About Deploying a Three-Tier Application with Transit Topology

 Typically, the APIC fabric hosts a three-tier application within a tenant network. In this example, the application is implemented by using three servers (a web server, an application server, and a database server). See the following figure for an example of a three-tier application.
The web server has the HTTP filter, the application server has the Remote Method Invocation (RMI) filter, and the database server has the Structured Query Language (SQL) filter. The application server consumes the SQL contract to communicate with the database server. The web server consumes the RMI contract to communicate with the application server. The traffic enters from the web server and communicates with the application server. The application server then communicates with the database server, and the traffic can also communicate externally.

To deploy the three-tier application, you must create the required EPGs, filters, and contracts.

A filter specifies the data protocols to be allowed or denied by a contract that contains the filter. A contract can contain multiple subjects. A subject can be used to realize uni- or bidirectional filters. A unidirectional filter is a filter that is used in one direction, either from consumer-to-provider (IN) or from provider-to-consumer (OUT) filter. A bidirectional filter is the same filter that is used in both directions. It is not reflexive.

Contracts are policies that enable inter-End Point Group (inter-EPG) communication. These policies are the rules that specify communication between application tiers. If no contract is attached to the EPG, inter-EPG communication is disabled by default. No contract is required for intra-EPG communication because intra-EPG communication is always allowed.
About Transit Routing

Transit routing enables border routers to perform bidirectional redistribution with other routing domains. Bidirectional redistribution passes routing information from one routing domain to another. Such redistribution lets the ACI fabric provide full IP connectivity between different routing domains. Doing so can also provide redundant connectivity by enabling backup paths between routing domains. For more information, see "ACI Transit Routing" in the *Cisco ACI Fundamentals Guide*.

Deploying a Three-Tier Application

Configure the tenant VRF and bridge domain.

```
apic1(config)# tenant t1
apic1(config-tenant)# vrf context v1
apic1(config-tenant-vrf)# contract enforce
apic1(config-tenant)# bridge-domain b1
apic1(config-tenant-bd)# vrf member v1
apic1(config-tenant)# interface bridge-domain b1
apic1(config-tenant-interface)# ip address 159.10.10.1/24 scope public
apic1(config-tenant-interface)# exit
```

Configure three EPGs: web, app, and db.

```
apic1(config-tenant)# application retail
apic1(config-tenant-app)# epg web
apic1(config-tenant-app-epg)# bridge-domain member b1
apic1(config-tenant-app-epg)# contract provider web
apic1(config-tenant-app-epg)# contract consumer app
apic1(config-tenant-app)# epg app
apic1(config-tenant-app-epg)# bridge-domain member b1
apic1(config-tenant-app-epg)# contract provider app
apic1(config-tenant-app-epg)# contract consumer db
apic1(config-tenant-app)# epg db
apic1(config-tenant-app-epg)# bridge-domain member b1
apic1(config-tenant-app-epg)# contract provider db
```

Configure VLAN domain.

```
apic1(config)# vlan-domain dom100
apic1(config-vlan)# vlan 100-200
```

Create port-channel and deploy the web EPG.

```
apic1(config)# leaf 101
apic1(config-leaf)# interface ethernet 1/2-5
apic1(config-leaf-if)# channel-group pol
apic1(config-leaf)# interface port-channel pol
apic1(config-leaf-if)# vlan-domain member dom100
apic1(config-leaf-if)# switchport trunk allowed vlan 101 tenant t1 application retail epg web
```

Create a vPC and deploy app and db EPGs.
Configure MP-BGP.

apic1(config)# bgp-fabric
apic1(config-bgp-fabric)# asn 100
apic1(config-bgp-fabric)# route-reflector spine 104,105

Configure External-l3 EPG.

apic1(config-tenant)# external-l3 epg l3epg1
apic1(config-tenant-l3ext-epg)# vrf member v1
apic1(config-tenant-l3ext-epg)# match ip 173.10.1.0/24
apic1(config-tenant-l3ext-epg)# contract consumer web

Configure VRF on Leaf, route-map and deploy external-l3 EPG.

apic1(config)# leaf 103
apic1(config-leaf)# vrf context tenant t1 vrf v1
apic1(config-leaf-vrf)# external-l3 epg l3epg1
apic1(config-leaf-vrf)# route-map map1
apic1(config-leaf-vrf-route-map)# match bridge-domain b1

Configure OSPF area on a sub-Interface.

apic1(config-leaf)# router ospf default
apic1(config-leaf-ospf)# vrf member tenant t1 vrf v1
apic1(config-leaf-ospf-vrf)# area 0.0.0.1 route-map map1 out
apic1(config-leaf)# interface ethernet 1/2
apic1(config-leaf-if)# no switchport
apic1(config-leaf-if)# vlan-domain member dom1
apic1(config-leaf)# interface ethernet 1/2.150
apic1(config-leaf-if)# vrf member tenant t1 vrf v1
apic1(config-leaf-if)# ip address 169.10.10.1/24
apic1(config-leaf-if)# ip router ospf default area 0.0.0.1

Configure filters.

apic1(config-tenant)# access-list http
apic1(config-tenant-acl)# match tcp dest 80
apic1(config-tenant-acl)# match tcp dest 443
apic1(config-tenant)# access-list rmi
Transit Routing with OSPF and BGP

This procedure configures transit routing between Site1 and Site2 for the three-tier application described in *Deploying a Three-Tier Application* in this chapter.

Configure External-l3 EPG (13epg2) for Site2.

```
apic1(config-tenant)# external-l3 epg 13epg2
apic1(config-tenant-l3ext-epg)# vrf member v1
apic1(config-tenant-l3ext-epg)# match ip 174.10.1.0/24
apic1(config-tenant-l3ext-epg)# contract consumer transit
apic1(config)# leaf 102
apic1(config-leaf)# vrf context tenant t1 vrf v1
apic1(config-leaf-vrf)# external-l3 epg 13epg2
```

Configure BGP connectivity over External SVI and export route corresponding to Site1.

```
apic1(config)# leaf 102
apic1(config-leaf-vrf)# route-map map200
apic1(config-leaf-vrf-route-map)# ip prefix-list p1 match 173.10.1.0/24
apic1(config-leaf-vrf-route-map)# contract consumer transit
apic1(config-leaf)# interface vlan 160
apic1(config-leaf-if)# vrf member tenant t1 vrf v1
apic1(config-leaf-if)# ip address 208.1.1.2/24
apic1(config-leaf-if)# interface ethernet 1/11
apic1(config-leaf-if)# vlan-domain member dom1
apic1(config-leaf-if)# switchport trunk allowed vlan 160 tenant t1 external-svi
apic1(config-leaf)# router bgp 100
apic1(config-bgp)# vrf member tenant t1 vrf v1
apic1(config-leaf-bgp-vrf)# neighbor 208.1.1.1
apic1(config-leaf-bgp-vrf-neighbor)# remote-as 200
apic1(config-leaf-bgp-vrf-neighbor)# update-source vlan 160
```
Configure contract provider on l3epg1 (Site1) to establish connection with l3epg2 (Site2).

```bash
apic1(config-tenant-l3ext-epg)# contract provider transit
```

Configure a route-map on Site1 to export the route corresponding to Site2.

```bash
apic1(config-leaf-vrf-route-map-match)# set metric 100
```

Configure ACL and contract for transit routing.

```bash
apic1(config-tenant-contract-subj)# access-group acl1 both
```