

Schemas and Application Templates for ACI Fabrics, 4.3.1

Table of Contents

New and changed information . . . 1
Shadow objects . . . 2
  Other use cases with shadow objects . . . 3
  Hide shadow objects in the APIC GUI . . . 4
Create schemas and templates . . . 6
  Importing schema elements from ACI fabrics . . . 8
    Re-importing previously Orchestrator-managed configurations from ACI fabrics . . . 8
    Import schema elements from ACI fabrics . . . 9
  Configure VRFs . . . 11
  Configure bridge domains . . . 12
    Configure a bridge domain’s fabric-local properties . . . 16
  Configure application profiles and EPGs . . . 18
    Configure an EPG’s fabric-local properties . . . 20
  Configure contracts and filters . . . 25
  View schemas . . . 28
Clone schemas . . . 29
Copyright . . . 30

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.

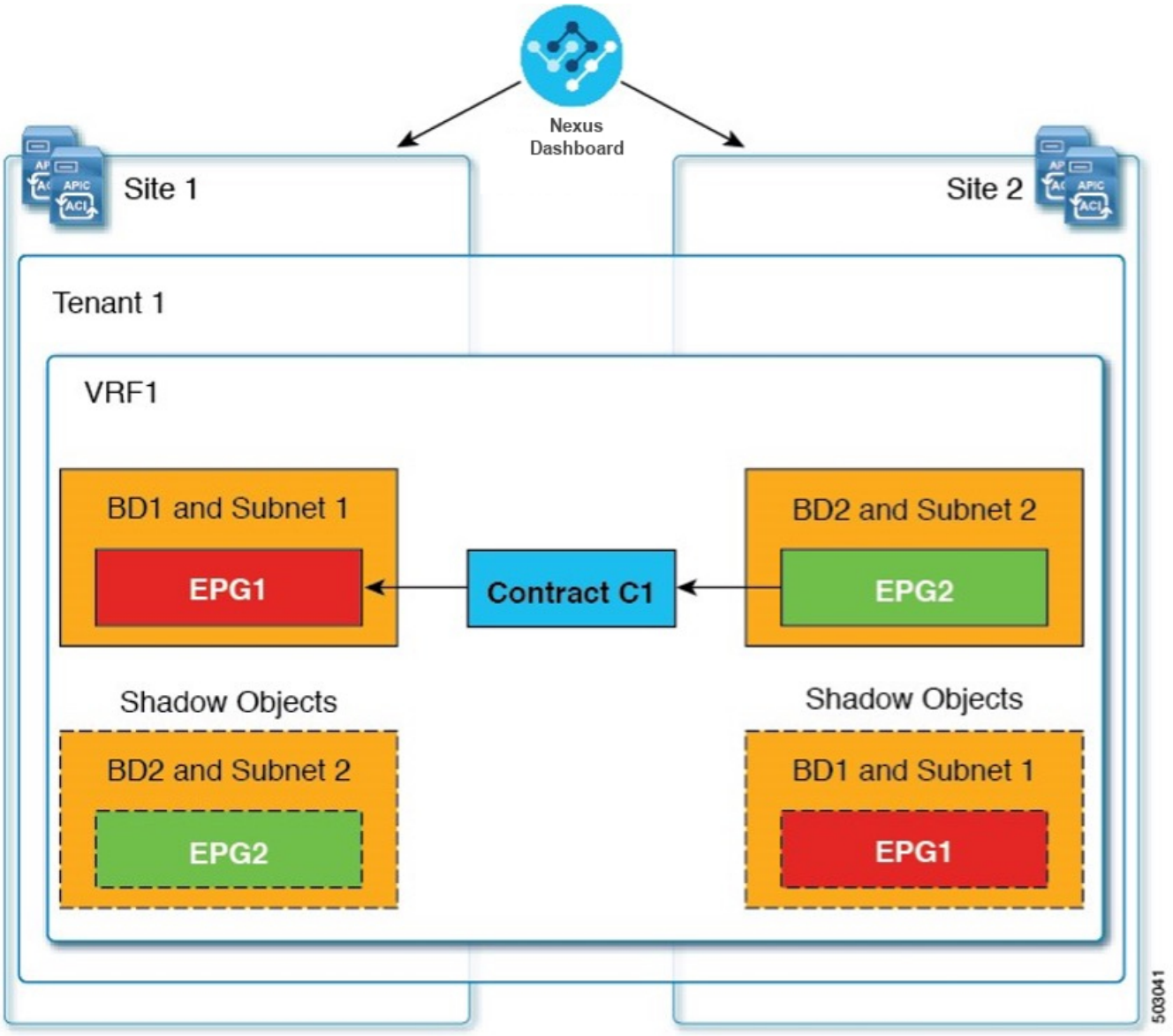
Release Version	Feature	Description
Nexus Dashboard 4.3.1	Support import for same-name policies in autonomous templates	Beginning with Nexus Dashboard 4.3.1, Nexus Dashboard supports importing same-name policies in autonomous templates and resolves relationship references within the fabric from which the objects are imported. During import, a same-named object in another fabric is not used to satisfy the relationship. For more information, refer to Import schema elements from ACI fabrics.

Shadow objects

When a contract exists between fabric-local EPGs in stretched VRF or in Shared Services use-cases where provider and consumer are in different VRFs and communicate through Tenant contracts, the EPGs and bridge domains (BDs) are mirrored on the remote fabrics. The mirrored objects appear as if they are deployed in each of these fabrics' controllers, while only actually being deployed in one of the fabrics. These mirrored objects are called "shadow" objects.

Shadow objects should not be removed using the APIC GUI.

For example, if a tenant and VRF are stretched between Fabric1 and Fabric2, provider EPG and its bridge domain are deployed in Fabric2 only, and consumer EPG and its domain are deployed in Fabric1 only, then corresponding shadow bridge domains and EPGs will be deployed as shown in the figure below. They appear with the same names as the ones that were deployed directly to each fabric.



The following objects can be shadowed:

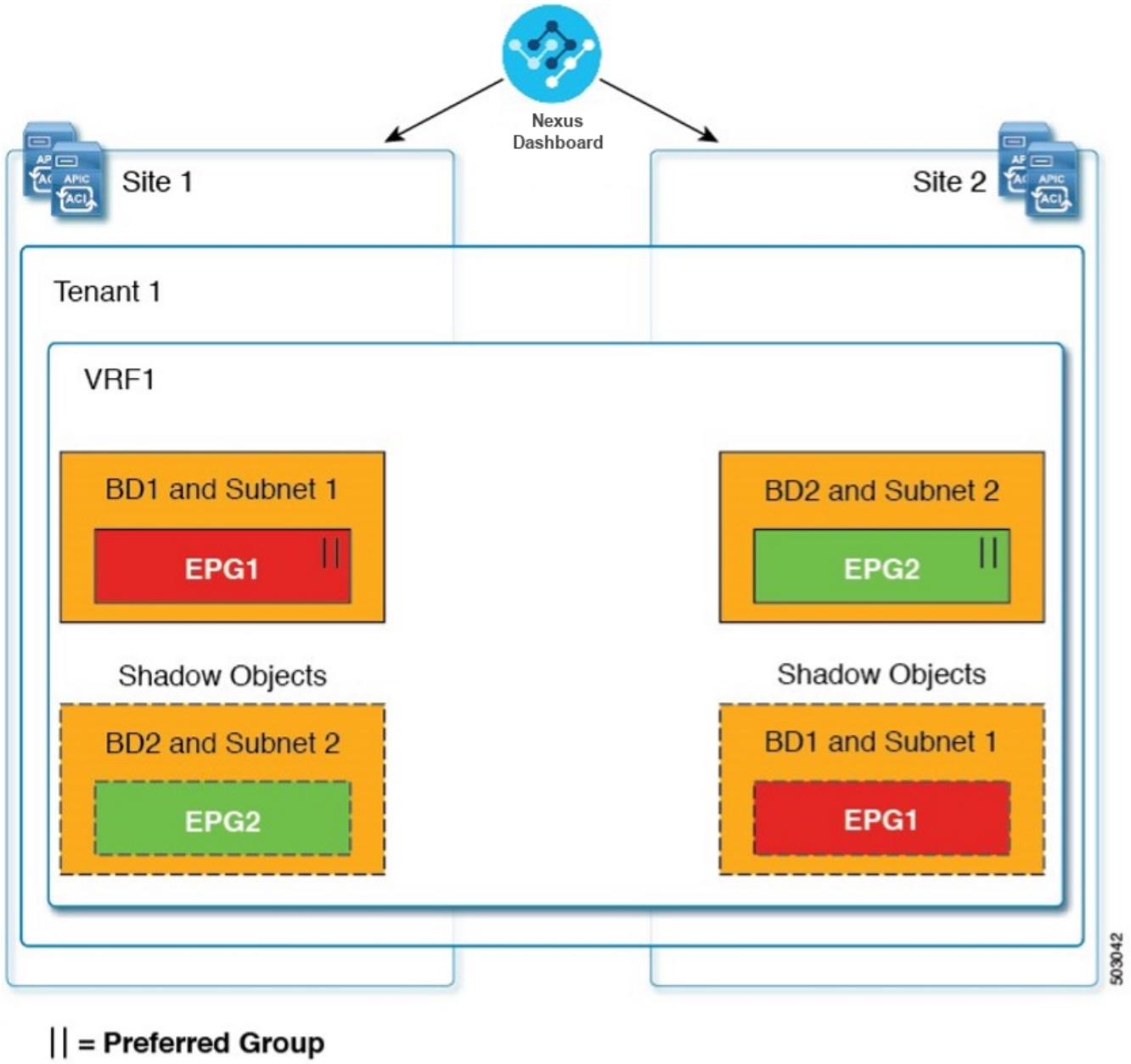
- VRFs

- Bridge Domains (BDs)
- L3Outs
- External EPGs
- Application Profiles
- Application EPGs
- Contracts

If your fabrics are running APIC release 5.0(2) or later, when you select a shadow object in the APIC GUI, you will see a This is a shadow object pushed by MSC to support inter-fabric policies. Do not make any changes or delete this object. warning at the top of main GUI pane. In addition, shadow EPGs that are not part of a VMM domain will not have static ports, while shadow BDs will have **No Default SVI Gateway** option enabled in the Nexus Dashboard GUI.

Other use cases with shadow objects

Shadow objects are also created in a number of other use cases, such as Preferred Group, vzAny, and Layer 3 Multicast.



In case of multicast, the shadow objects are created only for EPGs/BDs that have multicast sources connected and the option explicitly configured at the EPG level.

Hide shadow objects in the APIC GUI

Starting with APIC release 5.0(2), you can choose to show or hide the shadow objects created by the Nexus Dashboard in the on-premises fabric's APIC GUI.

If you want to hide shadow objects from the GUI, keep the following in mind:

- This option cannot be set globally from the Nexus Dashboard and must be set directly in each fabric's APIC as described in this section.
- The option to show shadow objects is turned off by default for all new APIC release 5.0(2) installations and upgrades, so previously visible objects may become hidden.
- Hiding shadow objects relies on a flag set by the Nexus Dashboard specifically for this feature. Shadow objects deployed by Nexus Dashboard will have the tag and can be hidden or shown using the APIC GUI setting.

- We recommend upgrading each fabric to APIC Release 5.0(2) before upgrading the Nexus Dashboard.

  Any objects deployed to fabrics running APIC Release 5.0(2) or later will be tagged with appropriate tags and can be shown or hidden using the APIC GUI without having to re-deploy them.

  If you upgrade the Orchestrator before the fabric's APIC, the fabric's objects will not be tagged and you will need to manually re-deploy the configuration after the fabric is upgraded for the flag to be set.

- If you ever downgrade your fabric to a release prior to Release 5.0(2), the shadow objects will no longer be hidden and you may see a different icon for them in the APIC GUI.
  1. Log in to the fabric's APIC.
  2. In the top right corner, click the **Manage my profile** icon and choose **Settings**.
  3. In the **Application Settings** window, enable or disable the **Show Hidden Policies** checkbox.

     The setting is stored in the user profile and is enable or disabled separately for each user.

  4. Repeat the process for any additional ACI fabrics.

Create schemas and templates

Before you begin:

- You must have at least one available tenant that you want to incorporate into your fabric.

  For more information, see Tenants and Tenant Policies Templates.

- You can create application profiles with the same name in different templates. However, Nexus Dashboard determines uniqueness using the APIC distinguished name (DN). For example, for a VRF, the DN uni/tn-<tenant>/ctx-<vrf-name> should be unique across all templates.

  The validation scope depends on the template type:

  - Multi-Fabric (stretched) templates: Nexus Dashboard checks DNs across all fabrics. The system rejects duplicate names under the same tenant, even when the templates are associated with different fabrics.
  - Autonomous templates: Nexus Dashboard checks names per fabric. The system allows duplicate names if the templates target separate fabrics; however, it rejects duplicate names if fabric associations overlap.

    Configuration examples:

  - Example 1: Multi-Fabric (stretched) templates: If Template A is associated with Fabric A and Fabric B, and Template B is associated with Fabric C and Fabric D, Nexus Dashboard rejects a VRF with the same name under the same tenant in both templates. This occurs because the VRF DN is identical, even though the fabric associations do not overlap.
  - Example 2: Autonomous templates: If Template A is associated with Fabric A and Template B is associated with Fabric B, Nexus Dashboard allows the same VRF name under the same tenant. If both templates are associated with the same fabric, Nexus Dashboard rejects the duplicate name.

Follow these steps to create a schema and template in the Nexus Dashboard.

1. Navigate to the **Orchestration** page.

   Manage > Orchestration

2. Create a new schema.
   a. Choose **Tenant Templates > Applications**.
   b. Click **Add Schema**.
   c. In the schema creation dialog, provide the **Name** and optional description for the schema and click **Add**.

      By default, the new schema is empty, so you must add one or more templates.

3. Create a template.
   a. In the schema page, click **Create New Template**.
   b. In the **Select a Template type** page, choose ACI Multi-Fabric and click **Next**.

- **ACI Multi-Fabric**—Templates that are used for Cisco ACI on-premises, which allow template and object stretching between multiple fabrics. This template supports two deployment types:

  - Multi-Fabric—The template can be associated to a single fabric (fabric-local policies) or to multiple fabrics (stretched policies) and the option should be selected for Multi-Fabric Network (ISN) or VXLAN inter-fabric communication to allow template and object stretching between multiple fabrics.
  - Autonomous—The template can be associated to one or more fabrics that are operated independently and are not connected through an inter-fabric Network (no inter-fabric VXLAN communication).

    Because autonomous fabrics are by definition that is isolated and do not have any inter-fabric connectivity, there is no shadow object configuration across fabrics and no cross-programming of pctags or VNIDs in the spine switches for inter-fabric traffic flow.

    The autonomous templates also allow for higher deployment scale.

  The following sections focus primarily on this type of templates.

c. In the right sidebar, provide the **Display Name** for the template.

d. (Optional) Provide a **Description**.

e. From the **Select a Tenant** drop-down, select the Tenant for this template.

   Keep in mind, the user account you're using to create a new schema must be associated with the tenant you are trying to add to it, otherwise the tenant will not be available in the drop-down list. Associating a user account with a tenant is described in Tenants and Tenant Policies Templates.

f. In the template view page, click **Save**.

   You must save the template after this initial configuration for extra options (such as fabric association) to become available.

g. Repeat this step to create any additional templates.

   For more information on schema and template design, see Schema and Template Design Considerations.

4. Assign the templates to fabrics.

   You deploy fabric configuration by deploying one template at a time to one or more fabrics. So you must associate the template with at least one fabric where you want to deploy the configuration.

   a. In the template view page, click **Actions** and choose **Add/Remove Fabrics**.

   b. In the **Add/Remove Fabrics <template>** dialog, select one or more fabrics where you want to deploy the template and click **Ok**.

What to do next:

After you have created a schema and one or more templates, you can proceed with editing the

templates as described in the following sections of this document based on your specific use cases. After you finish defining configurations, you can deploy the templates as described in Deploying Out of Sync Templates.

Importing schema elements from ACI fabrics

These sections describe how to import schema elements from ACI fabrics.

- Re-importing previously Orchestrator-managed configurations from ACI fabrics
- Import schema elements from ACI fabrics

Re-importing previously Orchestrator-managed configurations from ACI fabrics

This guidance applies when Nexus Dashboard Orchestrator is re-importing configurations from ACI fabrics that were previously managed from Nexus Dashboard Orchestrator (NDO). This scenario may occur when a Nexus Dashboard cluster is rebuilt and the orchestrator templates are being reconstructed from existing ACI configurations.

These sections explain how application-template configurations are represented and why configurations must be imported from every fabric before the first deployment.

- Application-template configuration levels
- Import behavior
- Re-importing previously NDO-managed configurations
- Required re-import procedure

Application-template configuration levels

An application template contains two types of configurations:

- **Template-level configuration** applies to all fabrics associated with the application template. NDO manages this as a common configuration and deploys it to each associated fabric.
- **Fabric-level configuration** is specific to an individual fabric associated with the application template. For example, EPG domain associations and static path bindings may be different for each fabric.

If an application template is associated with two fabrics, each fabric can have its own fabric-level configuration, while the template-level configuration remains common to both fabrics.

Import behavior

When a configuration is imported from an ACI fabric into an application template, NDO places the imported configuration at the template level or fabric level based on the object type and the portion of the configuration being imported.

A fabric-level configuration imported from one ACI fabric is specific to that fabric. NDO does not deploy it to other fabrics associated with the application template.

Re-importing previously NDO-managed configurations

A special case occurs when the ACI configuration was previously deployed and managed by NDO. These APIC objects retain NDO ownership annotations, and NDO treats them as NDO-managed objects.

When a configuration is re-imported into a newly reconstructed application template, the fabric-level configuration is populated only for the fabric selected for the import. The fabric-level configuration for other associated fabrics remains empty unless it is imported separately.

If an annotated fabric-level object exists on an ACI fabric but is missing from the fabric-level configuration for that fabric in the NDO template, NDO interprets the missing object as an intent to remove it.



When the template is deployed, NDO deletes the object from APIC, which could impact traffic.

Required re-import procedure

When reconstructing an application template that was a previously managed configuration in ACI, import the configuration from every fabric associated with the template before deploying it. This ensures that the fabric-level configuration for each fabric is represented in the NDO template and prevents NDO from interpreting previously managed objects as deleted intent.

Import schema elements from ACI fabrics

Before you begin:

You can create new objects and push them out to one or more fabrics or you can import existing fabric-local objects and manage them using the Nexus Dashboard. This section describes how to import one or more existing objects, while creating new objects is described later on in this document.

When importing policies from APIC into Nexus Dashboard, the common practice is to import some objects, such as VRFs or contracts, into a stretched template and other objects, such as non-stretched EPGs or BDs, into fabric-local templates. In autonomous templates, you can also import policies that have the same name from different fabrics.

Beginning with Nexus Dashboard 4.3.1, Nexus Dashboard supports importing same-name objects from different fabrics into autonomous templates. In autonomous templates, related references are resolved by matching the fabric from which the object is being imported. If an object with the same name already exists in Nexus Dashboard from another fabric, that object is not used to satisfy the relationship.

This change applies to autonomous templates only. Import behavior for multi-fabric templates remains unchanged. In multi-fabric templates, stretched objects continue to be managed as a single object across the associated fabrics in Nexus Dashboard.

For example, if a referenced object already existed in Nexus Dashboard and a new object was imported with the **Include Relations** option enabled, Nexus Dashboard could resolve the relationship to the existing same-named object instead of the object from the fabric currently being imported.

- Beginning with Nexus Dashboard 4.3.1, when you import an object into an autonomous template,

Nexus Dashboard resolves its related references by matching the fabric from which the object is being imported.

- As a result, if an object with the same name exists in a different fabric, Nexus Dashboard will not use that object to satisfy the relationship; only objects from the same source fabric are considered.



Nexus Dashboard used to support importing **copy** VRFs from the **common** tenant. Beginning with Nexus Dashboard 4.1.1, these VRFs may not appear as available options during import from ACI fabrics, depending on the release image.

To import one or more objects from fabrics:

1. Open the **Schema** where you want to import objects.
2. In the left sidebar, choose the **Template** where you want to import objects.
3. In the main pane click the **Import** button and choose the **Fabric** from which you want to import.
4. In the **Import from <fabric-name>** page that opens, choose one or more objects.



In autonomous templates, you can import objects that have the same name in different fabrics. During import, Nexus Dashboard matches referenced objects by fabric so that a same-named object in another fabric is not used to satisfy the relationship.

5. (Optional) Enable the **Import Relations** knob to import all related objects.

   For example, when importing a BD, enabling the **Import Relations** knob imports the associated VRF as well. If a VRF with the same name exists in more than one fabric, Nexus Dashboard resolves the BD to the VRF from the same fabric as the BD being imported.



As described previously, the **Import Relations** knob is enabled by default and cannot be disabled for objects whose related objects already exist in Nexus Dashboard. In autonomous templates, those relationships are resolved by using the matching fabric context for the imported object.

   Depending on the object type, additional objects may still appear in selection lists. However, the imported relationship is resolved by using the correct fabric match for the object being imported.

   This behavior applies to imported policy references in autonomous templates.

   Example

   Current state: A VRF named VRF1 has already been imported from Fabric A into an autonomous template in Nexus Dashboard.

   Action: You then import a BD from Fabric B that references a VRF with the same name, VRF1.

   Result: Nexus Dashboard resolves the BD to VRF1 from Fabric B. If VRF1 does not exist in Fabric B, Nexus Dashboard does not use VRF1 from Fabric A to satisfy the relationship.

6. Click **Import**.

Configure VRFs

Before you begin:

You must have the schema and template created and a tenant assigned to the template, as described in Create schemas and templates.

This section describes how to create a VRF.

1. Select the schema and template where you want to create the VRF.
2. Create the VRF.
   a. In the main pane, select **Create Object > VRF**.

      Alternatively, you can scroll down to the **VRFs** area and click **Create VRF**.

   b. In the properties pane, provide the **Display Name** for the VRF.
   c. (Optional) Provide a **Description**.
3. (Optional) Add one or more **Annotations**.

   This allows you to add arbitrary key:value pairs of metadata to an object as annotations (tagAnnotation). Annotations are provided for any custom purposes you may require, such as descriptions, markers for personal scripting or API calls, or flags for monitoring tools or orchestration applications such as your Nexus Dashboard. Because APIC ignores these annotations and merely stores them with other object data, there are no format or content restrictions imposed by APIC.

4. Configure the **On-Premises Properties** for the VRF.
   a. Specify **Policy Control Enforcement Preference**.

      Note that you cannot change the Policy Control Enforcement for newly created VRFs and the setting is locked to the enforced mode.

      However, you can use this to transition any VRF that you import from an ACI fabric that is configured as unenforced to the enforced mode after importing it. A typical use case is for brown field deployments where existing VRFs must be converted to enforced mode to support stretching them between fabrics. Once you have transitioned an imported VRF from unenforced to enforced in Nexus Dashboard, you will not be able to make further changes to this field.

      - Enforced-Security rules (contracts) will be enforced.
      - Unenforced-Security rules (contracts) will not be enforced.

   b. (Optional) Enable **IP Data-Plane Learning**.

      Defines if IP addresses are learned through data-plane packets for the VRF.

      When disabled, IP addresses are not learned from the data-plane packets. Local and remote MAC addresses are still learned, but local IP addresses are not learned from data packets.

      Regardless of whether this parameter is enabled or disabled, local IP addresses can still be learned from ARP, GARP, and ND.

c. (Optional) Enable **L3 Multicast** for the VRF.

   For additional information, see Layer 3 Multicast.

d. (Optional) Enable **vzAny** for the VRF.

   For additional information, see vzAny Contracts.

e. (Optional) Enable **Preferred Group** for the VRF.

   For additional information, see EPG Preferred Groups Overview and Limitations.

f. (Optional) Enable **BD Enforcement Status** for the VRF.

   By default, servers from an EPG of a given bridge domain can ping the SVI (subnet) of another bridge domain. If you wish to constrain a host to be able to ping only the SVI of the bridge domain to which it belongs, you can enable this BD Enforcement Status option configuration on the VRF. This blocks ICMP, TCP, and UDP traffic to the subnet IP address of bridge domains that are different from the one to which the server belongs.

Configure bridge domains

Before you begin:

- You must have the schema and template that is created and a tenant that is assigned to the template, as described in Create schemas and templates.
- You must have the VRF created as described in Configure VRFs.

This section describes how to configure a Bridge Domain (BD).

1. Select the schema and template where you want to create the bridge domain.
2. Create a bridge domain.
   a. In the main pane, select **+Create Object > Bridge Domain**.

      Alternatively, you can scroll down to the **Bridge Domains** area and click **Create Bridge Domain**.

   b. In the properties pane, provide the **Display Name** for the bridge domain.
   c. (Optional) Provide a **Description**.
3. (Optional) Add one or more **Annotations**.

   This allows you to add arbitrary key:value pairs of metadata to an object as annotations (tagAnnotation). Annotations are provided for any custom purposes that you may require, such as descriptions, markers for personal scripting or API calls, or flags for monitoring tools or orchestration applications such as your Cisco Nexus Dashboard. Because APIC ignores these annotations and merely stores them with other object data, there are no format or content restrictions that are imposed by APIC.

4. Configure **On-Premises Properties**.
   a. From the **Virtual Routing & Forwarding** drop-down, select the VRF for this BD.
   b. (Optional) Enable **L2 Stretch**.

c. (Optional) Enable **Inter-Fabric BUM Traffic Allow**.

   This option becomes available if you enabled **L2 Stretch**.

d. (Optional) Enable **Optimized WAN Bandwidth**.

   This option becomes available if you enabled **L2 Stretch**.

e. (Optional) Enable **Unicast Routing**.

   If this setting is enabled and a subnet address is configured, the fabric provides the default gateway function and routes the traffic. Enabling unicast routing also instructs the mapping database to learn the endpoint IP-to-VTEP mapping for this bridge domain. The IP learning is not dependent upon having a subnet that is configured under the bridge domain.

f. (Optional) Enable **L3 Multicast** for the BD.

   For additional information about Layer 3 multicast, see Layer 3 Multicast.

g. (Optional) Choose **L2 Unknown Unicast** mode.

   By default, unicast traffic is flooded to all Layer two-ports. If enabled, unicast traffic flooding is blocked at a specific port, only permitting egress traffic with MAC addresses that are known to exist on the port. The method can be Flood or Hardware Proxy.

   When the BD has L2 Unknown Unicast set to Flood, if an endpoint is deleted the system deletes it from both the local leaf switches and the remote leaf switches where the BD is deployed, by selecting Clear Remote MAC Entries. Without this feature, the remote leaf switch continues to have this endpoint learned until the timer expires.

   Modifying the L2 Unknown Unicast setting causes traffic to bounce (go down and up) on interfaces to devices attached to EPGs associated with this bridge domain.

h. (Optional) Choose **Unknown Multicast Flooding** mode.

   This is applicable for IPv4 unknown multicast traffic and is the node forwarding parameter for Layer 3 unknown multicast destinations.

   - Flood (default)-Unknown IPv4 multicast traffic is flooded on all front panel ports that are attached with the EPGs associated with this bridge domain. Flooding is not restricted to only M-Router ports of the bridge domain.
   - Optimized Flood-Send the data only to M-router ports in the bridge domain.

i. (Optional) Choose **IPv6 Unknown Multicast Flooding** mode.

   This is applicable for IPv6 unknown multicast traffic and is the node forwarding parameter for Layer 3 unknown multicast destinations.

   - Flood (default)-Unknown IPv6 multicast traffic is flooded on all front panel ports that are attached with the EPGs associated with this bridge domain. Flooding is not restricted to only M-Router ports of the bridge domain.
   - Optimized Flood-Send the data only to M-router ports in the bridge domain.

j. (Optional) Choose **Multi-Destination Flooding** mode.

The multiple destination forwarding method for Layer 2 multicast and broadcast traffic.

- Flood in BD-Sends the data to all ports on the same bridge domain.
- Drop-Drops Packet. Never sends the data to any other ports.
- Flood in Encapsulation-Send the data to all the EPG ports with the same VLAN within the bridge domain, except for the protocol packets which are flooded to the entire bridge domain.



This mode is supported only when the **L2 Stretch** option is disabled and is not supported for BDs that are stretched across fabrics.

k. (Optional) Enable **ARP Flooding**.

Enables ARP flooding, so that the Layer 2 broadcast domain maps IP addresses to the MAC addresses. If flooding is disabled, unicast routing will be performed on the target IP address.

Enables ARP flooding, so that ARP request will be flooded inside the Layer 2 broadcast domain. If the BD is stretched across fabrics, enabling ARP flooding is only possible with enabling **Inter-Fabric BUM Traffic Allow**. When ARP flooding is disabled, the leaf switch receiving the ARP request from a locally connected endpoint forwards it directly to the remote leaf switch where the target endpoint of the ARP request is connected (if the IP for the remote endpoint is known in the endpoint table) or to the spines (if the IP for the remote endpoint is not known in the endpoint table).

If you set the **L2 Unknown Unicast** mode to Flood, the **ARP Flooding** cannot be disabled. If the **L2 Unknown Unicast** mode is set to Hardware Proxy, ARP flooding can be enabled or disabled.

l. (Optional) Provide **Virtual MAC Address**.

The BD virtual MAC address and the subnet virtual IP address must be the same for all ACI fabrics for that bridge domain. Multiple bridge domains can be configured to communicate across connected ACI fabrics. The virtual MAC address and the virtual IP address can be shared across bridge domains.



Virtual MAC along with virtual IP subnet should used only for migration of individual fabrics to Nexus Dashboard-managed multi-fabric fabric. When the migration is completed, these flags can be disabled.

5. Add one or more **Subnets** for the BD.

a. Click **+Add Subnet**.

An **Add New Subnet** window opens.

b. Enter the subnet's **Gateway IP** address and a **Description** for the subnet that you want to add.

c. If necessary, enable **Treat as virtual IP address** option.

This option along with the **Virtual MAC Address** on the BD can be used for migration scenarios from individual Common Pervasive Gateway configuration to Nexus Dashboard-

managed Multi-Fabric deployments.

d. Select the **Scope** for the subnet.

The network visibility of the subnet.

- Private to VRF-Prevents the subnet from being announced over L3Out toward an external network domain.
- Advertised Externally-The subnet can be announced through L3Out toward an external network domain.

e. (Optional) Enable **Shared Between VRFs**.

Shared between VRFs-The subnet can be shared with and exported to multiple contexts (VRFs) in the same tenant or across tenants as part of a shared service. An example of a shared service is a routed connection to an EPG present in another context (VRF) in a different tenant. This enables traffic to pass in both directions across contexts (VRFs). An EPG that provides a shared service must have its subnet that is configured under that EPG (not under a bridge domain), and its scope must be set to advertised externally, and shared between VRFs.

Shared subnets must be unique across the contexts (VRF) involved in the communication. When a subnet under an EPG provides a Layer 3 external network shared service, such a subnet must be globally unique within the entire ACI fabric.

f. Leave the **No Default SVI Gateway** option unchecked.

Enabling this option means that only the proxy route (subnet route to spine proxy) is programmed on the leaf switches and no SVI is created, which means SVI cannot be used as the gateway.

We recommend that SVI is created by the BD subnet as the gateway and the **No Default SVI Gateway** option is enabled on the EPG instead because EPG subnets should only be used for route leaking.

g. (Optional) Enable **Querier** option.

Enables IGMP Snooping on the subnet

h. (Optional) Enable **Primary** option to designate the subnet as primary.

There can be one primary IPv4 subnet and one primary IPv6 subnet.

i. Click **Save**.

6. (Optional) Enable **EP Move Detection Mode**.

Uses the information that is received with a Gratuitous Address Resolution Protocol (GARP) packet to update the endpoint table when a specific IP address that was previously associated to one MAC address (mac-a) gets associated to a different MAC address (mac-b). This applies to the specific scenario where the move occurs on the same interface.

Although Cisco ACI can detect MAC and IP address movement between leaf switch ports, leaf switches, bridge domains, and EPGs, it does not detect the movement of an IP address to a new MAC address if the new MAC address is from the same interface and same EPG as the old MAC

address.

When the GARP-based detection option is enabled, Cisco ACI triggers an endpoint move based on GARP packets if the move occurs on the same interface and same EPG. If a GARP packet comes from the same interface and same EPG, then endpoint learning is triggered only when Unicast Routing, ARP Flooding, and "GARP based detection" are all enabled for the bridge domain.



The validation introduced in the previous release, which required that GARP-based endpoint move detection be enabled when VRF Data Plane Learning (DPL) was disabled, has been removed. From Nexus Dashboard release 4.1.1 and later, tenant configurations can proceed without requiring changes to bridge domains, even when DPL is disabled.

7. (Optional) Add an **IGMP Interface Policy**.

   You can configure several Tenant Policy templates and associate them with policy objects. For more information, see the "Create tenant policy templates" section in Tenants and Tenant Policies Templates for ACI Fabrics.

8. (Optional) Add an **IGMP Snoop Policy**.

   You can configure several Tenant Policy templates and associate them with policy objects. For more information, see the "Create tenant policy templates" section in Tenants and Tenant Policies Templates for ACI Fabrics.

9. (Optional) Add an **MLD Snoop Policy**.

   You can configure several Tenant Policy templates and associate them with policy objects. For more information, see the "Create tenant policy templates" section in Tenants and Tenant Policies Templates for ACI Fabrics.

10. (Optional) Add a **DHCP Policy**.

    For additional information, see DHCP Relay.

11. Configure the bridge domain's fabric-local properties as necessary.

    In addition to the template-level configurations, you can also define one or more fabric-local properties for the bridge domain, as described in Configure a bridge domain's fabric-local properties.

Configure a bridge domain's fabric-local properties

Before you begin:

You must have:

- Created the bridge domain and configured its template-level properties, as described in Configure bridge domains.
- Assigned the template that contains the bridge domain to one or more fabrics.

In addition to the template-level properties you typically configure for the object when you create it in

a template, you can also define one or more properties that are specific to each fabric to which you assign the template.

When you deploy the object to more than 1 fabric, the same template-level configurations are deployed to all fabrics, while the fabric-local configurations are deployed to those specific fabrics only.

1. Open the schema that contains the template with the bridge domain.
2. In the left sidebar, select the template that contains the bridge domain under the specific fabric that you want to configure.
3. In the main pane, select the bridge domain.

   For most fields, you see the values that you have configured at the template level, which you cannot edit here.

4. Click **+L3Out** to add an L3Out.

   This 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. In this case, you would also need to configure the subnet with the Advertised Externally flag. For more information, see the Inter-Fabric L3Out.

5. Enable **Host Route**.

   This enables Host-Based Routing on the bridge domain. When this knob is enabled, the border leaf switches will also advertise individual endpoint (EP) host-routes (/32 or /128 prefixes) along with the subnet. The host-route information is advertised only if the host is connected to the local Pod. If the EP is moved away from the local Pod or when the EP is removed from EP database, the route advertisement is then withdrawn.

6. If necessary, change the **SVI MAC Address**.

   The SVI MAC addresses must be unique per fabric, when virtual MAC and virtual IP are enabled for Common Pervasive Gateway (CPG) scenario. This field can also be used when CPG is not enabled, which will change the default router MAC of the BD.

7. Add one or more **Subnets** for the BD.

   The concept is the same as adding subnets to the BD at the template level, except the subnets will be configured for the bridge domain on this specific fabric only.

   a. Click **+Add Subnet**.

      An **Add New Subnet** window opens.

   b. Enter the subnet's **Gateway IP** address and a **Description** for the subnet that you want to add.

   c. Select the **Scope** for the subnet.

      The network visibility of the subnet.

      - Private to VRF-The subnet applies only within its tenant.
      - Advertised Externally-The subnet can be exported to a routed connection.

d. (Optional) Enable **Shared Between VRFs**.

   Shared between VRFs–The subnet can be shared with and exported to multiple contexts (VRFs) in the same tenant or across tenants as part of a shared service. An example of a shared service is a routed connection to an EPG present in another context (VRF) in a different tenant. This enables traffic to pass in both directions across contexts (VRFs). An EPG that provides a shared service must have its subnet that is configured under that EPG (not under a bridge domain), and its scope must be set to advertised externally, and shared between VRFs.

   Shared subnets must be unique across the contexts (VRF) involved in the communication. When a subnet under an EPG provides a Layer 3 external network shared service, such a subnet must be globally unique within the entire ACI fabric.

e. (Optional) Enable **No Default SVI Gateway**.

   Enabling this option means that only the proxy route (subnet route to spine proxy) is programmed on the leaf switches and no SVI is created, which means SVI cannot be used as the gateway.

   We recommend that SVI is created by the BD subnet as the gateway and the **No Default SVI Gateway** option is enabled on the EPG instead because EPG subnets should only be used for route leaking.

f. (Optional) Enable **Querier**.

   Enables IGMP Snooping on the subnet

g. (Optional) Enable **Primary** option to designate the subnet as primary.

   There can be one primary IPv4 subnet and one primary IPv6 subnet.

h. Click **Save**.

Configure application profiles and EPGs

Before you begin:

You must have the schema and template that is created and a tenant that is assigned to the template, as described in Create schemas and templates.

This section also assumes you have a Contract and a Bridge Domain created.

This section describes how to configure an Application Profile and an EPG.

1. Select the schema and template where you want to create the application profile.
2. Create an application profile.
   a. In the main pane, select **+Create Object > Application Profile**.

      Alternatively, you can scroll down to the **Application Profile** area and click **Create Application Profile**.

   b. In the right pane, provide the **Display Name** for the application profile.

You can create application profiles with the same name in different templates without any conflicts.

c. (Optional) Provide a **Description**.

3. Create an EPG.

a. In the main pane, select **+Create ObjectEPG**, then select the application profile where you want to create the EPG.

Alternatively, you can scroll down to the specific **Application Profile** area and click **Create EPG**.

b. In the right pane, provide the **Display Name** for the EPG.

c. (Optional) Provide a **Description**.

4. (Optional) Add one or more **Annotations** for the EPG.

This allows you to add arbitrary key:value pairs of metadata to an object as annotations (tagAnnotation). Annotations are provided for any custom purposes that you may require, such as descriptions, markers for personal scripting or API calls, or flags for monitoring tools or orchestration applications such as your Cisco Nexus Dashboard. Because APIC ignores these annotations and merely stores them with other object data, there are no format or content restrictions that are imposed by APIC.

5. Add a **Contract** for the EPG.

Creating contracts and filters is described in detail in Configure contracts and filters. If you already have a contract that is created:

a. Click **Add Contract**.

b. On the **Add Contract** dialog, enter the contract name and type.

c. Click **SAVE**.

6. (Optional) Add an **Intra-EPG Contract** for the EPG.

By default, communication between endpoints in an EPG is open, unless you enable Intra-EPG isolation under the EPG policy configuration.

With an intra-EPG contract, you can specify which traffic is allowed within an EPG based on protocol, ports, and other options specified by the contract's filters.

a. In the **Intra-EPG Contract** area, click **Add Contract**.

b. On the **Add Contract** dialog, enter the contract name and type.

c. Click **SAVE**.

7. From the **Bridge Domain** drop-down, select the bridge domain for this EPG.

If you are configuring an on-premises EPG, you must associate it with a bridge domain.

8. (Optional) Click **+ Subnet** to add a subnet to your EPG.

You may choose to configure a subnet on the EPG level rather than the bridge domain level, for example for a VRF route-leaking use-case.

a. On the **Add Subnet** dialog, enter the **Gateway IP** address and a description for the subnet you plan to add.

b. In the **Scope** field select either **Private to VRF** or **Advertised Externally**.

c. Click the check box for **Shared Between VRFs** if appropriate.

d. Click the check box for **No Default SVI Gateway** if appropriate.

e. Click **OK**.

9. (Optional) Enable microsegmentation.

   If you are configuring a microsegmentation EPG (uSeg), you must provide one or more uSeg attributes for matching endpoints to the EPG.

   a. Check the **uSeg EPG** check box.

   b. Click **+uSeg Attribute**.

   c. Provide the **Name** and **Type** for the uSeg attribute.

   d. Based on the attribute type you have selected, provide the attribute details.

      For example, if you have selected MAC for the attribute type, provide the MAC address to identify an endpoint in this EPG.

   e. Click **SAVE**.

10. (Optional) Enable intra-EPG isolation.

    By default, endpoints in EPG can freely communicate with each other. If you want to isolate the endpoints from each other, set the isolation mode to **Enforced**.

    intra-EPG endpoint isolation policies provide full isolation for virtual or physical endpoints; no communication is allowed between endpoints in an EPG that is operating with isolation enforced. Isolation-enforced EPGs reduce the number of EPG encapsulations required when many clients access a Common Service but are not allowed to communicate with each other.

11. (Optional) Enable Layer 3 multicast for the EPG.

    For additional information about Layer 3 multicast, see Layer 3 Multicast.

12. (Optional) Enable preferred group membership for the EPG.

    The Preferred Group feature allows you to include multiple EPGs within a single VRF to allow full communication between them with no need for contracts to be created. For additional information about EPG preferred group, see the "EPG Preferred Groups Overview and Limitations" section Preferred Group for ACI Fabrics.

13. Configure the EPG's fabric-local properties as necessary.

    In addition to the template-level configurations, you can also define one or more fabric-local properties for the EPG, as described in Configure an EPG's fabric-local properties.

Configure an EPG's fabric-local properties

Before you begin:

You must have:

- Created the application profile and EPG and configured the template-level properties, as described in Configure application profiles and EPGs.
- Assigned the template that contains the EPG to one or more fabrics.

In addition to the template-level properties you typically configure for the object when you create it in a template, you can also define one or more properties that are specific to each fabric to which you assign the template.

When you deploy the object to more than 1 fabric, the same template-level configurations are deployed to all fabrics, while the fabric-local configurations are deployed to those specific fabrics only.

1. Open the schema that contains the template with the EPG.
2. From the **View <Overview>** drop-down in the schema view, select the template that contains the EPG.
3. In the template view's main pane, click the **<fabric-name>** tab to select fabric-specific properties for the template.
4. In the main pane, click the EPG for which you want to update fabric-local properties.

   This opens the EPG's properties pane. For most fields, you see the values you have configured at the template level, which you cannot edit here.

5. Choose the **EPG Admin State**.

   This field is available only if the EPG belongs to a tenant other than infra or mgmt.

   When the EPG is in shutdown mode, the ACI policy configuration that is related to the EPG is removed from all the switches in the fabric. While the EPG still exists in the ACI Data Store, it is in inactive mode.

6. Add one or more **Subnets** for the EPG.
   a. Click **+Add Subnet**.

      An **Add New Subnet** window opens.

   b. Enter the subnet's **Gateway IP** address and a description for the subnet that you want to add.
   c. Select the **Scope** for the subnet.

      The network visibility of the subnet.

      - Private to VRF-Prevents the subnet from being announced through L3Out toward an external network domain.
      - Advertised Externally-The subnet can be announced through L3Out toward an external network domain.

   d. (Optional) Enable **Shared Between VRFs**.

      Shared between VRFs-The subnet can be shared with and exported to multiple contexts (VRFs) in the same tenant or across tenants as part of a shared service. An example of a

shared service is a routed connection to an EPG present in another context (VRF) in a different tenant. This enables traffic to pass in both directions across contexts (VRFs). An EPG that provides a shared service must have its subnet that is configured under the BD (not under the EPG), and its scope must be set to advertised externally, and shared between VRFs.

Shared subnets must be unique across the contexts (VRF) involved in the communication. When a subnet under an EPG provides a Layer 3 external network shared service, such a subnet must be globally unique within the entire ACI fabric.

e. (Optional) Enable **No Default SVI Gateway**.

Enabling this option means that only the proxy route (subnet route to spine proxy) is programmed on the leaf switches and no SVI is created, which means SVI cannot be used as the gateway.

We recommend enabling this option on the EPG subnets, which should only be used for route leaking and leaving this option disabled on the BD subnets so that the SVI can be used as a gateway.

f. Click **Ok** to save.

7. Add one or more **Static ports**.

a. Click **+Static Port**.

b. From the **Path Type** drop-down, select the type of port.

c. If configuring a physical interface, select the **Pod**

d. Choose whether you want to configure a single port or a range of ports.

For the interface configuration, you have an option to do it either by entering a single **Leaf** and a **Path** or by entering a range of **Leaf** for example, 120-125 and **Path** eg1/17-20. You will also have an option to enter a range of **Leaf** and associate it with one single **Path**, or enter a range of **Path** for one single **Leaf**.

However, after the configuration it will still be displayed as individual ports in the UI and will require individual changes for any future updates.

e. Select the **Port Encap VLAN**.

When manually configuring the port encap on a domain for an EPG, the VLAN ID must belong to a static VLAN block within a dynamic VLAN pool.

If EPG is enabled for microsegmentation at the template level, when a **Primary MICRO-SEG VLAN** is configured, the **Port Encap VLAN** is configured as an Isolated Secondary VLAN for the Primary VLAN. Traffic is sent from the host to the leaf switch using the secondary VLAN and return traffic from the leaf switch to the host is sent using the primary VLAN.

f. (Optional) Select the **Primary MICRO-SEG VLAN**.

The VLAN identifier for microsegmentation.

g. (Optional) Select the **Deployment Immediacy**.

When policies are downloaded to the leaf nodes, deployment immediacy can specify when

the policy is pushed into the hardware policy CAM:

- Immediate-Specifies that the policy is programmed in the hardware policy CAM when the policy is downloaded in the leaf switch software.
- On Demand-Specifies that the policy is programmed in the hardware policy CAM only when the first packet is received through the data path. This process helps to optimize the hardware space.

h. (Optional) Select the **Mode**.

The mode of the static association with the path. EPG tagging sees configuring a static path under an EPG:

- Trunk-The default deployment mode. Select this mode if the traffic from the host is tagged with a VLAN ID.
- Access (802.1P)-Select this mode if the traffic from the host is tagged with a 802.1p tag. When an access port is configured with a single EPG in built-in 802.1p mode, its packets exit that port untagged. When an access port is configured with multiple EPGs, one in built-in 802.1p mode, and some with VLAN tags, all packets exiting that access port are tagged VLAN 0 for EPG configured in built-in 802.1p mode and for all other EPGs packets exit with their respective VLAN tags. Only one built-in 802.1p EPG is allowed per access port.
- Access (Untagged)-Select this mode if the traffic from the host is untagged (without VLAN ID). When a leaf switch is configured for an EPG to be untagged, for every port, this EPG uses, the packets exit the switch untagged. Note that when an EPG is deployed as untagged, do not deploy that EPG as tagged on other ports of the same switch.

8. Add one or more **Static Leaf** nodes.
   a. Click **+Static Leaf**.
   b. From the **Leaf** drop-down, select the leaf node that you want to add.
   c. (Optional) In the **VLAN** field, provide the VLAN ID for tagged traffic.
9. Add one or more **Domains**.
   a. Click **+Domain**.
   b. Select the **Domain Association Type**.

   This is the type of the domain that you are adding:

   - VMM
   - Fibre Channel
   - L2 External
   - L3 External
   - Physical

   c. Select the **Domain Profile** name.
   d. Select the **Deployment Immediacy**.

   Deployment immediacy can specify when the policy is pushed:

- Immediate–Specifies that the policy is programmed in the hardware policy CAM when the policy is downloaded in the leaf switch software.
- On Demand–Specifies that the policy is programmed in the hardware policy CAM only when the first packet is received through the data path. This process helps to optimize the hardware space.

e. Select the **Resolution Immediacy**.

Specifies whether policies are resolved immediately or when needed. The options are:

- Immediate–Specifies that EPG policies are pushed to the leaf switch nodes upon hypervisor attachment to the VMware vSphere Distributed Switch (VDS). LLDP or OpFlex permissions are used to resolve the hypervisor to leaf node attachments.
- On Demand–Specifies that EPG policies are pushed to the leaf switch nodes only when a hypervisor is attached to VDS and a VM is placed in the port group (EPG).
- Pre-provision–Specifies that EPG policies are pushed to the leaf switch nodes even before a hypervisor is attached to the VDS. The download preprovisions the configuration on the switch.

f. For VMM domains, configure extra settings.

You can configure several extra properties for VMM domains directly from your Cisco Nexus Dashboard.

- **Port Bindings** – You can choose one of the following options:
  - Dynamic Binding
  - Ephemeral
  - Default
  - Static Binding

  For additional information about port binding, see the "Cisco ACI with VMware VDS Integration" chapter of the *Cisco ACI Virtualization Guide*
- **Netflow** – choose whether you want to enable NetFlow for the VMM domain.
- **Promiscuous Mode** – specifies whether to allow or reject unicast traffic that is not destined to the MAC addresses of the virtual machines attached to the trunk port group.
- **MAC Address Changes** – specifies whether to allow or reject MAC address changes for the network adapter within the VM.
- **Forged Transmits** – specifies whether to allow or reject forged transmits.

  A forged transmit occurs when a network adapter starts sending out traffic that identifies itself as something else. This security policy compares the effective address of the virtual network adapter and the source address inside an 802.3 Ethernet frame generated by the virtual machine to ensure that they match.
- **Custom EPG Name** – allows you to provide a custom name for the EPG associated with this VMM domain.

  When you associate an EPG to a VMM domain, APIC automatically creates a VMware

vCenter port group or a Microsoft VM network.), it is easier to manage the port groups or VM networks because you now have the option of giving the EPG a custom name

Configure contracts and filters

This section describes how to configure a contract, a filter, and assigns the filter to the contract. A filter is similar to an Access Control List (ACL), it is used to filter traffic through contracts that are associated to EPGs.

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

   You can create the contract in the same or different template as the objects (EPGs and external EPGs) to which you apply it. If the objects that use the contract are deployed to different fabrics, we recommend defining the contract in a template that is associated to multiple fabrics. However, this is not strictly required and even if the contract and filters are defined only as local objects in Fabric 1, Nexus Dashboard creates those objects in a remote Fabric 2 when a local EPG or external EPG in Fabric 2 must consume or provide that contract.

2. Create a filter.
   a. In the main pane, select **+Create Object > Filter**.

      Alternatively, you can scroll down to the **Filters** area, mouse over the tile, and click **Add Filter**.

   b. In the right pane, provide the **Display Name** for the filter.
   c. (Optional) Provide a **Description**.
3. (Optional) Add one or more **Annotations**.

   This allows you to add arbitrary key:value pairs of metadata to an object as annotations (tagAnnotation). Annotations are provided for any custom purposes that you may require, such as descriptions, markers for personal scripting or API calls, or flags for monitoring tools or orchestration applications such as your Cisco Nexus Dashboard. Because APIC ignores these annotations and merely stores them with other object data, there are no format or content restrictions that are imposed by APIC.

4. Create a filter entry.
   a. In the right pane, click **+ Add Entry**.

      The filter entry is a combination of network traffic classification properties. You can specify one or more options as described in the following step.

   b. Provide the **Name** for the filter.
   c. Choose the **Ether Type**.

      For example, ip.

   d. Choose the **IP Protocol**.

      For example, icmp.

   e. Choose the **Destination Port Range From** and **Destination Port Range To**.

The start and end of the destination ports range. You can define a single port by specifying the same value in both fields or you can define a range of ports from 0 to 65535. You can also choose to specify one of the server types instead of specific port numbers, for example http.

f. Enable **Match only fragments** option.

When enabled, the rule applies to any IP fragment with an offset that is greater than 0 (all IP fragments except the first). When disabled, the rule will not apply to IP fragments with an offset greater than 0 because TCP/UDP port information can only be checked in initial fragments.

g. Enable **Stateful** option.

When this option is enabled, any traffic coming from the provider back to the consumer will always have to have the ACK bit set in the packet or else the packets will be dropped.

h. Specify **ARP flag** (Address Resolution Protocol).

The **ARP Flag** is used when creating a specific filter for ARP and allows you to specify ARP request or ARP reply.

i. Choose the **Source Port Range From** and **Source Port Range To**.

The start and end of the source ports range. You can define a single port by specifying the same value in both fields or you can define a range of ports from 0 to 65535. You can also choose to specify one of the server types instead of specific port numbers, for example http.

j. Specify **TCP session rules**.

TCP session rules are used when creating a filter for TCP traffic and allow you to configure stateful ACL behavior.

k. Click **Ok** to save the filter.

l. Repeat this step to create any additional filter entries for this filter.

You can create and assign multiple filter entries for each filter.

5. Create a contract.

a. In the main pane, select **+Create Object > Contract**.

Alternatively, you can scroll down to the **Contract** area, mouse over the tile, and click **Add Contract**.

b. In the right pane, provide the **Display Name** for the contract.

c. (Optional) Provide a **Description**.

d. (Optional) Add one or more **Annotations**.

This allows you to add arbitrary key:value pairs of metadata to an object as annotations (tagAnnotation). Annotations are provided for any custom purposes that you may require, such as descriptions, markers for personal scripting or API calls, or flags for monitoring tools or orchestration applications such as your Cisco Nexus Dashboard. Because APIC ignores these annotations and merely stores them with other object data, there are no format or content

restrictions that are imposed by APIC.

e. Select the appropriate **Scope** for the contract.

Contract scope limits the contract's accessibility; the contract will not be applied to any consumer EPG outside the scope of the provider EPG:

- Application Profile
- VRF
- Tenant
- Global

f. Toggle the **Apply both directions** knob if you want the same filter to apply for both consumer-to-provider and provider-to-consumer directions.

If you enable this option, you must provide the filters only when and they apply for traffic in both directions. If you leave this option disabled, you must provide two sets of filter chains, one for each direction.



If you create and deploy a contract with **Apply both directions** enabled, you cannot simply disable the option and redeploy for the change to apply. To disable this option on an already deployed contract, you must delete the contract, deploy the template, then re-create the contract with the option that is disabled to correctly change the setting in your fabrics.

g. (Optional) From the **Service Graph** drop-down, select a service graph for this contract.

h. (Optional) From the **QoS Level** drop-down, select a value for this contract.

This value specifies the ACI QoS Level that will be assigned to the traffic using this contract. For more information, see QoS Preservation Across IPN.

If you leave this at Unspecified, the default QoS Level 3 is applied to the traffic.

6. Assign the filters to the contract.

a. In the main pane for template, select a contract. In the right pane, scroll down to the **Filter Chain** area and click **+ Add Filter** to add a filter to the contract.

b. In the **Add Filter Chain** window that opens, select the filter that you added in previous step from the **Name** drop-down list.

c. Select the **Action** for the filter.

When adding filters, you can choose whether to permit or deny traffic that matches the filter criteria. For deny filters, you can set the priority of the filter to one of four levels: default, low, medium, or high; the permit filters always have the default priority. For more information on ACI contracts and filters, see *Cisco ACI Contract Guide*.

d. Click **Ok** to add the filter to the contract.

e. If you disabled the Apply both directions option on the contract, repeat this step for the other filter chain.

f. (Optional) You can create and assign multiple Filters to each Contract.

If you want to create extra filter for the same contract:

- Repeat Step 2 and Step 3 to create another filter along with its filter entries.
- Then repeat this step to assign the new filter to this Contract.

View schemas

After you have created one or more schemas, they are displayed both on the Dashboard and the Schemas page.

You can use the functionality available on these two pages to monitor the usage and the health of your schemas when they are deployed. You can also access and edit specific areas of the implemented schema policies using the Cisco Nexus Dashboard GUI.

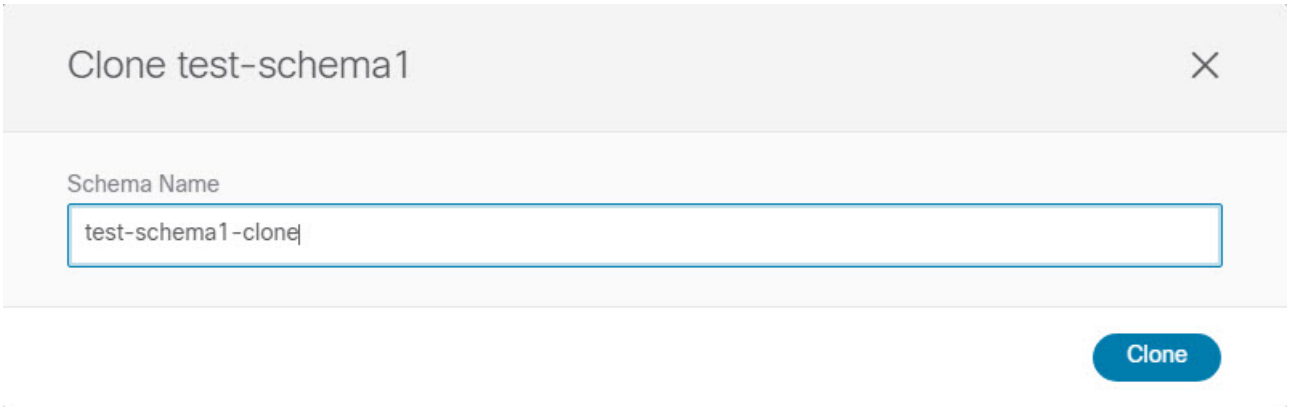
Clone schemas

This section describes how to create a copy of an existing schema and all its templates using the " Clone Schema" feature in the **Schemas** screen.

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

   Manage > Orchestration

2. Choose the schema to clone.
   a. Choose **Tenant Templates > Applications**.
   b. From the action menu **(...)** menu next to the name of the schema you want to clone, select **Clone**.
3. Provide the name for the new schema and click **Clone**.



Clone Schema

After you click **Clone**, the UI will display Cloning of <schema-name> was successful. message and the new schema will be listed in the **Schemas** screen.

The new schema is created with the exact same templates (and their tenants' association), object, and policy configurations as the original schema.

Note that while the templates, objects, and configurations are copied, the fabric association is not preserved and you must reassociate the template in the cloned schema with any fabrics where you want to deploy them. Similarly, you must provide any fabric-specific configurations for the template objects after you associate it with the fabrics.

4. (Optional) Verify that the schema and all its templates were copied.

   You can verify that the operation completed successfully by comparing the two schemas.

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