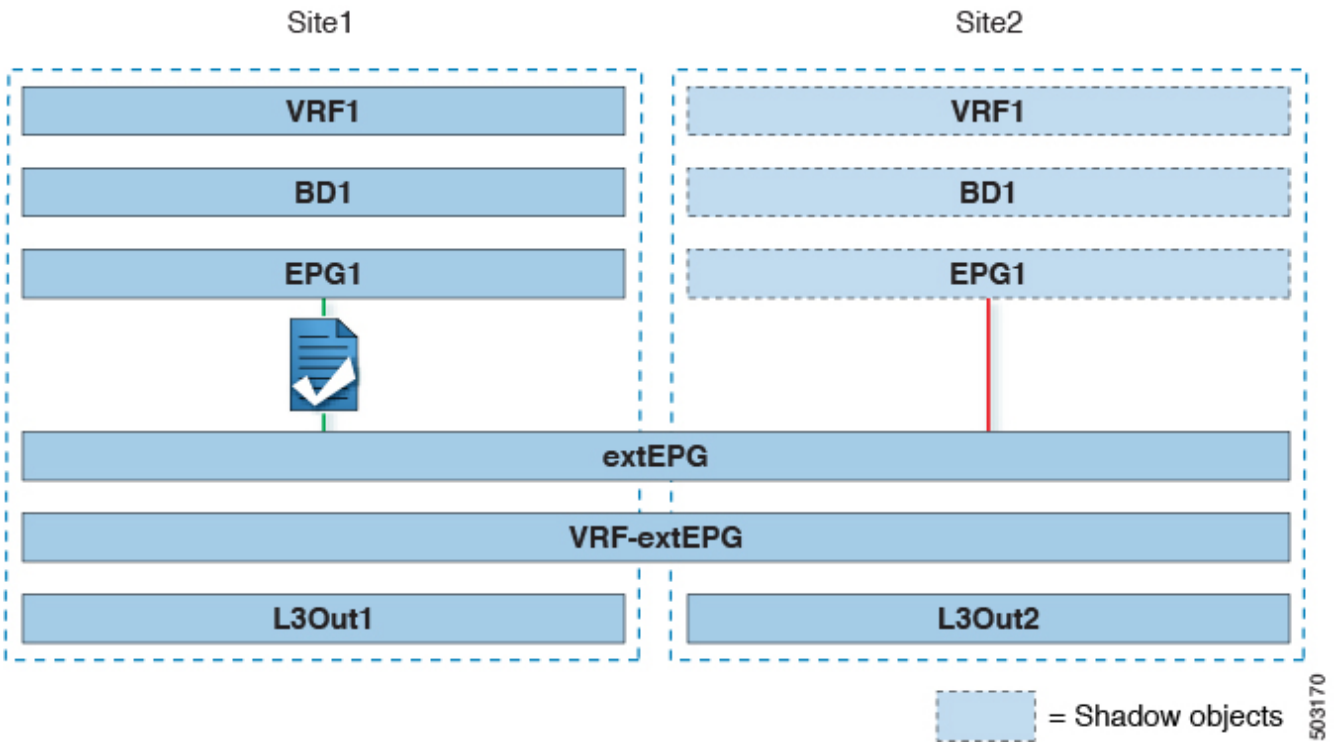
For this specific shared service, use case, you are also required to enable the **Shared Route Control** and the **Shared Security Import** flags for one or more subnets that are associated to one or more external EPGs of the remote L3Out. If you are using the 0.0.0.0/0 prefix for classification on the external EPG, in addition to the **Shared Route Control** flag, also enable the **Aggregate Shared Routes** flag.

- The contract that you will use between the application EPG and the L3Out external EPG, as described in Creating a Contract for Inter-Fabric L3Out.

  We recommend creating the contract and the filter in the stretched template (template-stretched).

This section describes the configuration that is required to allow endpoints that are part of an application EPG in one VRF to communicate with the external network domain reachable through an L3Out deployed in another fabric and different VRF, this is also known as " Shared Services" .

This scenario is recommended when the L3Outs in separate fabrics provide access to a common set of external resources. It simplifies the policy definition and external traffic classification, while still allowing you to apply route-map policies separately on each L3Out for the independent APIC domains.




*Stretched External EPG, Fabric-Local L3Outs and Application EPGs*

The following steps describe the configuration that is required to implement the use case shown in Figure 3.

1. Navigate to the **Orchestration** window.

   **Manage > Orchestration**

2. Choose **Tenant Templates > Applications**.
3. Select the schema and template for the application EPG and bridge domain.

   In this use case, you associate the template to Fabric1.

4. Configure an application EPG and its bridge domain belonging to a separate VRF from the L3Out's.

   If you already have an EPG that will use the inter-fabric L3Out, you can skip this step.

   You can create a new or import an existing EPG and bridge domain as you typically would.

5. Assign the contract to the application EPG.
   a. Select the EPG.
   b. In the right sidebar, click **+Contract**.
   c. Select the contract that you created in previous section and its type.

      You can choose whether the application EPG is the consumer or the provider.



If the application EPG is configured as provider, you must configure the subnet that is already defined under the BD also under the EPG to leak that route into the L3Out VRF. The same flags that are used under the BD for the subnet should also be set under the EPG. In addition to that, for the subnet under the EPG the flag **No default SVI Gateway** should also be enabled, since the default gateway function is enabled at the BD level.

6. Assign the contract to the external EPG mapped to the L3Outs.
   a. Select the template-stretched where the external EPG is located.
   b. Select the external EPG.
   c. In the right sidebar, click **+Contract**.
   d. Select the contract that you created in previous section and its type.

      If you chose the application EPG to be the consumer, choose provider for the external EPG. Otherwise, choose consumer for the external EPG.

7. Associate the application EPG's bridge domain with the L3Out.

   This enables the BD subnet to be advertised out of the L3Out toward the external network domain. The subnet(s) associated to the BD must be configured with the **Advertised Externally** option to be advertised out of the L3Out

   a. In the left sidebar, under **Fabrics**, select the application EPG's template.
   b. Select the bridge domain associated with the application EPG.
   c. In the right sidebar, click **+L3Out**.
   d. Select the inter-fabric L3Out you created.

      For the use case shown in Figure 1, associate the BD to both the L3Outs defined in Fabric 1 and Fabric 2 to ensure that the external network can have access to the EPG from both paths.

Specific policies can be associated to the L3Out or to the external routers to ensure that a specific L3Out path is normally preferred for inbound traffic. We recommend this when the EPG and BD are local to a fabric (as in the specific example) to avoid suboptimal inbound traffic path through the remote fabric's L3Out.

8. Deploy the schema.

# Inter-Fabric Transit Routing

*Before you begin:*

You must have the following already configured:

- External connectivity (L3Out) in each fabric, as described in External Connectivity (L3Out).

  In this use case, a separate L3Out will be imported or created in each fabric-specific template.

- A schema with three templates.

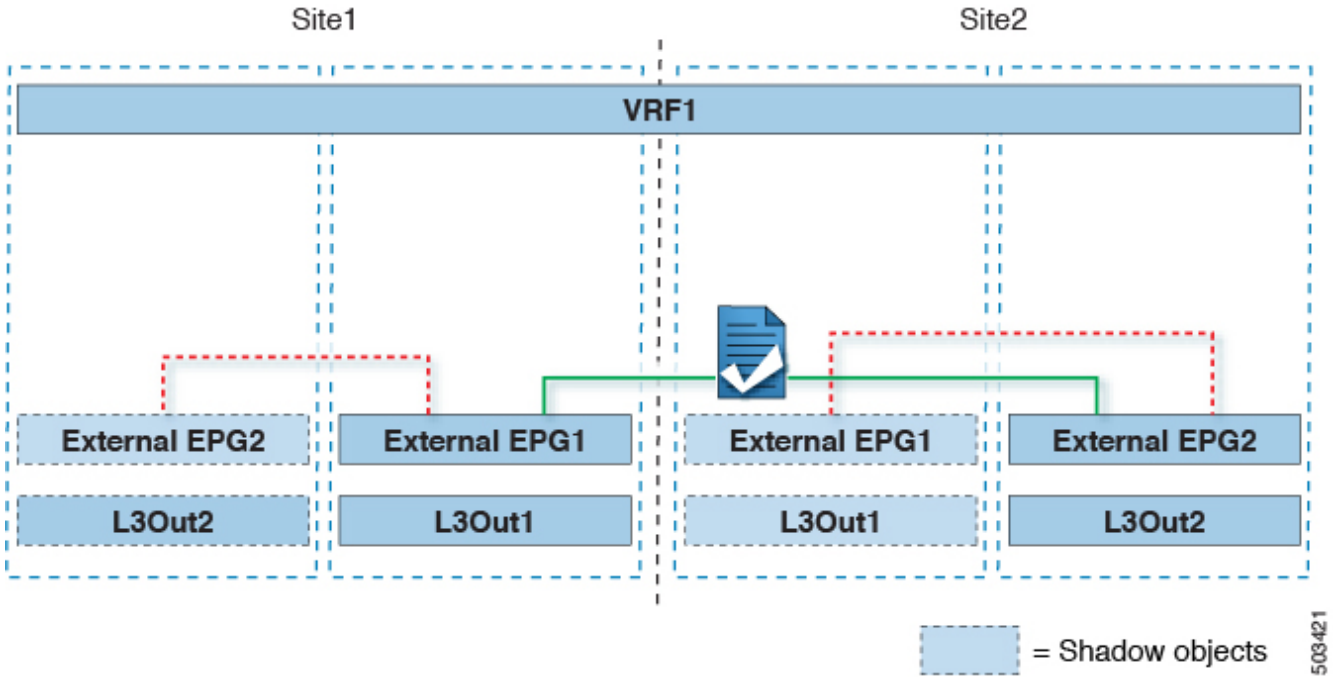
  Create a template for each fabric (for example, template-fabric1 and template-fabric2 ) where you configure the objects unique to that fabric, such as the application EPGs and the L3Outs. In addition, create a separate template (for example, `template-stretched `) that you use for the stretched objects, which in this case will be the external EPG.

- Two different external EPGs for two different L3Outs in different fabrics. You can use the same procedure to create both external EPGs, as described in Configuring External EPG to Use Inter-Fabric L3Out.
- The contract that you use between the L3Out external EPGs defined in each fabric, as described in Creating a Contract for Inter-Fabric L3Out.

  We recommend creating the contract and the filter in the stretched template (template-stretched).

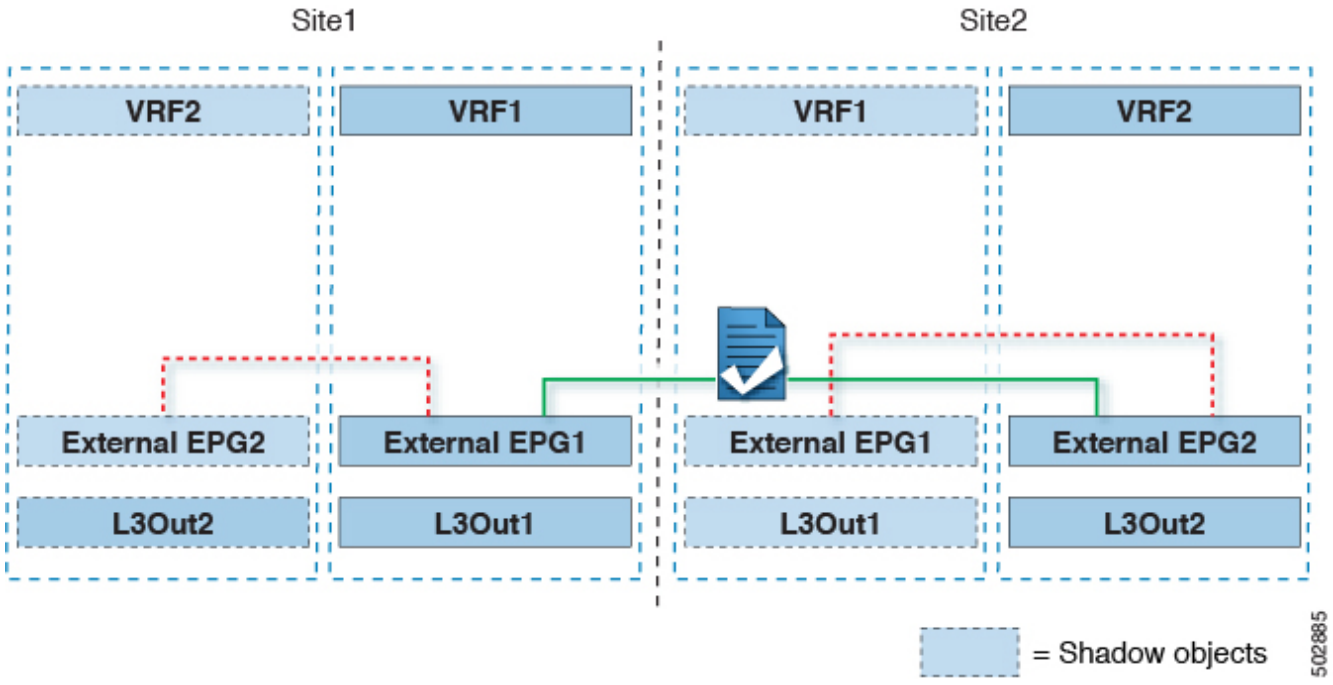
This section describes the use cases where the Multi-Fabric domain acts as a distributed router allowing communication between entities (endpoints, network devices, service nodes, and so forth) connected behind L3Outs deployed in different fabrics, a functionality normally saw as inter-fabric transit routing. The inter-fabric transit routing is supported for intra-VRF and inter-VRF use cases.

The figure below shows two L3Outs (L3Out1 and L3Out2) configured in different fabrics. Each L3Out is associated with a respective external EPG (External EPG1 and External EPG2). A contract between the two external EPGs allows communication between entities that are connected behind two different L3Outs in two different fabrics.




*Intra-VRF Inter-Fabric Transit Routing*

A similar configuration can be used when each fabric's L3Outs are in different VRFs.




*Inter-VRF Inter-Fabric Transit Routing*

The figures above show the two scenarios where the external EPGs and associated L3Outs are deployed as fabric-local objects; inter-fabric transit routing can support all the combinations where neither external EPG is stretched, one of them is stretched, or both are stretched between fabrics.

When deploying inter-fabric transit routing, the assumption is that the different external EPGs defined across fabrics are providing access to different external address spaces (not overlapping). A couple of options are hence possible for the configuration of the prefix that is used for classification:

- Define the 0.0.0.0/0 prefix for one of the external EPGs and specific prefixes on the other.

  The external prefixes that are received on L3Out1 must be advertised out of L3Out2 and

conversely.

- Define specific prefixes for each external EPG. In this case, you must ensure that the prefixes are not overlapping to avoid a fault from being raised by the fabric's APIC when the shadow external EPG is created in that fabric for a contract between the local and remote external EPGs.

  When using specific prefixes, the same prefixes that are configured for classification on External EPG1 must be configured with the **Export Route Control** flag set on External EPG2 and conversely.



No matter which of the two classifications approaches you deploy, for the inter-VRF scenario you must also set the **Shared Route Control** (in addition to **Aggregate Shared Routes** if using 0.0.0.0/0) and the **Shared Security Import** flags.

1. Navigate to the **Orchestration** window.

   **Manage > Orchestration**

2. Choose **Tenant Templates > Applications**.
3. Assign the contract to one of the external EPGs.
   a. Select the schema and template where the external EPG is located.
   b. Select the external EPG.
   c. In the right sidebar, click **+Contract**.
   d. Select the contract that you created in previous section and its type.

      Choose consumer or provider.
4. Assign the contract to the other external EPG.
   a. Select the schema and template where the external EPG is located.
   b. Browse to the template where the external EPG is located.
   c. Select the external EPG.
   d. In the right sidebar, click **+Contract**.
   e. Select the contract that you created in previous section and its type.

      Choose provider or consumer.
5. Deploy the templates to appropriate fabrics.

# Copyright

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.

The documentation set for this product strives to use bias-free language. For the purposes of this documentation set, bias-free is defined as language that does not imply discrimination based on age, disability, gender, racial identity, ethnic identity, sexual orientation, socioeconomic status, and intersectionality. Exceptions may be present in the documentation due to language that is hardcoded in the user interfaces of the product software, language used based on RFP documentation, or language that is used by a referenced third-party product.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: https://www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

© 2017-2026 Cisco Systems, Inc. All rights reserved.

**Americas Headquarters**
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706

USA
https://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883