



論理デバイスの定義

- デバイス クラスタについて (1 ページ)
- 具象デバイスについて (3 ページ)
- トランкиングの概要 (3 ページ)
- レイヤ 4～レイヤ 7 サービスのエンドポイント グループについて (3 ページ)
- グラフ コネクタに対する静的なカプセル化の使用 (4 ページ)
- GUI を使用したレイヤ 4～レイヤ 7 サービスデバイスの設定 (4 ページ)
- NX OS スタイル CLI を使用したレイヤ 4～レイヤ 7 の作成 (7 ページ)
- NX-OS スタイルの CLI を使用したハイ アベイラビリティ クラスタの作成 (12 ページ)
- NX-OS スタイルの CLI を使用した仮想デバイスの作成 (13 ページ)
- 論理デバイスを作成する XML の例 (14 ページ)
- GUI を使用したデバイスの変更 (16 ページ)
- GUI を使用してレイヤ 7 仮想 ASA デバイスにレイヤ 4 でのトランкиングを有効化 (17 ページ)
- REST API を使用してレイヤ 7 仮想 ASA デバイスにレイヤ 4 でのトランкиングを有効化 (18 ページ)
- REST API とともにインポートされたデバイスの使用 (18 ページ)
- NX-OS スタイルの CLI を使用した別のテナントからのデバイスの作成 (19 ページ)
- GUI を使用したデバイスのインポートの確認 (19 ページ)

デバイス クラスタについて

デバイス クラスタ (別名論理デバイス) は、単一のデバイスとして機能する1つ以上の具象デバイスです。デバイス クラスタには、そのデバイス クラスタのインターフェイス情報を説明するクラスタ (論理) インターフェイスがあります。サービス グラフ テンプレートのレンダリング時に、機能ノードコネクタはクラスタ (論理) インターフェイスに関連付けられます。Application Policy Infrastructure Controller (APIC) は、サービス グラフ テンプレートのインスタンス化およびレンダリング時に機能ノードコネクタにネットワーク リソース (VLAN) を割り当て、クラスタ (論理) インターフェイスにネットワーク リソースをプログラミングします。

■ デバイス クラスタについて

Cisco APIC では、グラフのインスタンス化時にサービスグラフに対してネットワークリソースのみを割り当てて、ファブリック側のみをプログラミングできます。この動作は、既存のオーケストレータまたはデバイスクラスタ内のデバイスをプログラムする `dev-op` ツールがすでにある環境では有効です。

Cisco APIC はデバイス クラスタおよびデバイスのトポロジ情報（論理インターフェイスと具象インターフェイス）を把握する必要があります。この情報により、Cisco APIC はリーフスイッチの適切なポートをプログラミングできます。また、Cisco APIC ではこの情報をトラブルシューティング ウィザードの目的で使用できます。さらに、Cisco APIC はカプセル化の割り当てに使用する `DomP` との関係も把握する必要があります。

デバイスクラスタまたは論理デバイスは、物理デバイスまたは仮想デバイスのいずれかです。デバイスクラスタは、そのクラスタの一部である仮想マシンが、VMM ドメインを使用して Cisco APIC と統合されたハイパーバイザ上に存在する場合、仮想と見なされます。これらの仮想マシンが VMM ドメインの一部ではない場合、仮想マシンインスタンスであっても物理デバイスとして扱われます。



(注) 論理デバイスには、VMware VMM ドメインまたは SCVMM VMM ドメインのみを使用できます。

次の設定が必要です。

- 論理デバイス (`vnsLDevViP`) およびデバイス (`cDev`) の接続情報
- サポートする機能タイプ (go-through、go-to、L1、L2) に関する情報

サービス グラフ テンプレートは、管理者が定義するデバイス選択ポリシー（論理デバイス コンテキストと呼ばれます）に基づく特定のデバイスを使用します。

管理者は、アクティブ/スタンバイモードで最大2つの具象デバイスをセットアップできます。

デバイス クラスタをセットアップするには、次のタスクを実行する必要があります。

- ファブリックに具象デバイスを接続します。
- Cisco APIC を使用してデバイス クラスタを構成します。



(注) Cisco APIC は、2つのデバイスのクラスタに IP アドレスが重複して割り当てられているかどうかを検証しません。Cisco APIC は、2つのデバイスのクラスタが同じ管理 IP アドレスを持っている場合、不適切なデバイスのクラスタをプロビジョニングすることができます。デバイスクラスタで IP アドレスが重複している場合には、いずれかのデバイスの IP アドレスの設定を削除し、管理 IP アドレスの設定のためにプロビジョニングされた IP アドレスが重複していないことを確認してください。

具象デバイスについて

具象デバイスとしては、物理デバイスと仮想デバイスがあり得ます。デバイスが仮想デバイスの場合は、コントローラ (vCenter または SCVMM コントローラ) と仮想マシン名を選択する必要があります。具象デバイスには、具象インターフェイスがあります。具象デバイスが論理デバイスに追加されると、具象インターフェイスが論理インターフェイスにマッピングされます。サービス グラフ テンプレートのインスタンス化時に、VLAN および VXLAN は、論理インターフェイスとの関連付けに基づいた具象インターフェイス上でプログラミングされます。

トランкиングの概要

レイヤ4～レイヤ7仮想 ASA デバイスのトランкиングを有効にでき、これはトランク ポート グループを使用してエンドポイント グループのトラフィックを集約します。トランкиングを使用せず、仮想サービス デバイスには各インターフェイスに1個の VLAN のみ所有し、最大10個のサービス グラフを所有できます。トランкиングが有効にしている状態では、仮想サービス デバイスはサービス グラフの数を無制限に設定できます。

トランク ポート グループについての詳細は、『Cisco ACI Virtualization Guide』を参照してください。

レイヤ4～レイヤ7サービスのエンドポイント グループについて

Application Policy Infrastructure Controller (APIC) を使用すると、グラフのインスタンス化中にグラフ コネクタに使用するエンドポイント グループを指定できます。これにより、グラフ導入のトラブルシューティングが容易になります。APIC は、指定されたレイヤ4～レイヤ7サービス エンドポイント グループを使用してリーフスイッチにカプセル化情報をダウンロードします。また、APIC はこのエンドポイント グループを使用して仮想デバイスの分散仮想スイッチにポート グループを作成します。さらに、レイヤ4～レイヤ7サービスのエンドポイント グループを使用して、グラフ コネクタのエラー情報や統計情報も集約します。

導入されたグラフ リソースへの可視性の向上に加えて、レイヤ4～レイヤ7サービスのエンドポイント グループも使用して、特定のグラフ インスタンスに使用する静的なカプセル化を指定することもできます。このカプセル化は、複数のグラフ インスタンス間でレイヤ4～レイヤ7サービスのエンドポイント グループを共有することによって、複数のグラフ インスタンス間で共有することもできます。

グラフ コネクタと共にレイヤ4～レイヤ7サービスのエンドポイントをどのように使用できるかを示す XML コードの例については、[レイヤ4～レイヤ7サービスのエンドポイント グループとコネクタを関連付ける XML の例 \(16 ページ\)](#) を参照してください。

■ グラフ コネクタに対する静的なカプセル化の使用

グラフ コネクタに対する静的なカプセル化の使用

Application Policy Infrastructure Controller (APIC) は、処理中にさまざまなサービス グラフにカプセル化を割り当てます。一部の使用例では、サービス グラフ内の特定のコネクタに使用するカプセル化を明示的に指定できます。これは静的なカプセル化と呼ばれます。静的なカプセル化は、物理サービスを持つサービス デバイス クラスタがあるサービス グラフ コネクタについてのみサポートされます。仮想サービス デバイスがあるサービス デバイス クラスタは、そのサービス デバイス クラスタに関連付けられた VMware または SCVMM ドメインからの VLAN を使用します。

静的なカプセル化は、レイヤ 4～レイヤ 7 サービスのエンドポイント グループの一部としてカプセル化値を指定することによってグラフ コネクタで使用できます。レイヤ 4～レイヤ 7 サービスのエンドポイントで静的なカプセル化の使用方法を示す XML コードの例については、[レイヤ 4～レイヤ 7 サービスのエンドポイント グループで静的なカプセル化を使用する XML の例 \(16 ページ\)](#) を参照してください。

GUI を使用したレイヤ 4～レイヤ 7 サービス デバイスの設定

レイヤ 4～レイヤ 7 サービス デバイスを作成すると、物理デバイスまたは仮想マシンのいずれかに接続できます。接続先のタイプによって、フィールドが若干異なります。物理デバイスに接続する場合は、物理インターフェイスを指定します。仮想マシンに接続する場合は、VMM ドメイン、仮想マシン、および仮想インターフェイスを指定します。さらに、不明モデルを選択することで、接続を手動で設定することもできます。

始める前に

- ・テナントを作成しておく必要があります。

手順

-
- ステップ 1** メニュー バーで、[Tenants] > [All Tenants] の順に選択します。
- ステップ 2** [Work] ペインで、テナントの名前をダブルクリックします。
- ステップ 3** [Navigation] ウィンドウで、**Tenant *tenant_name* > Services > L4-L7 > Devices** を選択します。
- ステップ 4** 作業 ウィンドウで、**Actions > Create L4-L7 Devices** を選択します。
- ステップ 5** [Create L4-L7 Devices] ダイアログ ボックスで、[General] セクションの次のフィールドに入力します。

名前	説明
[名前 (Name)] フィールド	デバイスの名前を入力します。

名前	説明
[Service Type] ドロップダウンリスト	<p>サービス タイプを選択します。タイプは次のとおりです。</p> <ul style="list-style-type: none"> ADC ファイアウォール その他 <p>(注) レイヤ1/レイヤ2 ファイアウォール設定の場合は、[その他 (Other)] を選択します。</p>
[Device Type] ボタン	デバイス タイプを選択します。
[Physical Domain] ドロップダウンリストまたは[VMM Domain] ドロップダウンリスト	物理ドメインまたはVMM ドメインを選択します。
スイッチング モード (Cisco ACI Virtual Edgeのみ)	<p>Cisco ACI Virtual Edge仮想ドメインでは、次のモードのいずれかを選択します:</p> <ul style="list-style-type: none"> AVE : トラフィックは Cisco ACI Virtual Edge を介して切り替えられます。 native : トラフィックは VMware DVS を介して切り替えられます。
View ラジオボタンを表示します。	<p>デバイスのビューを選択します。ビューとしては、次のものがあります:</p> <ul style="list-style-type: none"> 単一ノード : 1つのノードのみ HA ノード : ハイアベイラビリティノード (2ノード) クラスタ : 3ノード以上

■ GUI を使用したレイヤ4～レイヤ7サービスデバイスの設定

名前	説明
コンテキスト認識	<p>デバイスのコンテキスト認識。認識は次のとおりです。</p> <ul style="list-style-type: none"> ・単一 (Single) : プロバイダーネットワークでホストされる特定のタイプの複数のテナントでは、デバイスクラスタを共有できません。特定のユーザーの特定のテナントにデバイスクラスタを提供する必要があります。 ・複数 (Multiple) : プロバイダーネットワークでホストされる特定のタイプの複数のテナント全体でデバイスクラスタを共有できます。たとえば、同じデバイスを共有する2つのホスティング会社が存在する可能性があります。 <p>デフォルトは単一 (Single)です。</p> <p>(注) ロードバランサであるレイヤ4～レイヤ7サービスデバイスを作成する場合、コンテキスト認識パラメータは使用されないため無視できます。5.2(1)リリース以降、このパラメータは廃止され、Cisco APIC は値を無視します。</p>
機能タイプ	<p>機能種別は次のとおりです。</p> <ul style="list-style-type: none"> ・GoThrough : 透過モード ・GoTo : ルーテッドモード ・L1 : レイヤ1ファイアウォールモード ・L2 : レイヤ2ファイアウォールモード <p>デフォルトはGoToです。</p> <p>(注) レイヤ1またはレイヤ2モードの場合、チェックボックスをオンにしてアクティブ/アクティブモードを有効にします。有効にすると、レイヤ1/レイヤ2 PBR デバイスのアクティブ/アクティブ展開/ECMP パスがサポートされます。</p>

ステップ6 [Device 1] セクションで、次のフィールドに入力します。

名前	説明
[VM] ドロップダウンリスト	(仮想デバイスタイプの場合のみ) 仮想マシンを選択します。

ステップ7 [Device Interfaces] テーブルで、[+] ボタンをクリックしてインターフェイスを追加し、次のフィールドに入力します。

名前	説明
[Name] ドロップダウンリスト	インターフェイス名を選択します。
[vNIC] ドロップダウンリスト	(仮想デバイスタイプの場合のみ) vNICを選択します。
[Path] ドロップダウンリスト	(物理デバイスタイプまたはL3Outのインターフェイスの場合のみ) インターフェイスが接続されるポート、ポートチャネル、仮想ポートチャネルを選択します。

ステップ8 [Update] をクリックします。

ステップ9 (HA クラスタの場合のみ) 各デバイスのフィールドに入力します。

ステップ10 [クラスタインターフェイス (Cluster Interfaces)] セクションのフィールドに入力します。

[+] をクリックしてクラスタインターフェイスを追加し、次の詳細を入力します。

名前	説明
[Name] ドロップダウンリスト	クラスタインターフェイスの名前を入力します。
Concrete Interfaces ドロップダウンリスト	具象インターフェイスを選択します。ドロップダウンリストのインターフェイスは、手順7で作成したデバイスインターフェイスに基づいています。
[拡張 LAG ポリシー (Enhanced Lag Policy)] ドロップダウンリスト	(オプション) デバイスのVMM ドメインに構成されているLAG ポリシーを選択します。 このオプションは、[デバイスタイプ (Device Type)](手順5で説明)を [Virtual (仮想)]に選択した場合にのみ使用できます。

HA クラスタでは、クラスタのインターフェイスが、クラスタ内の両方の具体デバイスにある対応するインターフェイスにマッピングされていることを確認してください。

ステップ11 [完了 (Finish)] をクリックします。

NX OS スタイル CLI を使用したレイヤ4～レイヤ7の作成

レイヤ4～レイヤ7デバイスを作成するときに、物理デバイスまたは仮想マシンのいずれかに接続できます。物理デバイスに接続する場合は、物理インターフェイスを指定します。仮想マ

■ NX OS スタイル CLI を使用したレイヤ4～レイヤ7の作成

シンに接続する場合は、VMM ドメイン、仮想マシン、および仮想インターフェイスを指定します。



(注)

ロード バランサであるレイヤ4～レイヤ7デバイスを設定する場合、[コンテキスト認識] パラメータは使用されません。[コンテキスト認識] パラメータには、無視可能なシングル コンテキストのデフォルト値があります。

始める前に

- ・テナントを作成しておく必要があります。

手順

ステップ1 コンフィギュレーション モードを開始します。

例：

```
apic1# configure
```

ステップ2 テナントのコンフィギュレーション モードを開始します。

```
tenant tenant_name
```

例：

```
apic1(config)# tenant t1
```

ステップ3 レイヤ4～レイヤ7デバイス クラスタを追加します。

```
1417 cluster name cluster_name type cluster_type vlan-domain domain_name
[function function_type] [service service_type]
```

パラメータ	説明
name	デバイス クラスタの名前。
type	デバイス クラスタのタイプ。値は次のとおりです。 <ul style="list-style-type: none"> virtual physical
vlan-domain	VLAN の割り当てに使用するドメイン。このドメインは、仮想デバイスの場合はVMM ドメイン、物理デバイスの場合は物理ドメインである必要があります。
switching-mode (Cisco ACI Virtual Edgeのみ)	(オプション) 次のいずれかのモードを選択します。 <ul style="list-style-type: none"> AVE : Cisco ACI Virtual Edge を通過するトライフィックのスイッチ。 ネイティブ : VMware DVS を通過するトライフィックのスイッチ。これはデフォルト値です。

パラメータ	説明
機能	(任意) 機能タイプ。値は次のとおりです。 <ul style="list-style-type: none">• go-to• go-through• L1• L2
service	(任意) サービス タイプ。ADC 固有またはファイアウォール固有のアイコンおよび GUI を表示するために GUI で使用します。値は次のとおりです。 <ul style="list-style-type: none">• ADC• FW• OTHERS

例：

物理デバイスの場合は、次のように入力します。

```
apic1(config-tenant)# 1417 cluster name D1 type physical vlan-domain phys
  function go-through service ADC
```

仮想デバイスの場合は、次のように入力します。

```
apic1(config-tenant)# 1417 cluster name ADCCluster1 type virtual vlan-domain mininet
```

ステップ4 1つ以上のクラスタ デバイスをデバイス クラスタに追加します。

```
cluster-device device_name [vcenter vcenter_name] [vm vm_name]
```

パラメータ	説明
vcenter	(仮想デバイスの場合のみ) 仮想デバイスの仮想マシンをホストする VCenter の名前。
vm	(仮想デバイスの場合のみ) 仮想デバイスの仮想マシンの名前。

例：

物理デバイスの場合は、次のように入力します。

```
apic1(config-cluster)# cluster-device C1
apic1(config-cluster)# cluster-device C2
```

仮想デバイスの場合は、次のように入力します。

```
apic1(config-cluster)# cluster-device C1 vcenter vcenter1 vm VM1
apic1(config-cluster)# cluster-device C2 vcenter vcenter1 vm VM2
```

ステップ5 1つ以上のクラスタインターフェイスをデバイス クラスタに追加します。

```
cluster-interface interface_name [vlan static_encap]
```

■ NX OS スタイル CLI を使用したレイヤ4～レイヤ7の作成

パラメータ	説明
vlan	(仮想デバイスの場合のみ) クラスタインターフェイスのスタティックなカプセル化。VLAN の値は、1～4094 とする必要があります。

例：

物理デバイスの場合は、次のように入力します。

```
apic1(config-cluster)# cluster-interface consumer vlan 1001
```

仮想デバイスの場合は、次のように入力します。

```
apic1(config-cluster)# cluster-interface consumer
```

ステップ6 1つ以上のメンバーをクラスタインターフェイスに追加します。

```
member device device_name device-interface interface_name
```

パラメータ	説明
デバイス	cluster-device コマンドを使用して、このデバイスにすでに追加されている必要があるクラスタデバイスの名前。
device-interface	クラスタデバイス上のインターフェイスの名前。

例：

```
apic1(config-cluster-interface)# member device C1 device-interface 1.1
```

ステップ7 メンバーにインターフェイスを追加します。

```
interface {ethernet ethernet_port | port-channel port_channel_name [fex fex_ID] |  
vpc vpc_name [fex fex_ID]} leaf leaf_ID
```

インターフェイスではなく vNIC を追加する場合は、このステップをスキップします。

パラメータ	説明
ethernet	(イーサネットまたはFEXイーサネットインターフェイスの場合のみ) クラスタデバイスがCisco Application Centric Infrastructure (ACI) ファブリックに接続されるリーフ上のイーサネットポート。FEXイーサネットメンバーを追加する場合は、FEX IDとFEXポートの両方を次の形式で指定します。 <i>FEX_ID/FEX_port</i> 次に例を示します。 101/1/23 FEX IDは、クラスタデバイスがファブリックエクステンダにどこで接続するかを指定します。
port-channel	(ポートチャネルまたはFEXポートチャネルインターフェイスの場合のみ) クラスタデバイスがACIファブリックに接続されるポートチャネル名。

パラメータ	説明
vpc	(バーチャルポートチャネルまたはFEXバーチャルポートチャネルインターフェイスの場合のみ) クラスタデバイスがACIファブリックに接続されるバーチャルポートチャネル名。
fex	(ポートチャネル、FEXポートチャネル、バーチャルポートチャネル、またはFEXバーチャルポートの場合のみ) ポートチャネルまたはバーチャルポートチャネルの形成に使用するスペース区切りリスト形式の FEX ID。
leaf	クラスタデバイスがどこで接続するかのスペース区切りリスト内のリーフID。

例：

イーサネットインターフェイスの場合は、次のように入力します。

```
apic1(config-member)# interface ethernet 1/23 leaf 101
apic1(config-member)# exit
```

FEX イーサネットインターフェイスの場合は、次のように入力します。

```
apic1(config-member)# interface ethernet 101/1/23 leaf 101
apic1(config-member)# exit
```

ポートチャネルインターフェイスの場合は、次のように入力します。

```
apic1(config-member)# interface port-channel pc1 leaf 101
apic1(config-member)# exit
```

FEX ポートチャネルインターフェイスの場合は、次のように入力します。

```
apic1(config-member)# interface port-channel pc1 leaf 101 fex 101
apic1(config-member)# exit
```

バーチャルポートチャネルインターフェイスの場合は、次のように入力します。

```
apic1(config-member)# interface vpc vpcl leaf 101 102
apic1(config-member)# exit
```

FEX バーチャルポートチャネルインターフェイスの場合は、次のように入力します。

```
apic1(config-member)# interface vpc vpcl leaf 101 102 fex 101 102
apic1(config-member)# exit
```

ステップ8 メンバーに vNIC を追加します。

```
vnic "vnic_name"
```

vNIC の代わりにインターフェイスを追加する場合は、前のステップを参照してください。

パラメータ	説明
vnic	クラスタデバイスの仮想マシンのvNICアダプタの名前。名前を二重引用符で囲みます。

例：

```
apic1(config-member)# vnic "Network adapter 2"
apic1(config-member)# exit
```

NX-OS スタイルの CLI を使用したハイ アベイラビリティ クラスタの作成

ステップ9 デバイスの作成が完了したら、コンフィギュレーションモードを終了します。

例：

```
apic1(config-cluster-interface)# exit
apic1(config-cluster)# exit
apic1(config-tenant)# exit
apic1(config)# exit
```

NX-OS スタイルの CLI を使用したハイ アベイラビリティ クラスタの作成

次に、NX-OS スタイルの CLI を使用してハイ アベイラビリティ クラスタを作成する手順の例を示します。

手順

ステップ1 コンフィギュレーションモードを開始します。

例：

```
apic1# configure
```

ステップ2 テナントのコンフィギュレーションモードを開始します。

```
tenant tenant_name
```

例：

```
apic1(config)# tenant t1
```

ステップ3 クラスタを作成します。

例：

```
apic1(config-tenant)# l4l7 cluster name ifav108-asa type physical vlan-domain phyDom5 servicetype FW
```

ステップ4 クラスタデバイスを追加します。

例：

```
apic1(config-cluster)# cluster-device C1
apic1(config-cluster)# cluster-device C2
```

ステップ5 プロバイダークラスタインターフェイスを追加します。

例：

```
apic1(config-cluster)# cluster-interface provider vlan 101
```

ステップ6 インターフェイスにメンバーデバイスを追加します。

例：

```
apic1(config-cluster-interface)# member device C1 device-interface Po1
apic1(config-member)# interface vpc VPCPolASA leaf 103 104
apic1(config-member)# exit
apic1(config-cluster-interface)# exit
apic1(config-cluster-interface)# member device C2 device-interface Po2
apic1(config-member)# interface vpc VPCPolASA-2 leaf 103 104
apic1(config-member)# exit
apic1(config-cluster-interface)# exit
```

ステップ7 別のプロバイダー クラスタ インターフェイスを追加します。

例：

```
apic1(config-cluster)# cluster-interface provider vlan 102
```

ステップ8 最初のインターフェイスからこの新しいインターフェイスに同じメンバーデバイスを追加します。

例：

```
apic1(config-cluster-interface)# member device C1 device-interface Po1
apic1(config-member)# interface vpc VPCPolASA leaf 103 104
apic1(config-member)# exit
apic1(config-cluster-interface)# exit
apic1(config-cluster-interface)# member device C2 device-interface Po2
apic1(config-member)# interface vpc VPCPolASA-2 leaf 103 104
apic1(config-member)# exit
apic1(config-cluster-interface)# exit
```

ステップ9 クラスタ作成モードを終了します。

例：

```
apic1(config-cluster)# exit
```

NX-OS スタイルの CLI を使用した仮想デバイスの作成

次に、NX-OS スタイルの CLI を使用して仮想デバイスを作成する手順の例を示します。

手順

ステップ1 コンフィギュレーション モードを開始します。

例：

```
apic1# configure
```

ステップ2 テナントのコンフィギュレーション モードを開始します。

```
tenant tenant_name
```

例：

```
apic1(config)# tenant t1
```

ステップ3 クラスタを作成します。

論理デバイスを作成する XML の例

例 :

```
apic1(config-tenant)# 1417 cluster name ifav108-citrix type virtual vlan-domain ACIVswitch servicetype
ADC
```

ステップ4 クラスタデバイスを追加します。

例 :

```
apic1(config-cluster)# cluster-device D1 vcenter ifav108-vcenter vm NSVPX-ESX
```

ステップ5 コンシューマクラスタインターフェイスを追加します。

例 :

```
apic1(config-cluster)# cluster-interface consumer
```

ステップ6 コンシューマインターフェイスにメンバーデバイスを追加します。

例 :

```
apic1(config-cluster-interface)# member device D1 device-interface 1_1
apic1(config-member)# interface ethernet 1/45 leaf 102
ifav108-apic1(config-member)# vnic "Network adapter 2"
apic1(config-member)# exit
apic1(config-cluster-interface)# exit
```

ステップ7 プロバイダークラスタインターフェイスを追加します。

例 :

```
apic1(config-cluster)# cluster-interface provider
```

ステップ8 プロバイダーインターフェイスに同じメンバーデバイスを追加します。

例 :

```
apic1(config-cluster-interface)# member device D1 device-interface 1_1
apic1(config-member)# interface ethernet 1/45 leaf 102
ifav108-apic1(config-member)# vnic "Network adapter 2"
apic1(config-member)# exit
apic1(config-cluster-interface)# exit
```

ステップ9 クラスタ作成モードを終了します。

例 :

```
apic1(config-cluster)# exit
```

論理デバイスを作成する XML の例

LDevVip オブジェクトを作成する XML の例

次の XML の例では、LDevVip オブジェクトを作成します。

```
<polUni>
  <fvTenant name="HA_Tenant1">
    <vnsLDevVip name="ADCCluster1" devtype="VIRTUAL" managed="no">
```

```

        <vnsRsALDevToDomP tDn="uni/vmmp-VMware/dom-mininet"/>
    </vnsLDevVip>
</fvTenant>
</polUni>

```

Cisco ACI Virtual Edge の場合、次の XML の例では、スイッチングモードが `ave` である Cisco ACI Virtual Edge VMM ドメインに関連付けられた `LDevVip` オブジェクトが作成されます。

```

<polUni>
    <fvTenant name="HA_Tenant1">
        <vnsLDevVip name="ADCCluster1" devtype="VIRTUAL" managed="no">
            <vnsRsALDevToDomP switchingMode="AVE" tDn="uni/vmmp-VMware/dom-mininet_ave"/>
        </vnsLDevVip>
    </fvTenant>
</polUni>

```

AbsNode オブジェクトを作成する XML の例

次の XML の例では、`AbsNode` オブジェクトを作成します。

```

<fvTenant name="HA_Tenant1">
    <vnsAbsGraph name="g1">
        <vnsAbsTermNodeProv name="Input1">
            <vnsAbsTermConn name="C1">
            </vnsAbsTermConn>
        </vnsAbsTermNodeProv>

        <!-- Node1 provides a service function -->
        <vnsAbsNode name="Node1" managed="no">
            <vnsAbsFuncConn name="outside" >
            </vnsAbsFuncConn>
            <vnsAbsFuncConn name="inside" >
            </vnsAbsFuncConn>
        </vnsAbsNode>

        <vnsAbsTermNodeCon name="Output1">
            <vnsAbsTermConn name="C6">
            </vnsAbsTermConn>
        </vnsAbsTermNodeCon>

        <vnsAbsConnection name="CON2" >
            <vnsRsAbsConnectionConns
                tDn="uni/tn-HA_Tenant1/AbsGraph-g1/AbsTermNodeCon-Output1/AbsTConn"/>
            <vnsRsAbsConnectionConns
                tDn="uni/tn-HA_Tenant1/AbsGraph-g1/AbsNode-Node1/AbsFConn-outside"/>
        </vnsAbsConnection>

        <vnsAbsConnection name="CON1" >
            <vnsRsAbsConnectionConns
                tDn="uni/tn-HA_Tenant1/AbsGraph-g1/AbsNode-Node1/AbsFConn-inside"/>
            <vnsRsAbsConnectionConns
                tDn="uni/tn-HA_Tenant1/AbsGraph-g1/AbsTermNodeProv-Input1/AbsTConn"/>
        </vnsAbsConnection>
    </vnsAbsGraph>
</fvTenant>

```

■ レイヤ4～レイヤ7サービスのエンドポイントグループとコネクタを関連付ける XML の例

レイヤ4～レイヤ7サービスのエンドポイントグループとコネクタを関連付ける XML の例

次に、レイヤ4～レイヤ7サービスのエンドポイントグループとコネクタを関連付ける XML の例を示します。

```
<fvTenant name="HA_Tenant1">
  <vnslDevCtx ctrctNameOrLbl="any" descr="" dn="uni/tn-HA_Tenant1/lDevCtx-c-any-g-any-n-any" graphNameOrLbl="any" name="" nodeNameOrLbl="any">
    <vnslDevCtxToLDev tDn="uni/tn-HA_Tenant1/lDevVip-ADCCluster1"/>
    <vnslIfCtx connNameOrLbl="inside" descr="" name="inside">
      <vnslIfCtxToSvcEPg tDn="uni/tn-HA_Tenant1/ap-sap/SvcEPg-EPG1"/>
      <vnslIfCtxToBD tDn="uni/tn-HA_Tenant1/BD-provBD1"/>
      <vnslIfCtxToLIf tDn="uni/tn-HA_Tenant1/lDevVip-ADCCluster1/lIf-inside"/>
    </vnslIfCtx>
    <vnslIfCtx connNameOrLbl="outside" descr="" name="outside">
      <vnslIfCtxToSvcEPg tDn="uni/tn-HA_Tenant1/ap-sap/SvcEPg-EPG2"/>
      <vnslIfCtxToBD tDn="uni/tn-HA_Tenant1/BD-consBD1"/>
      <vnslIfCtxToLIf tDn="uni/tn-HA_Tenant1/lDevVip-ADCCluster1/lIf-outside"/>
    </vnslIfCtx>
  </vnslDevCtx>
</fvTenant>
```

レイヤ4～レイヤ7サービスのエンドポイントグループで静的なカプセル化を使用する XML の例

次の XML の例では、レイヤ4～レイヤ7サービスエンドポイントグループで静的なカプセル化を使用しています。

```
<polUni>
  <fvTenant name="HA_Tenant1">
    <fvAp name="sap">
      <vnssSvcEPg name="EPG1" encaps="vlan-3510">
        </vnssSvcEPg>
      </fvAp>
    </fvTenant>
  </polUni>
```

GUI を使用したデバイスの変更

デバイスを作成した後で、そのデバイスを変更することができます。



(注)

デバイスを作成するか、または既存のクラスタにデバイスを追加するには、「デバイスの作成」の手順を使用する必要があります。

手順

ステップ1 メニューバーで、[Tenants] > [All Tenants] の順に選択します。

ステップ2 [Work] ペインで、テナントの名前をダブルクリックします。

ステップ3 [Navigation] ウィンドウで、**Tenant *tenant_name* > Services > L4-L7 > Devices > *device_name*** を選択します。
[Work] ウィンドウにデバイスに関する情報が表示されます。

ステップ4 **General** セクションではいくつかのパラメータを変更するコ音ができます。

Device 1 セクションでは、インターフェイスの追加、または既存のインターフェイスのパスの変更を行えます。インターフェイスを追加するには、+ボタンをクリックします。パスを変更するには、変更するパスをダブルクリックします。

ステップ5 パラメータを変更した後、**Submit** をクリックします。

GUI を使用してレイヤ 7 仮想 ASA デバイスにレイヤ 4 でのトランкиングを有効化

次の手順では、GUI を使用したレイヤ 7 仮想 ASA デバイスにレイヤ 4 でのトランкиングが有効にします。

始める前に

- ASA デバイスの仮想レイヤ 7 にレイヤ 4 に設定した必須。

手順

ステップ1 メニューバーで、[Tenants] > [All Tenants] の順に選択します。

ステップ2 [Work] ペインで、テナントの名前をダブルクリックします。

ステップ3 [Navigation] ウィンドウで、**Tenant *tenant_name* > Services > L4-L7 > Devices > *device_name*** を選択します。

ステップ4 [Work] ウィンドウで、**Trunking Port** チェック ボックスをオンにします。

ステップ5 [Submit] をクリックします。

■ REST API を使用してレイヤ 7 仮想 ASA デバイスにレイヤ 4 でのランキングを有効化

REST API を使用してレイヤ 7 仮想 ASA デバイスにレイヤ 4 でのランキングを有効化

次の手順では、REST API を使用して、レイヤ 7 仮想の ASA デバイスにレイヤ 4 でのランキングを有効にする例を示します。

始める前に

- ASA デバイスの仮想レイヤ 7 にレイヤ 4 に設定した必須。

手順

名前付きレイヤ 7 デバイスにレイヤ 4 でのランキングを有効にする `InsiemeCluster` :

```
<polUni>
  <fvTenant name="tenant1">
    <vnsLDevVip name="InsiemeCluster" devtype="VIRTUAL" trunking="yes">
      ...
      ...
    </vnsLDevVip>
  </fvTenant>
</polUni>
```

REST API とともにインポートされたデバイスの使用

次の REST API ではインポートされたデバイスを使用します。

```
<polUni>
  <fvTenant dn="uni/tn-tenant1" name="tenant1">
    <vnsLDevIf ldev="uni/tn-mgmt/lDevVip-ADCCluster1"/>
    <vnsLDevCtx ctrctNameOrLbl="any" graphNameOrLbl="any" nodeNameOrLbl="any">
      <vnsRsLDevCtxToLDev tDn="uni/tn-tenant1/lDevIf-[uni/tn-mgmt/lDevVip-ADCCluster1]"/>

      <vnsLIfCtx connNameOrLbl="inside">
        <vnsRsLIfCtxToLIf
          tDn="uni/tn-tenant1/lDevIf-[uni/tn-mgmt/lDevVip-ADCCluster1]/lDevIfLIf-inside"/>
          <fvSubnet ip="10.10.10.10/24"/>
            <vnsRsLIfCtxToBD tDn="uni/tn-tenant1/BD-tenant1BD1"/>
        </vnsLIfCtx>
        <vnsLIfCtx connNameOrLbl="outside">
          <vnsRsLIfCtxToLIf
            tDn="uni/tn-tenant1/lDevIf-[uni/tn-mgmt/lDevVip-ADCCluster1]/lDevIfLIf-outside"/>
            <fvSubnet ip="70.70.70.70/24"/>
              <vnsRsLIfCtxToBD tDn="uni/tn-tenant1/BD-tenant1BD4"/>
            </vnsLIfCtx>
          </vnsLIfCtx>
        </fvTenant>
      </polUni>
```

NX-OS スタイルの CLI を使用した別のテナントからのデバイスの作成

共有サービスのシナリオでは、別のテナントからデバイスをインポートできます。

手順

ステップ1 コンフィギュレーションモードを開始します。

例：

```
apic1# configure
```

ステップ2 テナントのコンフィギュレーションモードを開始します。

```
tenant tenant_name
```

例：

```
apic1(config)# tenant t1
```

ステップ3 デバイスをインポートします。

```
1417 cluster import-from tenant_name device-cluster device_name
```

パラメータ	説明
import-from	デバイスのインポート元のテナントの名前。
device-cluster	指定したテナントからインポートするデバイスクラスタの名前。

例：

```
apic1(config-tenant)# 1417 cluster import-from common device-cluster d1
apic1(config-import-from)# end
```

GUI を使用したデバイスのインポートの確認

GUI を使用して、デバイスが正常にインポートされたことを確認することができます。

手順

ステップ1 メニュー バーで、[Tenants] > [All Tenants] の順に選択します。

ステップ2 [Work] ペインで、テナントの名前をダブルクリックします。

■ GUI を使用したデバイスのインポートの確認

ステップ3 [Navigation] ウィンドウで、Tenant *tenant_name* > Services > L4-L7 > Imported Devices > *device_name* を選択します。

デバイス情報が [Work] ペインに表示されます。

翻訳について

このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。