



CHAPTER 3

Creating a FlexUNI/EVC Policy

This chapter contains an overview of FlexUNI/EVC support in ISC, as well as the basic steps to create a FlexUNI/EVC policy. It contains the following sections:

- [Overview of FlexUNI/EVC Support in ISC, page 3-1](#)
- [Defining the FlexUNI/EVC Policy, page 3-5](#)
- [Setting the Service Options, page 3-7](#)
- [Setting the FlexUNI Attributes, page 3-9](#)
- [Setting the Interface Attributes, page 3-14](#)
- [Enabling Template Association, page 3-19](#)

For information on creating FlexUNI/EVC service requests, see [Chapter 4, “Managing a FlexUNI/EVC Service Request.”](#)



Note

For Ethernet (E-Line and E-LAN) services, use of the FlexUNI/EVC policy and service request is recommended. If you are provisioning services using the FlexUNI/EVC syntax, or plan to do so in the future, use the FlexUNI/EVC service. Existing services that have been provisioned using the L2VPN and VPLS service policy types are still supported and can be maintained with those service types. For ATM and FRoMPLS services, use the L2VPN service policy, as before.

Overview of FlexUNI/EVC Support in ISC

Flexible user network interface (FlexUNI) is a generic approach for creating Ethernet services in ISC. It can, if supported by the hardware, be used for all Ethernet provisioning. (For information on what platforms support FlexUNI/EVC see [Platform Support for FlexUNI/EVC in ISC 5.1, page 3-3](#).) The FlexUNI/EVC policy is flexible and generic and allows for service designers to provide greater service offerings than available through traditional ISC L2VPN and VPLS services.

Certain line cards have interfaces that support the Cisco IOS Ethernet Virtual Circuit (EVC) syntax. These interfaces can be configured with either EVC infrastructure features or with switch-port command-line interface commands (Class). FlexUNI optionally supports the EVC CLI syntax/infrastructure. For this reason, the FlexUNI policies and service requests are referred to by the umbrella term “FlexUNI/EVC.” However, it is important to note that FlexUNI/EVC policies and service request are not tied to the new EVC syntax. Service endpoints can use non-EVC syntax also.

Services leveraging the FlexUNI/EVC infrastructure are varied in nature, and there is not always a clear delineation between different services. This is because FlexUNI/EVC provides great flexibility in the way these services can be delivered. This can make it challenging to define the services. For example, a traditional ERS could be delivered in several ways by variations of the Class on the platform.

The FlexUNI/EVC policy and associated service request offer a generic and flexible service construct to support device capabilities. This policy is flexible enough to cater to different service offerings using the EVC architecture. It allows service designers to utilize most of the EVC features in a flexible manner, to match the hardware and platform capabilities.

The FlexUNI/EVC policy can be used to create only a FlexUNI/EVC service request and not any other existing ISC service request types, such as L2VPN, VPLS, and so on. Likewise, a FlexUNI/EVC service request can be created using only a FlexUNI/EVC policy and not any other existing ISC policies.

The FlexUNI/EVC infrastructure provides several benefits to Carrier Ethernet (CE) deployments, including:

- Flexible frame matching.
- Flexible VLAN tag manipulation and/or translation.
- Multiple services on the same port.
- Flexible service mapping.
- VLAN scaling and locally significant VLANs.

FlexUNI/EVC supports a variety of network configurations, such as the following:

- Provisioning of Ethernet access as a EVC-capable EWS interface on the N-PE.
- Interconnecting Ethernet accesses terminating on a single Cisco 7600 N-PE on one or multiple ports in a bridge domain.
- Interconnecting Ethernet accesses terminating on multiple Cisco 7600 N-PEs in a VPLS service.
- Services that combine the existing services with the Ethernet access, including the ERS/EWS inter-working service.
- Provisioning of E-Line services, in which one or both N-PE interfaces are FlexUNI.

FlexUNI/EVC Features

This section summarizes the features supported by the FlexUNI/EVC policy and service request in ISC:

- Choice of topology:
 - Customer edge device (CE) directly connected.
 - CE connected through Ethernet access devices.
- Choice of platforms:
 - FlexUNI/EVC on all N-PEs.
 - FlexUNI/EVC on none of the N-PEs.
 - Mix of FlexUNI/EVC and the old infrastructure. This allows both the old and new platforms to co-exist, in order to ensure continued support for deployed platforms.

- Choice of connectivity across the MPLS core (with or without bridge-domain):
 - Pseudo wires
 - VPLS
 - Local (local connects)
- Flexible VLAN handling mechanism that deals with up to two levels of VLAN tags:
 - VLAN matching for service classification. This provides the ability to match both outer and inner VLAN tags, or the ability to match a range of inner VLAN tags.
 - VLAN manipulations, such as pop outer tag, pop inner tag, push outer tag, push inner tag, and VLAN translations (1:1, 2:1, 1:2, 2:2).
- Flexible forwarding options:
 - Configure a pseudowire on the MPLS core directly under a service instance (for E-Line only).
 - Configure a pseudowire on the MPLS core under a switch virtual interface (SVI) by associating it to a bridge domain.



Note The appropriate VLAN manipulations are applicable to pseudowire in both cases.

- Associate traffic from different interfaces and/or VLANs onto a single bridge domain, with appropriate VLAN manipulations for VPLS.
- Associate traffic from different interfaces and/or VLANs onto a single bridge domain with appropriate VLAN manipulations for local connects.

Platform Support for FlexUNI/EVC in ISC 5.1

In ISC 5.1, the following platforms are supported for the FlexUNI/EVC service:

- IOS 12.2(33)SRB and SRC
- ES20 line cards (2x10GE and 20x1GE)
- Shared Port Adaptor (SPA) Interface Processor-400 (SIP-400) line cards, version 2.0 (2x1GE and 5x1GE)

The interfaces on the ES20 and SIP-400 line cards support the IOS EVC syntax.

Two example platform scenarios are covered in the next sections. Note that the UNI characteristics and the FlexUNI capabilities of the N-PE are not inter-dependent.

Example 1

FlexUNI/EVC service requests allow operators to add links with either a EVC-capable interface and/or the nonEVC-capable interface on the N-PE. For example, an operator can add three links to a FlexUNI/EVC service request (with VPLS connectivity) with the following configurations:

- Link one has a Cisco 67xx interface and an IOS 12.2 (33)SRB image on a Cisco 7600 N-PE.
- Link two has a Cisco 67xx interface and an IOS 12.2 (33)SRB image on a Cisco 7600 N-PE.
- Link three has an ES20-based interface and an IOS 12.2 (33)SRB image on a Cisco 7600 N-PE.

Example 2

As far as Layer 2 access nodes are concerned, configurations on the UNI/NNI of a U-PE and/or PE-AGG are not influenced by the FlexUNI/EVC capability on the N-PE. However, if a selected named physical circuit (NPC) with N-PE interface is configured with FlexUNI/EVC, it cannot be provisioned for traditional configuration. An error will be generated while saving the service request.

On the other hand, if a selected NPC with N-PE interface is configured without FlexUNI, it cannot be provisioned for FlexUNI configuration. An error will be generated while saving the service request.

For example, if for link one of a FlexUNI/EVC service request, if the encapsulation is selected as dot1Q, the interface can share other L2 ERS/VPLS ERMS UNIs on the same U-PE/PE-AGG.

If the N-PE interface that is part of the NPC being picked is already configured with non-FlexUNI/EVC features (using an existing L2VPN or VPLS service request), you cannot configure FlexUNI/EVC on it.



Note

If “Dot1Q Tunnel” is selected as the encapsulation type, the port cannot be shared with other services.

Device Roles with FlexUNI/EVC

Presently, ISC has U-PE, PE-AGG and N-PE devices. The basic PE device role association of ISC continues for FlexUNI/EVC policy and service requests. In this release of ISC, there are no changes made to the PE role assignment. A device having FlexUNI/EVC capabilities will not call for a change in the existing role assignment in ISC. However, FlexUNI/EVC capabilities in ISC are supported only for interfaces on N-PE and not on PE-AGG or U-PE devices.



Note

ISC does not support customer edge devices (CEs) for FlexUNI/EVC. If the access port contains any DSLAMS, non-Cisco Ethernet devices and/or other Cisco devices that are not supported by ISC, such nodes and beyond are not in the scope of ISC. In such cases, from the ISC perspective, the interface on the first ISC-managed device is the UNI.

Topology Overview for FlexUNI/EVC

This section provides examples of various topologies supported with FlexUNI/EVC. As mentioned in the note at the end of section [Device Roles with FlexUNI/EVC, page 3-4](#), ISC does not support customer edge devices (CEs) with FlexUNI/EVC. References to the term “CE” in the following topology variations (such as “CE directly connected” and so on) is only to indicate how the customer or third-party devices connect to the N-PE. For all the cases involving FlexUNI/EVC, the CE is not supported in ISC. Also, any provider device that is not supported by ISC, and which is used in the access circuit, marks the boundary for the scope of ISC, beyond which no devices (that is, towards the CE, and including the unsupported node) is managed by ISC.

CE Directly Connected and FlexUNI

With this combination, the UNI is the interface on a supported line card, with EVC capability configured. ISC does not configure ISC’s standard UNI functionality (for example, port-security, storm control, and Layer 2 Protocol Tunneling). This is because of lack of command support on the FlexUNI/EVC-capable hardware. Operators can use templates to configure relevant platform supported parameters to realize any of these features not provided by ISC. ISC configures only the service instance with VLAN

manipulations and pseudowire, VPLS, or local-connect on the UNI. NPCs are not needed while creating such links because NPCs are only required when there are access nodes between the N-PE and CE. Other intermediate Ethernet access nodes are not involved in this topology.

CE Directly Connected and No FlexUNI

This is similar to the UNI on N-PE case in ISC. The FlexUNI/EVC service request can be used to create such links with older Cisco 7600 platforms (that is, N-PE interfaces without FlexUNI/EVC capability), but with plans of adding one or more future links with EVC support. If not, one could use the existing ERS/EWS/ERMS/EMS functionality in ISC. NPCs are not needed while creating such links because NPCs are only required when there are access nodes between the N-PE and CE. Other intermediate Ethernet access nodes are not involved in this topology.

CE Not Directly Connected and FlexUNI

This topology involves the following configurations:

- UNI on a U-PE or PE-AGG to which the CE is connected.
- Ethernet U-PE and/or PE-AGGs.
- N-PE with FlexUNI-capable interface on the CE-facing side.

All service-specific parameters, such as port-security, L2 Protocol Tunneling, storm control, and so on, are applicable to the UNI (Standard UNI) in such links. The U-PE and/or PE-AGG configurations will also have no change in CLIs. However, the EVC commands are applicable only on the N-PE (on the CE-facing interface). NPCs are used while creating such links.

CE Not Directly Connected and No FlexUNI

This link is identical to an attachment circuit in existing ISC implementations. This has a standard UNI as in existing ISC services. NPCs are used while creating such links.

A Note on Checking of Configurations

ISC attempts to provision all configurations generated by a FlexUNI/EVC service request. ISC does not perform any prior checks to verify if the CLIs are compatible with the specific devices being provisioned. This is to ensure flexibility of support for device/platform features, which could change over time. Hence, it is important for the service designer or operator to carefully create the FlexUNI/EVC policies and service requests.

Defining the FlexUNI/EVC Policy

You must define a FlexUNI/EVC policy before you can provision a service. A policy can be shared by one or more service requests that have similar service requirements.

A policy is a template of most of the parameters needed to define a FlexUNI/EVC service request. After you define it, a FlexUNI/EVC policy can be used by all the FlexUNI/EVC service requests that share a common set of characteristics. You create a new FlexUNI/EVC policy whenever you create a new type of service or a service with different parameters. FlexUNI/EVC policy creation is normally performed by experienced network engineers.

An Editable check box in for an attribute in the policy gives the network operator the option of making a field editable. If the value is set to editable, the service request creator can the value(s) of the particular policy attribute. If the value is *not* set to editable, the service request creator cannot change the attribute.

You can also associate Cisco IP Solution Center (ISC) templates and data files with a service request. See [Appendix B, “Working with Templates and Data Files,”](#) for more about using templates and data files in service requests.

To define a FlexUNI/EVC policy, you start by setting the service type attributes. To do this, perform the following steps.

Step 1 Choose **Service Design > Policies**.

The Policies window appears.

Step 2 Click **Create**.

Step 3 Choose **FlexUNI (EVC) Policy**.

The EVC Policy Editor - Service Type window appears, as shown in [Figure 3-1](#).

Figure 3-1 EVC Policy Editor - Service Type

Attribute	Value
Policy Name *	<input type="text"/>
Policy Owner:	<input checked="" type="radio"/> Customer <input type="radio"/> Provider <input type="radio"/> Global Policy
Customer *	<input type="button" value="Select"/>
Policy Type:	ETHERNET

Note: * - Required Field

- Step 1 of 5 -

< Back Next > Finish Cancel

211664

Step 4 Enter a **Policy Name** for the FlexUNI/EVC policy.

Step 5 Choose the **Policy Owner** for the FlexUNI/EVC policy.

There are three types of FlexUNI/EVC policy ownership:

- Customer ownership
- Provider ownership
- Global ownership—Any service operator can make use of this policy.

This ownership has relevance when the ISC Role-Based Access Control (RBAC) comes into play. For example, a FlexUNI/EVC policy that is customer-owned can only be seen by operators who are allowed to work on this customer-owned policy. Similarly, operators who are allowed to work on a provider’s network can view, use, and deploy a particular provider-owned policy.

- Step 6** Click **Select** to choose the owner of the FlexUNI/EVC policy.
The policy owner was established when you created customers or providers during ISC setup. If the ownership is global, the Select function does not appear.
- Step 7** Click **Next**.
The EVC Policy Editor - Service Options window appears, as shown in [Figure 3-2](#).
- Step 8** Continue with the steps contained in the next section, [Setting the Service Options, page 3-7](#).

Setting the Service Options

This section describes how to set the service options for the FlexUNI/EVC policy, as shown in [Figure 3-2](#).

Figure 3-2 EVC Policy Editor - Service Options Window

Attribute	Value	Editable
CE Directly Connected To FlexUNI	<input type="checkbox"/>	
All Links Terminate On FlexUNI	<input type="checkbox"/>	
MPLS Core Connectivity Type	PSEUDOWIRE	
Configure With Bridge Domain	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Note: * - Required Field

- Step 2 of 5 -

< Back Next > Finish Cancel

211665

The **Editable** check box gives you the option of making a field editable. If you check the **Editable** check box, the service operator who is using this FlexUNI/EVC policy can modify the editable parameter during FlexUNI/EVC service request creation.

To set the FlexUNI/EVC service options, perform the following steps.

- Step 1** Check the **CE Directly Connected to FlexUNI** check box if the CEs are directly connected to the N-PE.
This check box is not checked by default. Usage notes:
- If the check box is checked, a service request created using this policy can have only directly connected links. No Ethernet access nodes will be involved.
 - If the check box is unchecked, a service request created using this policy might or might not have Ethernet access nodes in the links.

- When a CE is directly connected to the N-PE, NPCs are not applicable to the link while creating service requests.
- When a CE is not directly connected to the N-PE, NPCs are used during service request creation, as per standard ISC behavior. There is no change in NPC implementation to support FlexUNI/EVC functionality.

Step 2 Check the **All Links Terminate on FlexUNI** check box if all links need to be configured with FlexUNI/EVC features.

This check box is not checked by default. Usage notes:

- If the check box is checked, a service request created using such policy will have all links using the FlexUNI/EVC feature.
- If the check box is unchecked, zero or more links can use the FlexUNI/EVC feature. This ensures that existing platforms can still be used in one or more links while delivering the services. This allows the possibility of a link with FlexUNI/EVC support being added in the future.



Note If the check box is unchecked, in the service request creation process the user must indicate whether or not the created link is FlexUNI or non-FlexUNI.

- If no links are expected to use the FlexUNI/EVC feature even in the future (for example, if the provider is not planning to upgrade to the EVC infrastructure for the service that is being created), existing ISC policy types (L2VPN or VPLS) can be used instead of FlexUNI/EVC.

Step 3 Choose an **MPLS Core Connectivity Type** from the drop-down list.



Note

The core option supports MPLS only. There is no L2TPv3 support for this service.

The choices are:

- **PSEUDOWIRE**—Choose this option to allow connectivity between two N-PEs across the MPLS core. This option does not limit the service to point-to-point (E-Line). This is because even with the PSEUDOWIRE option selected, there can still be multiple CEs connected to a bridge domain on one or both sides of the pseudowire.
- **VPLS**—Choose this option to allow connectivity between multiple N-PEs across the MPLS core. There is no limit on the number of N-PEs across the MPLS core within a service request. However, many service requests can refer to the same customer-associated VPN.
- **LOCAL**—Choose this option for local connect cases in which there is no connectivity required across the MPLS core.

Local connect supports the following scenarios:

- All interfaces on the N-PE are FlexUNI-capable and using the EVC infrastructure. This is configured by associating all of the customer traffic on these interfaces to a bridge domain. This consumes a VLAN ID on the N-PE (equal to the bridge domain ID).
- Some interfaces on the N-PE are FlexUNI-capable, while others are switch-port-based. In such cases, all of the customer traffic on the interfaces that are configured with the EVC infrastructure are associated to a bridge domain. The traffic on the non-FlexUNI interfaces (and all the access nodes/interfaces beyond this N-PE) are configured with the Service Provider VLAN ID, where the Service Provider VLAN ID is the same as the bridge domain ID for the EVC-based services.

- Only two interfaces on the N-PE are involved, and both are based on FlexUNI-capable line cards. In the first case, the operator might choose not to configure the bridge domain option. In this case, the **connect** command that is used for the local connects are used, and the global VLAN is conserved on the device. If the operator chooses to configure with the bridge domain option, both interfaces are associated to a bridge domain ID, so that additional local links can be added to the service in future. This consumes a VLAN ID (bridge domain ID) on the N-PE.

Step 4 Check the **Configure With Bridge Domain** check box to determine bridge domain characteristics.

The behavior of the Configure With Bridge-Domain option works in tandem with the choice you selected in the MPLS Core Connectivity Type option, as follows.

- **PSEUDOWIRE** as the MPLS Core Connectivity Type. There are two cases:
 - A. With FlexUNI:
 - If **Configure With Bridge Domain** is checked, the policy configures pseudowires under SVIs associated to the bridge domain.
 - If **Configure With Bridge Domain** is unchecked, the policy will configure pseudowires directly under the service instance. This conserves the global VLAN.
 - B. Without FlexUNI:
 - If **Configure With Bridge Domain** is checked, the policy configures pseudowires as in L2VPN services (with SVIs).
 - If **Configure With Bridge Domain** is unchecked, the policy configures pseudowires directly under subinterfaces.

Only pseudowires can be either configured directly under service instance of the corresponding FlexUNI-capable interface or under SVIs associated to the bridge domain.

 - **LOCAL** as the MPLS Core Connectivity Type:
 - If **Configure With Bridge Domain** is checked, the policy allows either point-to-point or multipoint local connect services.
 - If **Configure With Bridge Domain** is unchecked, ISC allows only point-to-point local connects without bridge domain.
 - **VPLS—Configure With Bridge Domain** is checked by default and non-editable.

Step 5 Click **Next**.

The EVC Policy Editor - FlexUNI Attribute window appears, as shown in [Figure 3-3](#).

Step 6 Continue with the steps contained in the next section, [Setting the FlexUNI Attributes, page 3-9](#).

Setting the FlexUNI Attributes

This section describes how to set the FlexUNI attributes for the FlexUNI/EVC policy, as shown in [Figure 3-3](#).

Figure 3-3 EVC Policy Editor - FlexUNI Attribute Window

Attribute	Value	Editable
Service Attribute		
AutoPick Service Instance ID	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Enable Pseudowire Redundancy	<input type="checkbox"/>	<input checked="" type="checkbox"/>
AutoPick VC ID	<input type="checkbox"/>	<input checked="" type="checkbox"/>
AutoPick Bridge Domain VLAN ID	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Match Criteria		
Both Tags	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Inner VLAN Ranges	<input type="checkbox"/>	
Rewrite Criteria		
Pop Outer	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Push Outer	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Translate Outer	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Pop Inner	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Push Inner	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Translate Inner	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Note: * - Required Field

- Step 3 of 5 -

< Back Next > Finish Cancel

FlexUNI attributes are organized under the following categories:

- Service Attributes
- VLAN Match Criteria
- VLAN Rewrite Criteria

The following sections describe how to set the options under each category.

Setting the Service Attributes

To set the FlexUNI service attributes, perform the following steps.

- Step 1** Check the **AutoPick Service Instance ID** check box to specify that the service instance ID will be autogenerated and allocated to the link during service request creation.

If the check box is unchecked, while setting the ISC link attributes during service request creation, ISC will prompt the operator to specify the service instance ID.

Usage notes:

- The service instance ID represents an Ethernet Flow Point (EFP) on an interface in the EVC infrastructure. The service instance ID is locally significant to the interface. This ID has to be unique only at the interface level. The ID must be a value from 1 to 8000.
- There are no resource pools available in ISC from which to allocate the service instance IDs.
- It is the responsibility of the operator creating the service request to maintain the uniqueness of the ID at the interface level.

**Note**

Enable PseudoWire Redundancy is unsupported in this release of ISC. FlexUNI/EVC support does not allow specifying a different backup peer or different backup interface on the same peer. It is advisable to disable pseudowire redundancy (uncheck the check box for this attribute) in all FlexUNI/EVC policies and to make it noneditable (uncheck its **Editable** check box). This ensures the corresponding Pseudowire Redundancy attribute cannot be enabled in service requests based on FlexUNI policies.

Step 2 Check the **AutoPick VC ID** check box to have ISC autopick the VC ID during service request creation. If this check box is unchecked, the operator will be prompted to specify a VC ID during service request creation.

Usage notes:

- This attribute is available only if MPLS Core Connectivity of Type was set as PSEUDOWIRE or VPLS in the Service Options window (see [Setting the Service Options, page 3-7](#)).
- When AutoPick VC ID is checked, ISC allocates a VC ID for pseudowires from the ISC-managed VC ID resource pool.
- If MPLS Core Connectivity of Type is VPLS, ISC allocates the VPLS VPN ID from the ISC-managed VC ID resource pool.

Step 3 Check the **AutoPick Bridge Domain/VLAN ID** check box to have ISC autopick the VLAN ID for the service request during service request creation.

If this check box is unchecked, the operator will be prompted to specify a VLAN ID during service request creation.

Usage notes:

- AutoPick Bridge Domain/VLAN ID consumes a global VLAN ID on the device.
- The bridge domain/VLAN ID is picked from the existing ISC VLAN pool. Once the VLAN ID is assigned in the service request, ISC makes the VLAN ID unavailable for subsequent service requests.
- In the case of manual VLAN ID allocation, ISC does not manage the VLAN ID if the ID lies outside the range of an ISC-managed VLAN pool. In this case, the operator must ensure the uniqueness of the ID in the Ethernet access domain. If an operator specifies a VLAN ID that is within the range of an ISC-managed VLAN pool and the VLAN ID is already in use in the access domain, ISC displays an error message indicating that the VLAN ID is in use.

Note on Access VLAN IDs

An access VLAN ID is of local significance to the FlexUNI-capable ports. It should not be confused with the global VLANs. This can be visualized as a partitioning of the Ethernet access network beyond the FlexUNI ports into several subEthernet access domains (one each for a FlexUNI-capable port).

However, all the service interfaces on the Ethernet access nodes beyond the FlexUNI ports will have this very same VLAN ID for a link. This ID must be manually specified by the operator when setting the link attributes during service request creation. The operator must ensure the uniqueness of the ID across the FlexUNI-demarcated Ethernet access domain.

These VLAN IDs are not managed by ISC by means of locally-significant VLAN pools. But once a VLAN ID is assigned for a link in the service request, ISC makes the VLAN unavailable for subsequent service requests within the Ethernet access domain demarcated by the FlexUNI. Likewise, if a manually-specified VLAN is already in use in the access domain delimited by the FlexUNI, ISC will display an error message indicating that the new VLAN ID being specified is already in use on the NPC. The operator will be prompted to specify a different VLAN ID, which will be provisioned on the L2 access nodes.

- Step 4** Continue with the steps contained in the next section, [Setting the VLAN Matching Criteria Attributes, page 3-12](#).
-

Setting the VLAN Matching Criteria Attributes

Prior to the introduction of the FlexUNI capability, service providers could either deploy service-multiplexed services (ERS/ERMS or EVPL/EVCS) or service-bundled services on a single port. Both could not be supported simultaneously due to the limitations in the infrastructure, which only allowed matching the outer-most VLAN tag.

One of the key benefits of FlexUNI/EVC support in ISC is to provide a flexible means to examine the VLAN tags (up to two levels) of the incoming frames and associate them to appropriate Ethernet Flow Points (EFPs). This allows service providers to deploy simultaneously both the service-multiplexed and service-bundled services on a single port.

To set the FlexUNI VLAN matching criteria attributes, perform the following steps.

- Step 1** Check the **Both Tags** check box to enable service requests created with the policy to match both the inner and outer VLAN tags of the incoming frames.
- If you do not check this check box, service requests created with the policy will match only the outer VLAN tag of the incoming frames.
- Checking the Both Tags attribute causes the Inner VLAN Ranges attribute (covered in the next steps) to appear in the FlexUNI Attribute window.
- Step 2** Check the **Inner VLAN Ranges** check box to enable the range of inner VLAN tags to be specified during service request creation.
- If the check box is unchecked, the range of inner VLAN tags are not allowed. In this case, the operator must specify discrete VLAN IDs during service request creation.
- Step 3** Continue with the steps contained in the next section, [Setting the VLAN Rewrite Criteria Attributes, page 3-12](#).
-

Setting the VLAN Rewrite Criteria Attributes

Together with VLAN matching criteria, VLAN rewrite makes the FlexUNI/EVC infrastructure very powerful and flexible. The following VLAN rewrite options are supported:

- Pop one or two tags.
- Push one or two tags.
- Translation (1:1, 2:1, 1:2, 2:2).

Be aware of the following considerations when setting the VLAN rewrite criteria attributes:

- Only one kind of rewrite can be done on every CE-facing FlexUNI link.
- All VLAN rewrites are done using the **symmetric** keyword on the ingress traffic (for example, **rewrite ingress tag pop 2 symmetric**).

- For any service instance, only one type of rewrite option (pop, push, or translate) is allowed per instance. For example, if pop out is enabled, push inner, push outer, translate inner, and translate outer are not available.

To set the FlexUNI VLAN rewrite criteria attributes, perform the following steps.

-
- Step 1** Check the **Pop Outer** check box to pop the outer VLAN ID tag of the incoming frames that fulfill the match criteria.
- If this check box is unchecked, the outer tag of the incoming traffic is not popped.
- Step 2** Check the **Pop Inner** check box to pop the inner VLAN ID tag of the incoming frames that fulfill the match-criteria.
- If this check box is unchecked, the inner tag is not popped. Note that, if Pop Inner is checked, Pop Outer is automatically checked.
- Step 3** Check the **Push Outer** check box to impose an outer VLAN ID tag onto the incoming frames that fulfill the match criteria.
- If this check box is unchecked, no outer tag is imposed on the incoming frames.
- Usage notes:
- If Push Outer is checked, all service requests created with the policy push a dot1q outer tag on the incoming frames matching the match criteria. When creating the link during service creation, the operator can specify an outer tag with a value from 1 to 4096.
 - This attribute is available regardless of the number of tags used in the match criteria. Whether the incoming traffic is double tagged or single tagged, if Push Outer is enabled, all corresponding service requests push an outer tag. All subsequent nodes consider only the outer-most two tags (if FlexUNI-capable) or just one tag (not FlexUNI-capable) and treat the inner-most tags transparently as payload.
 - This VLAN ID is not derived from ISC-managed VLAN ID pools.
- Step 4** Check the **Push Inner** check box to impose an inner VLAN ID tag onto the incoming frames that fulfill the match criteria.
- This operation pushes both an inner and an outer tag onto the incoming packet, not just an inner tag. If this check box is unchecked, no inner tag is imposed on the incoming frames.
- Usage notes:
- If Push Inner is checked, all service requests created with the policy push a dot1q inner tag on the incoming frames matching the match criteria. When creating the link during service creation, the operator can specify an inner tag with a value from 1 to 4096.
 - If Push Inner is checked, Push Outer is automatically checked.
 - This attribute is available regardless of the number of tags used in the match criteria. Regardless of whether the incoming traffic is double tagged or single tagged, if Push Inner is enabled, all corresponding service requests push an inner tag. All subsequent nodes consider only the outer-most two tags (if FlexUNI-capable) or just one tag (not FlexUNI-capable) and treat the inner-most tags transparently as payload.
 - This VLAN ID is not derived from ISC-managed VLAN ID pools.
- Step 5** Check the **Translate Outer** check box to allow the operator to specify a target outer VLAN ID during service request creation.
- The outer tag of all the incoming frames that fulfill the match criteria are translated to this ID. If the check box is unchecked, no outer tag translation is performed. See [Table 3-1](#).

Step 6 Check the **Translate Inner** check box to allow the operator to specify a target inner VLAN ID during service request creation.

The inner tag of all the incoming frames that fulfill the match criteria are translated to this ID. If the check box is unchecked, no inner tag translation is performed. See [Table 3-1](#).

**Note**

[Table 3-1](#) summarizes the realization of different VLAN translations available in the FlexUNI/EVC infrastructure. The second and third columns (Match Outer Tag and Match Inner Tag) refer to policy settings. The last two columns (Translate Outer Tag and Translate Inner Tag) indicate the VLAN translation that occurs on the incoming frames.

Table 3-1 VLAN Translation Summary Table

Type	Match Outer Tag	Match Inner Tag	Translate Outer Tag	Translate Inner Tag
1:1	True	N/A	Yes	No
1:2	True	N/A	Yes	Yes
2:1	True	True	Yes	No
2:2	True	True	Yes	Yes

Step 7 Click **Next**.

The EVC Policy Editor - Interface Attribute window appears, as shown in [Figure 3-3](#).

Step 8 Continue with the steps contained in the next section, [Setting the Interface Attributes, page 3-14](#).

Setting the Interface Attributes

This step of creating the FlexUNI/EVC policy involves setting the interface attributes, as shown in the EVC Policy Editor - Interface Attribute window in [Figure 3-4](#). The attributes you can configure in this window are grouped under the following categories:

- N-PE/U-PE information
- Speed and duplex information
- ACL name and MAC addresses
- UNI port security
- Storm control
- L2 protocol tunneling

In some cases, checking an attribute causes additional attributes to appear in the GUI. This is covered in the steps that follow.

**Note**

If the CE is directly connected to an N-PE, only speed, duplex, UNI shutdown, and other generic options are presented. In this case, port security, storm control, L2 protocol tunneling, and other advanced features are not supported due to the current platform limitations. If these features are needed for a service, the service provider must deploy Layer 2 Ethernet access nodes beyond the FlexUNI to support these requirements.

Figure 3-4 EVC Policy Editor - Interface Attributes Window

EVC Policy Editor-Interface Attribute

Attribute	Value	Editable
N-PEU-PE Information		
Encapsulation	DOT1QTRUNK	<input checked="" type="checkbox"/>
Standard UNI Port	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
UNI Shutdown	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Keep Alive	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Speed and Duplex Information		
Link Speed	Auto	<input checked="" type="checkbox"/>
Link Duplex	None	<input checked="" type="checkbox"/>
Use Existing ACL Name		
Port-Based ACL Name		<input checked="" type="checkbox"/>
UNI MAC Addresses		<input type="checkbox"/> Edit
Filter BPDU	<input type="checkbox"/>	<input checked="" type="checkbox"/>
UNI Port Security		
Protocol Tunneling	<input type="checkbox"/>	<input checked="" type="checkbox"/>
VLAN Translation		
	<input checked="" type="radio"/> No <input type="radio"/> 1:1 <input type="radio"/> 2:1	<input checked="" type="checkbox"/>
MTU size (1500-9216)		<input checked="" type="checkbox"/>
PW Tunnel Selection ⓘ	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Note: * - Required Field

- Step 4 of 5 -

< Back Next > Finish Cancel

To set the FlexUNI/EVC interface attributes, perform the following steps.

Step 1 Choose an **Encapsulation** type.

The choices are:

- **DOT1QTRUNK**—Configures the UNI as a trunk with 802.1q encapsulation. If the UNI belongs to a directly connected and FlexUNI link, this setting signifies that the incoming frames are 802.1q encapsulated and that they match the VLAN ID configured for the link. This specific topology does not involve a trunk UNI as such.
- **DOT1QTUNNEL**—Configures the UNI as an 802.1q tunnel (also known as a dot1q tunnel or Q-in-Q) port.
- **ACCESS**—Configures the UNI as an access port.

Step 2 Check the **Standard UNI Port** check box to enable port security.

This is the default. When you uncheck the check box, the port is treated as an uplink with no security features, and the window dynamically changes to eliminate items related to port security.

Step 3 Check the **UNI Shutdown** check box if you want to leave the UNI port shut during service activation, for example, when the service provider wants to deploy a service in the network but wants to activate it at a later time.

- Step 4** Check the **Keep Alive** check box to configure keepalives on the UNI port.
By default, this check box is unchecked, which causes the command **no keepalive** to be provisioned on the UNI port. This prevents a CPE from sending keepalive packets to the U-PE, for security purposes. This attribute is editable, in order to support modification on a per-service request basis.
- Step 5** Enter a **Link Speed** (optional) of None, 10, 100, 1000, Auto, or nonegotiate.
- Step 6** Enter a **Link Duplex** (optional) of None, Full, Half, or Auto.
- Step 7** Check the **Use Existing ACL Name** check box if you want to assign your own named access list to the port.
By default, this check box is not checked and ISC automatically assigns a MAC-based ACL on the customer facing UNI port, based on values you enter in **UNI MAC addresses** (below).
- Step 8** Enter a **Port-Based ACL Name** (if you checked the **Use Existing ACL Name** check box, as mentioned in the previous step).



Note ISC does not create this ACL automatically. The ACL must already exist on the device, or be added as part of a template, before the service request is deployed. Otherwise, deployment will fail.

- Step 9** Enter one or more Ethernet MAC addresses in **UNI MAC addresses**.
This selection is present only if you uncheck the **Use Existing ACL Name** check box. Click the **Edit** button to bring up a pop-up window in which you enter MAC addresses to be allowed or denied on the port. You can also specify a range of addresses by setting a base MAC address and a filtered MAC address.
- Step 10** Check the **UNI Port Security** check box (see [Figure 3-5](#)) if you want to provision port security-related CLIs to the UNI port by controlling the MAC addresses that are allowed to go through the interface.
- a. For **Maximum Number of MAC address**, enter the number of MAC addresses allowed for port security.
 - b. For **Aging**, enter the length of time the MAC address can stay on the port security table.
 - c. For **Violation Action**, choose what action will occur when a port security violation is detected:
 - **PROTECT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value.
 - **RESTRICT**—Drops packets with unknown source addresses until a sufficient number of secure MAC addresses are removed to drop below the maximum value and causes the Security Violation counter to increment.
 - **SHUTDOWN**—Puts the interface into the error-disabled state immediately and sends an SNMP trap notification.
 - d. In the **Secure MAC Addresses** field, enter one or more Ethernet MAC addresses.

Figure 3-5 UNI Port Security




UNI Port Security	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Maximum MAC Address(1 - 8448)	<input type="text"/>	<input checked="" type="checkbox"/>
Aging (in minutes)(0 - 1440)	<input type="text"/>	<input checked="" type="checkbox"/>
Violation Action	PROTECT	<input checked="" type="checkbox"/>
Secure MAC Addresses	<input type="text"/>	<input checked="" type="checkbox"/>

211668

- Step 11** Check the **Enable Storm Control** check box (see [Figure 3-6](#)) to help prevent the UNI port from being disrupted by a broadcast, multicast, or unicast storm.

Enter a threshold value for each type of traffic. The value, which can be specified to two significant digits, represents the percentage of the total available bandwidth of the port. If the threshold of a traffic type is reached, further traffic of that type is suppressed until the incoming traffic falls below the threshold level.




Figure 3-6 Enable Storm Control

Enable Storm Control	<input checked="" type="checkbox"/>	
UNI Storm Control		
Unicast Traffic(0.0 - 100.0%) 	<input type="text"/>	<input checked="" type="checkbox"/>
Broadcast Traffic(0.0 - 100.0%) 	<input type="text"/>	<input checked="" type="checkbox"/>
Multicast Traffic(0.0 - 100.0%) 	<input type="text"/>	<input checked="" type="checkbox"/>

211669

- Step 12** Check the **Protocol Tunnelling** check box (see [Figure 3-7](#)) if you want to define the Layer 2 Bridge Protocol Data Unit (BPDU) frames that can be tunneled over the core to the other end.

Figure 3-7 Protocol Tunnelling

Protocol Tunnelling	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Enable cdp	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
cdp shutdown threshold(0-4096)	<input type="text"/>	<input checked="" type="checkbox"/>
cdp drop threshold(0-4096) 	<input type="text"/>	<input checked="" type="checkbox"/>
Enable vtp	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
vtp shutdown threshold(0-4096)	<input type="text"/>	<input checked="" type="checkbox"/>
vtp drop threshold(0-4096) 	<input type="text"/>	<input checked="" type="checkbox"/>
Enable stp	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
stp shutdown threshold(0-4096)	<input type="text"/>	<input checked="" type="checkbox"/>
stp drop threshold(0-4096) 	<input type="text"/>	<input checked="" type="checkbox"/>
Recovery Interval (in seconds)(30-86400)	<input type="text"/>	<input checked="" type="checkbox"/>

211670

For each protocol that you choose, enter the shutdown threshold and drop threshold for that protocol:

- a. **Enable cdp**—Enable Layer 2 tunnelling on Cisco Discover Protocol (CDP).
- b. **cdp shutdown threshold**—Enter the number of packets per second to be received before the interface is shut down.

- c. **cdp drop threshold**—Enter the number of packets per second to be received at which point the interface will start dropping CDP packets.
- d. **Enable vtp**—Enable Layer 2 tunnelling on VLAN Trunk Protocol (VTP).
- e. **vtp shutdown threshold**—Enter the number of packets per second to be received before the interface is shut down.
- f. **vtp drop threshold**—Enter the number of packets per second to be received at which point the interface will start dropping VTP packets.
- g. **Enable stp**—Enable Layer 2 tunnelling on Spanning Tree Protocol (STP).
- h. **stp shutdown threshold**—Enter the number of packets per second to be received before the interface is shut down.
- i. **stp drop threshold**—Enter the number of packets per second to be received at which point the interface will start dropping STP packets.
- j. **Recovery Interval**—Enter the amount of time, in seconds, to wait before recovering a UNI port.

Step 13 Specify the type of **VLAN Translation** for this policy by clicking the appropriate radio button.

The choices are:

- **No**—No VLAN translation is performed. (This is the default.)
- **1:1**—1:1 VLAN translation.
- **2:1**—2:1 VLAN translation.



Note For detailed coverage of setting up VLAN translation, see [Appendix C, “Setting Up VLAN Translation.”](#)



Note VLAN translation is only supported on links that are specified as non-FlexUNI at the service request level.

Step 14 Enter the **MTU Size** in bytes.

The maximum transmission unit (MTU) size is configurable and optional. The default size is 9216, and the range is 1500 to 9216. ISC does not perform an integrity check for this customized value. If a service request goes to the Failed Deploy state because this size is not accepted, you must adjust the size until the Service Request is deployed.

In ISC 5.1, different platforms support different ranges.

- For the 3750 and 3550 platforms, the MTU range is 1500 to 1546.
- For the Cisco 7600 Ethernet port, the MTU size is always 9216. Even with the same platform and same IOS release, different line cards support the MTU differently. For example, older line cards only take an MTU size of 9216 and newer cards support 1500 to 9216. However, ISC 5.1 uses 9216 in both cases.
- For the Cisco 7600 SVI (interface VLAN), the MTU size is 1500 to 9216.

Step 15 Check the **PW Tunnel Selection** check box if you want to be able to manually select the Traffic Engineering (TE) tunnel for the pseudowire connecting point-to-point N-PEs.

This attribute is unchecked by default.

Subsequently, when you create a service request based on this policy, you must specify the TE tunnel ID in a field provided. ISC uses the tunnel information to create and provision a pseudowire class that describes the pseudowire connection between two N-PEs. This pseudowire class can be shared by more than one pseudowire, as long as the pseudowires share the same tunnel ID and remote loopback address. You are responsible to ensure that the tunnel interface and associated ID are configured. During service request creation when you specify the tunnel ID number, ISC does not check the validity of the value. That is, ISC does not verify the existence of the tunnel.

- Step 16** If you would like to enable template association for this policy, click the **Next** button.
See the section [Enabling Template Association, page 3-19](#) for information about this feature.
- Step 17** To save the FlexUNI/EVC policy, click **Finish**.
-

To create a service request based on a FlexUNI/EVC policy, see [Chapter 4, “Managing a FlexUNI/EVC Service Request.”](#)

Enabling Template Association

The ISC template feature gives you a means to download free-format CLIs to a device. If you enable templates, you can create templates and data files to download commands that are not currently supported by ISC.

- Step 1** To enable template association for the policy, click the **Next** button in EVC Policy Editor - Interface Attribute window (before clicking **Finish**).
- The Template Association window appears. In this window, you can enable template support and, optionally, associate templates and data files with the policy. For instructions about associating templates with policies and how to use the features in this window, see [Appendix B, “Working with Templates and Data Files”](#).
- Step 2** When you have completed setting up templates and data files for the policy, click **Finish** in the Template Association window to close it and return to the Policy Editor window.
- Step 3** To save the FlexUNI/EVC policy, click **Finish**.
-

To create a service request based on a FlexUNI/EVC policy, see [Chapter 4, “Managing a FlexUNI/EVC Service Request.”](#)

