
cisco Live!
Let's go
#CiscoLive

The bridge to possible

ACI Objects

How to avoid getting your configuration wires crossed

Chris Merkel – Technical Solutions Architect
Davis Creemer – Technical Solutions Architect

BRKDCN-2647

cisco Live!

Cisco Webex App

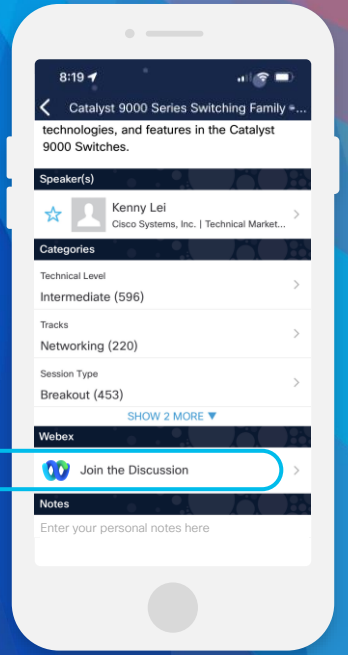
Questions?

Use Cisco Webex App to chat
with the speaker after the session

How

1. Find this session in the Cisco Live Mobile App
2. Click “Join the Discussion”
3. Install the Webex App or go directly to the Webex space
4. Enter messages/questions in the Webex space

Webex spaces will be moderated by the speaker until June 9, 2023.

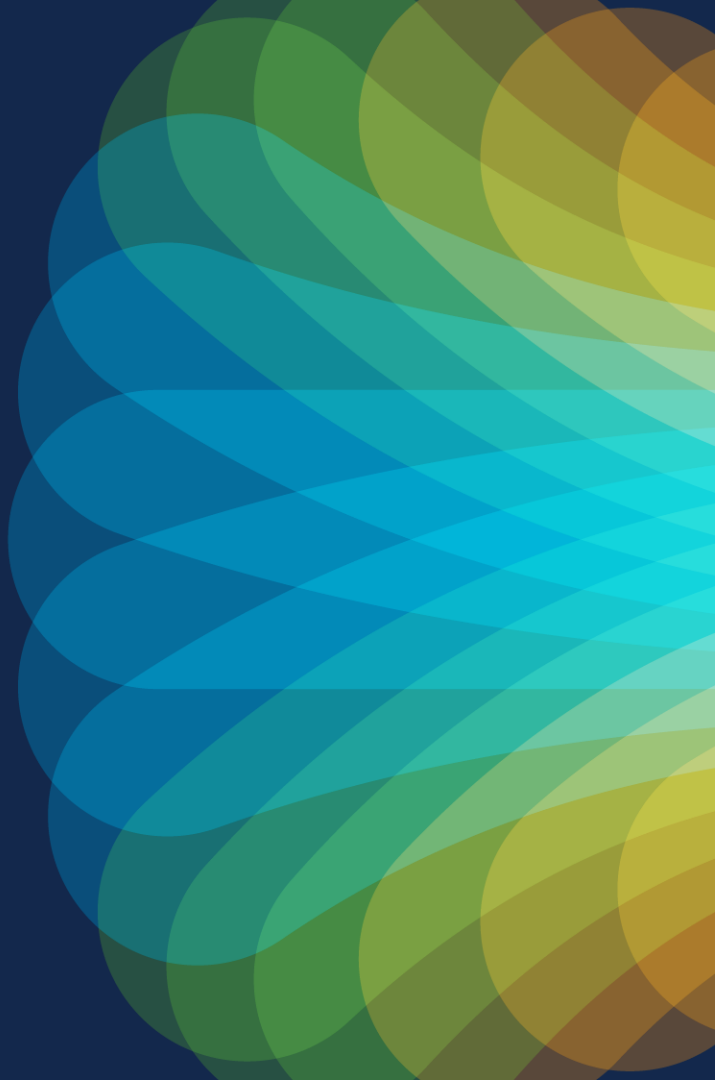


https://ciscolive.ciscoevents.com/ciscolivebot/#BRKDCN-2647

Agenda

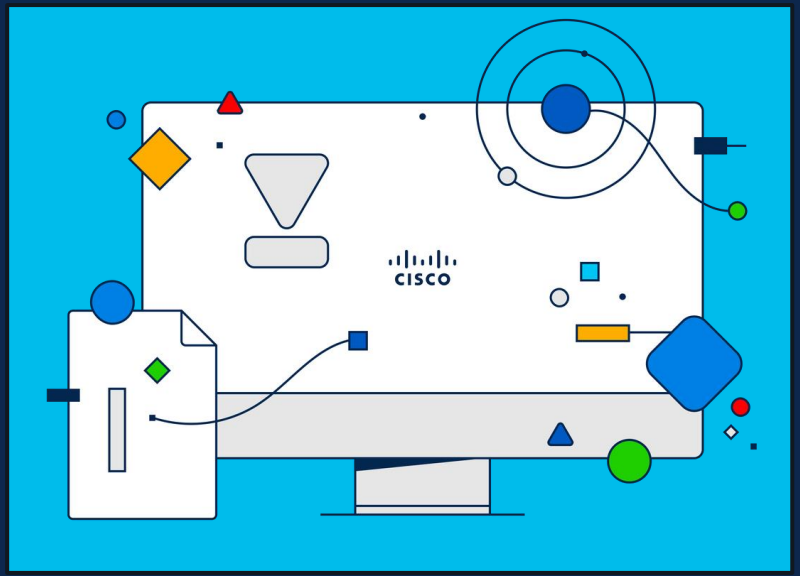
- Introduction
- The ACI Object Model
- General Recommendations
- Fabric Objects
- Tenant Objects

Introduction



Goals for this session

- Gain a general understanding of the ACI Object Model
- Understand how to navigate the object model and see its relationships
- Review commonly used objects to understand how they can be used efficiently
- Gain insights on how to think about creating objects

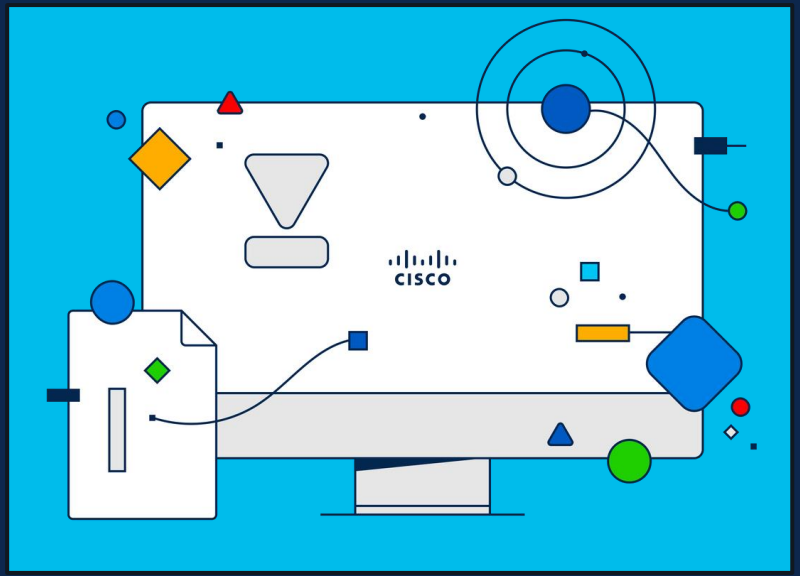


Goals for this session

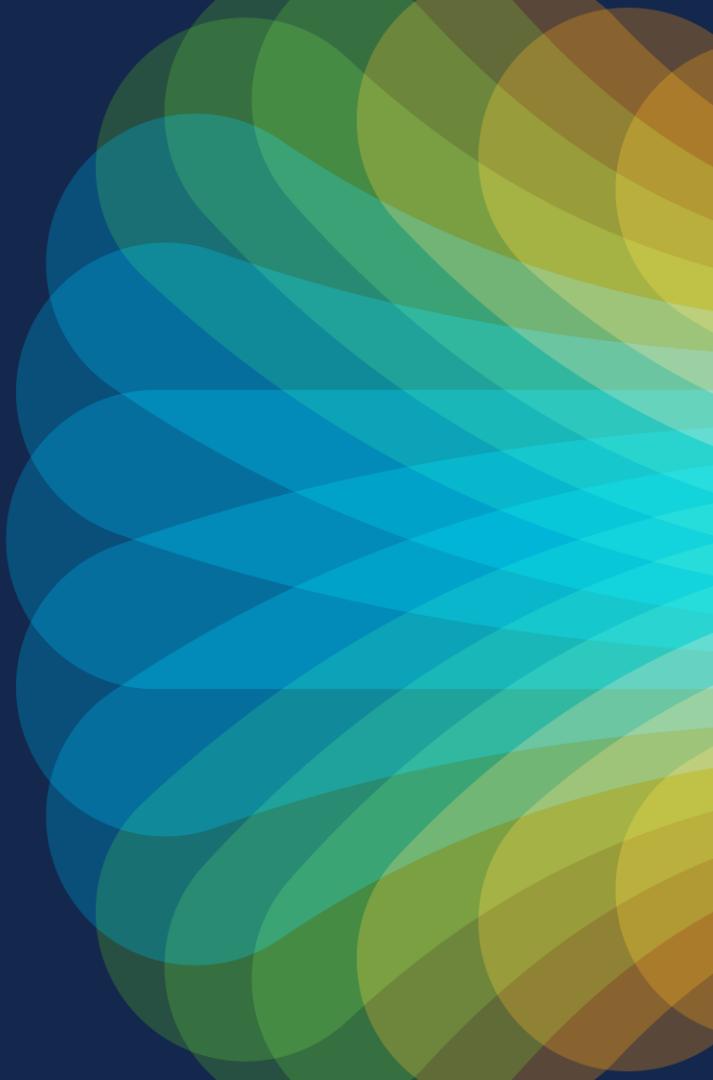
The recommendations from this session are guidance toward best-practice.

If you are not currently following these design practices, you are not “doing it wrong” or “heading toward disaster”.

Do not go back to your hotel room tonight and log into your production fabric to change everything!



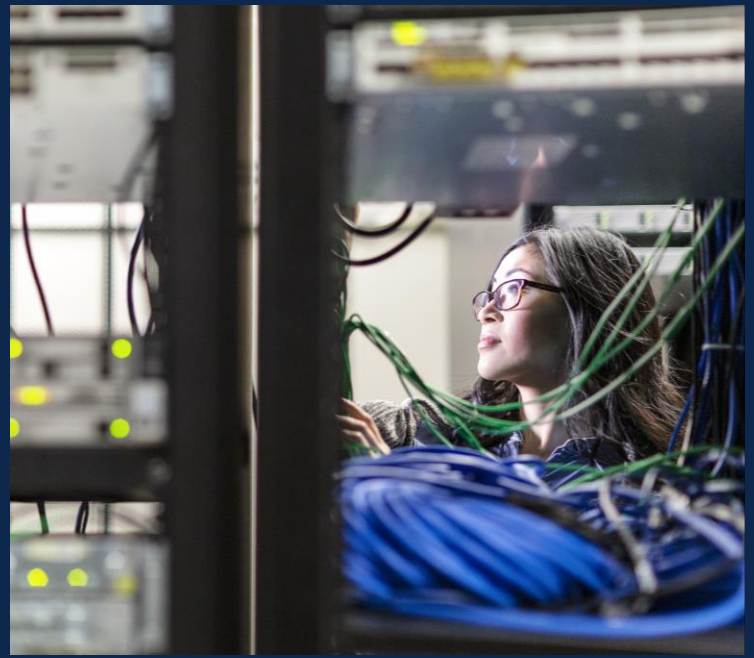
The ACI Object Model



Why are we talking about the ACI Object Model?

ACI maps networking concepts into an object-oriented architecture. This allows the model to encapsulate information and action into small, localized, manageable pieces.

Understanding the scope and scale of what an object knows and does helps to know when an object can be reused and when a new one should be created.

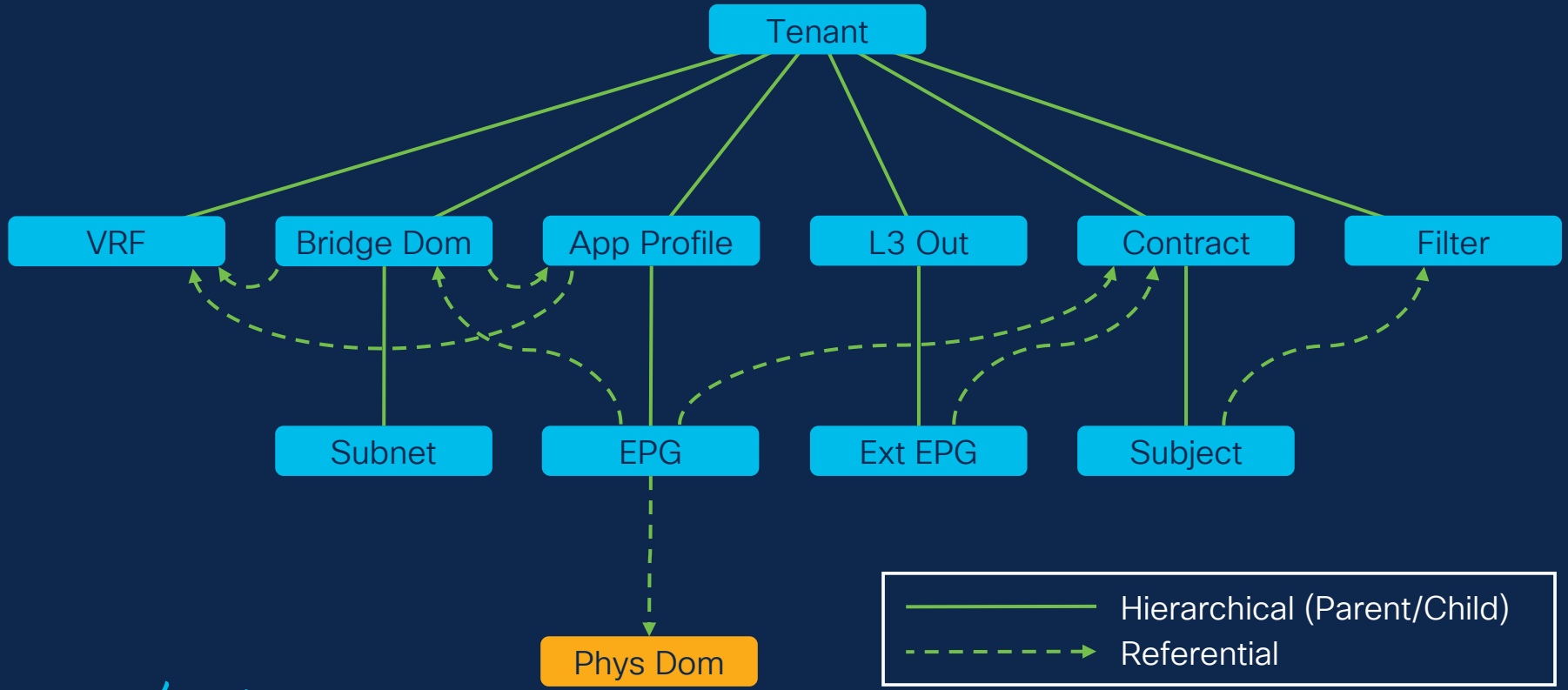


Why are we talking about the ACI Object Model?

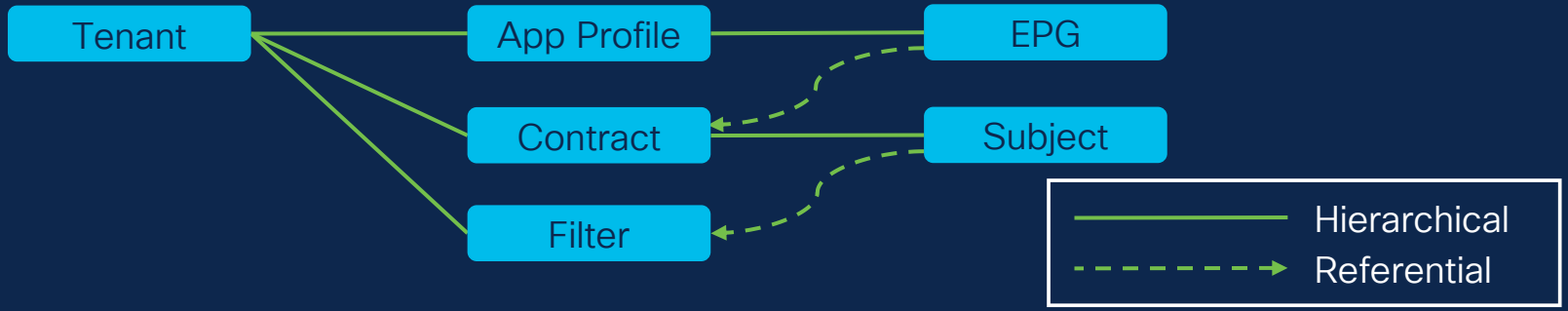
Object sprawl can lead to complexity and death-by-1000-papercuts.

But!.... Think about the impact scope of a change for an object. If this is too broad for comfort, consider creating one or more like objects for fault containment.

ACI Management Information Tree (MIT)



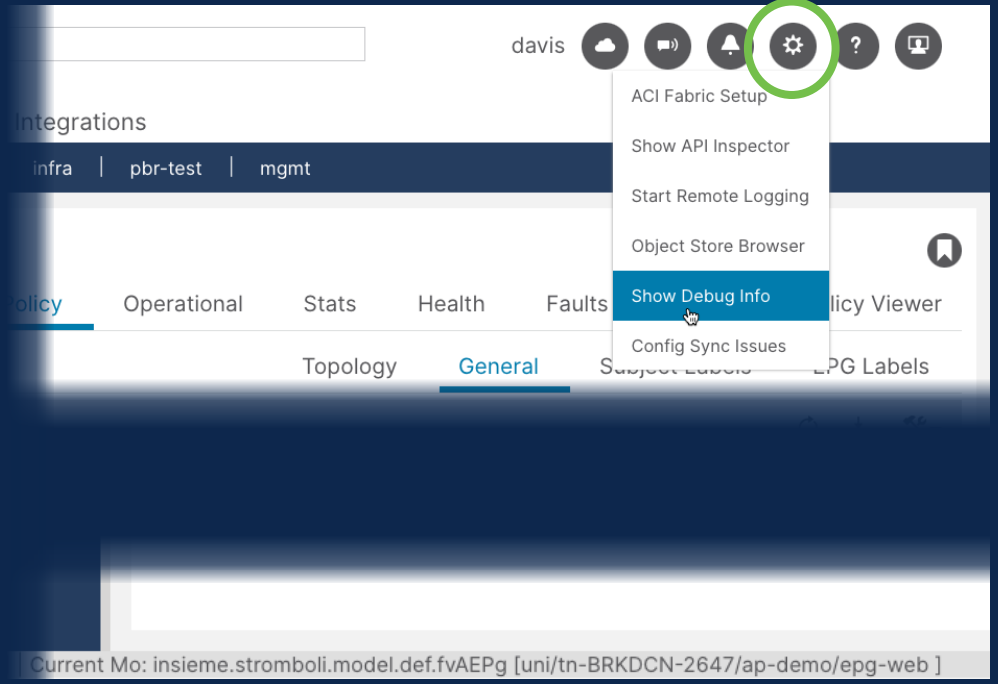
Hierarchical and referential relationships?




```
{
  "fvRsProv": {
    "attributes": {
      "dn": "uni/tn-BRKDCN-2647/ap-demo/epg-db/rsprov-demo-db:mysql",
        "tDn": "uni/tn-BRKDCN-2647/brc-demo-db:mysql",
        "tRn": "brc-demo-db:mysql"
    }
  }
}
```

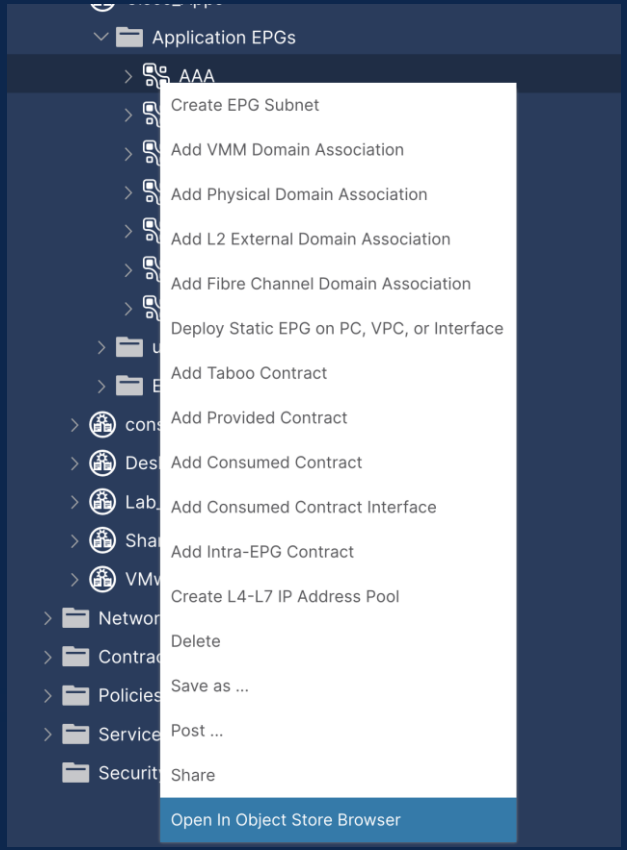
Observing the ACI Object Model

"Show Debug Info"



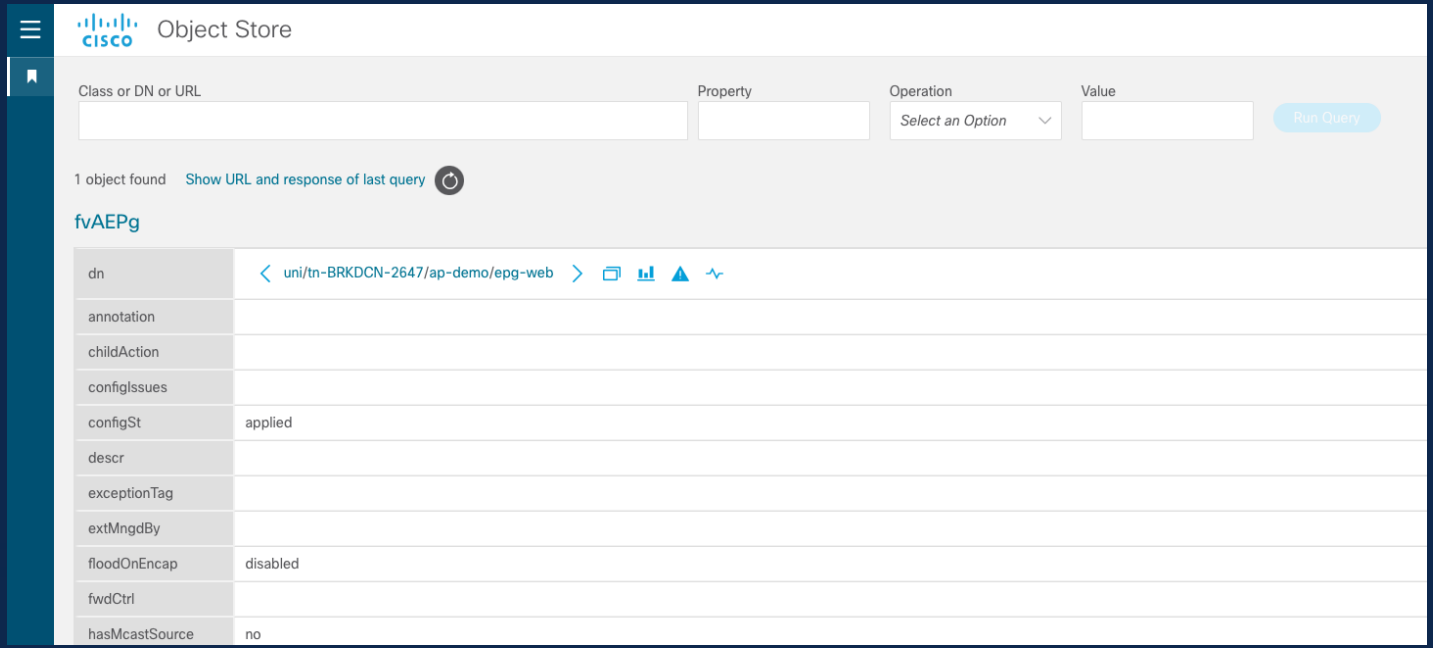
Observing the ACI Object Model

"Object Store Browser"

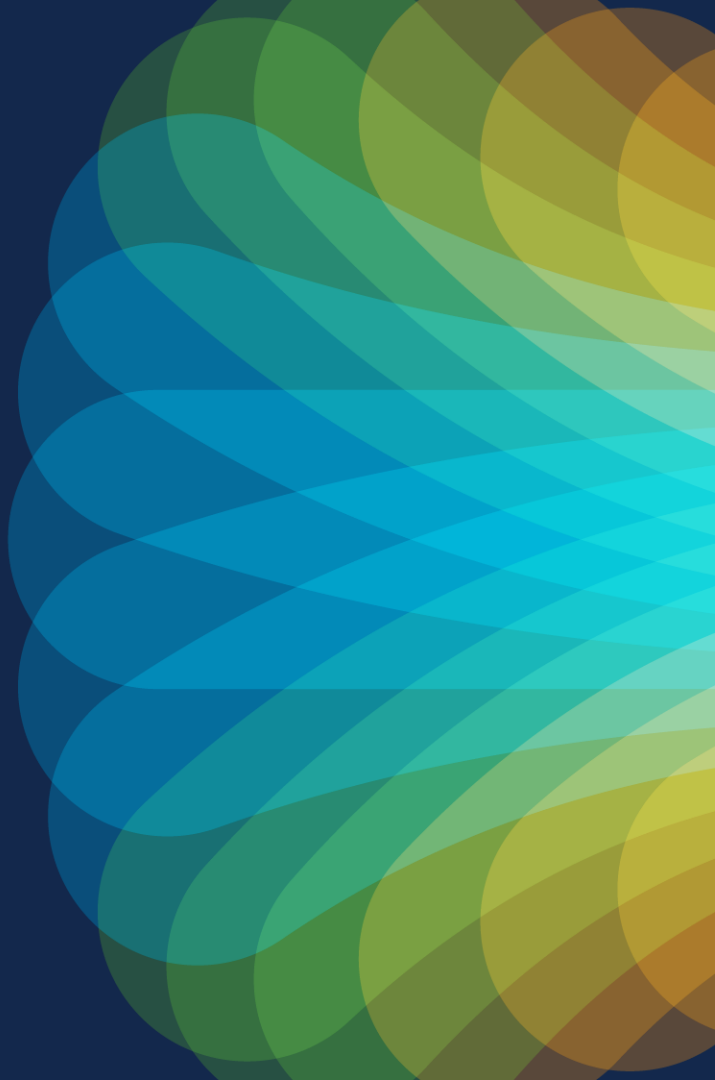


Observing the ACI Object Model

"Object Store Browser"



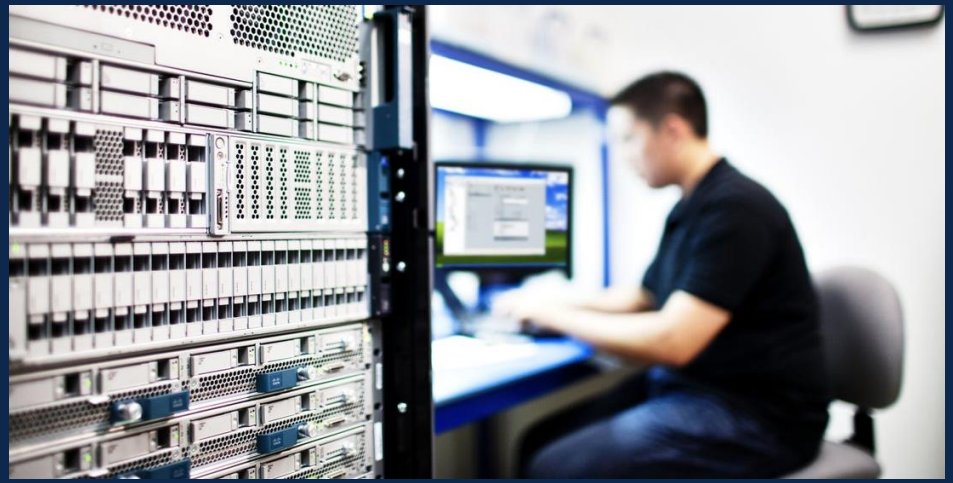
General Recommendations



ACI Design Principals

1. Keep it simple!!!
2. Use ACI the way it was intended to be used.
3. Don’t break assumptions / No surprises.
4. Be consistent.
5. Make it possible to go on vacation!

ACI Object Naming



Naming of objects in ACI affords the ability to make the objects self documenting. Take time to think about this when creating objects and use this to your advantage.

Recommendation: Naming Conventions

Short and simple

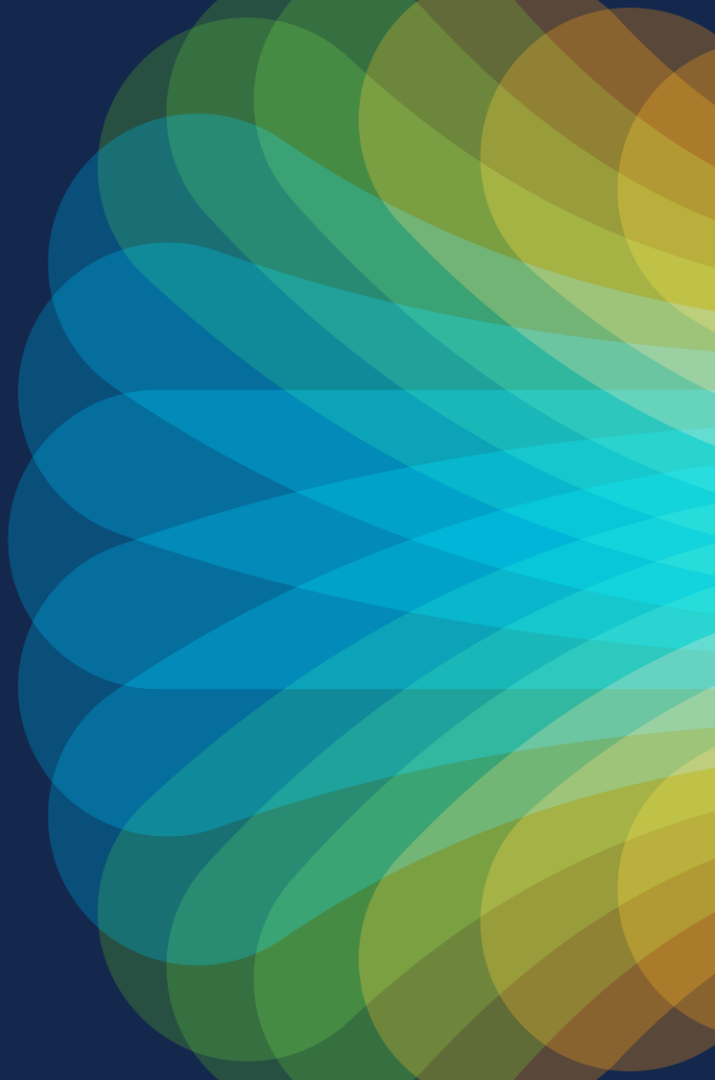
Once created objects cannot be renamed.
Leverage Alias for detailed name.

Prefix/Suffix are Unnecessary

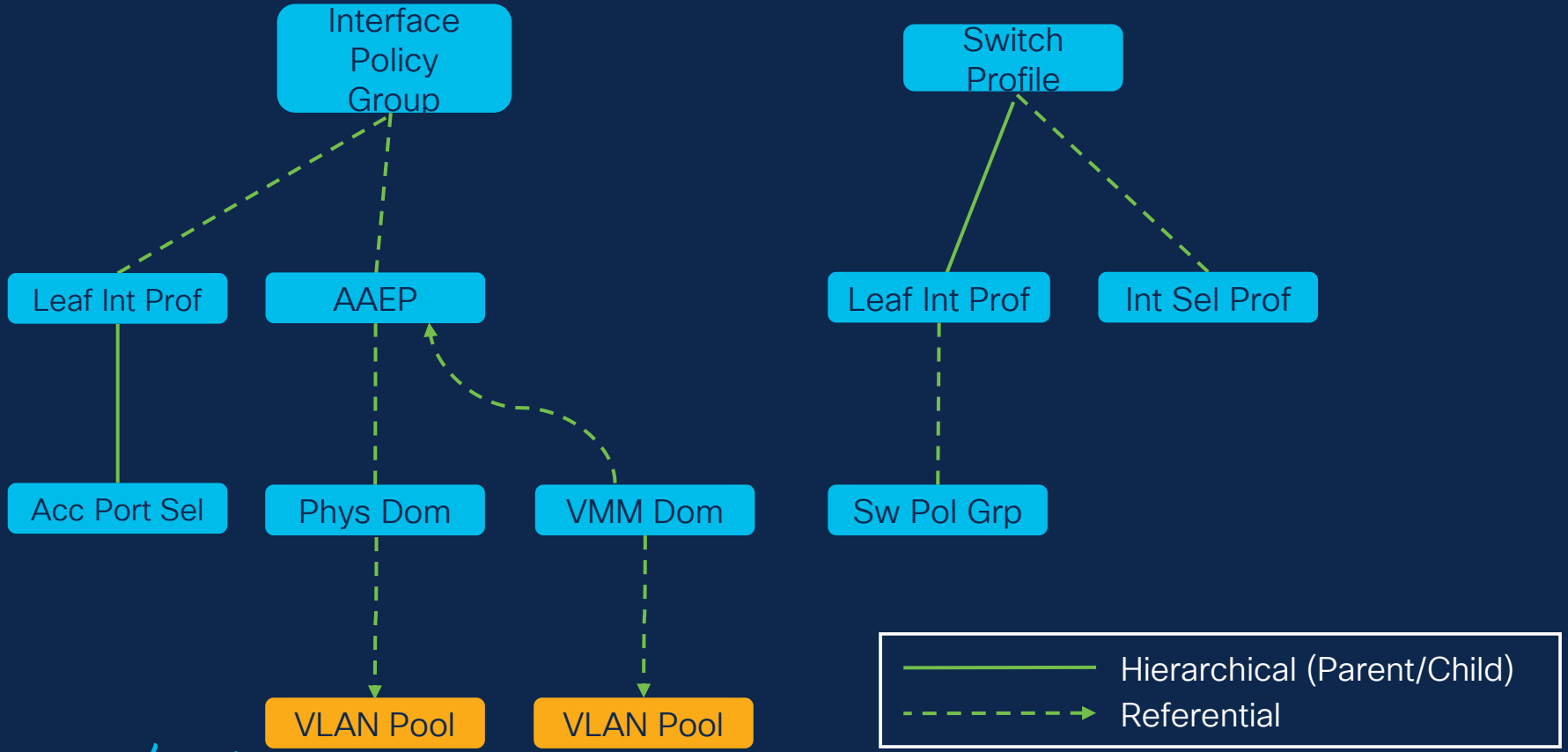
GUI menus already limit your view to object type, and DNs embed prefixes.

e.g., uni/tn-BRKDCN-2647/ap-demo/epg-web

Fabric Objects



Access Policies Object Model



Domains

A domain is used to define the scope of VLANs in the Cisco ACI fabric.

- Think about aligning a Domain with the Attachable Access Entity Profile (AAEP or AEP)
- External Domains – not really used, easier to map L2 trunks directly to EPG – consider using BDPU filter to block TCNs that would flush EPG endpoints
- Configured Access Policies – a way to see all possible ports an EPG could be configured on based on the domain association.
- Can also be used with RBAC to restrict which domains can be attached to an EPG

Domains - Broad

EPG - ND-Data

Client Endpoints | Configured Access Policies | Contracts

Domain	Switch Profile	Attachable Entity Profile	Interface Profile	PC/vPC/If Policy Group	Node	Path Endpoint
static	pod1-leaf-102	static	pod1-leaf-101-102	static-binding-trunk	102	[eth1/46]
static	pod1-leaf-102	static	pod1-leaf-101-102	static-binding-trunk	102	[eth1/46]
static	pod1-leaf-101	static	pod1-leaf-101-102	static-binding-trunk	101	[eth1/46]
static	pod1-leaf-104	prod	pod1-leaf-104	aciTrexDataPorts1G	104	[eth1/17]
static	pod1-leaf-103	prod	pod1-leaf-103	aciTrexDataPorts	103	[eth1/17]
static	pod1-leaf-102	prod	pod1-leaf-101-102	aciTrexDataPorts	102	[eth1/49]
static	pod1-leaf-101	prod	pod1-leaf-101-102	aciTrexDataPorts	101	[eth1/49]
static	pod1-leaf-102	prod	pod1-leaf-101-102	aciTrexDataPorts	102	[eth1/17]
static	pod1-leaf-101	prod	pod1-leaf-101-102	aciTrexDataPorts	101	[eth1/17]
static	pod1-leaf-102	prod	pod1-leaf-101-102	srv_10g	102	[eth1/18]
static	pod1-leaf-101	prod	pod1-leaf-101-102	srv_10g	101	[eth1/18]
static	pod2-leaf-202	prod	pod2-leaf-201-202	srv_10g	202	[eth1/17]
static	pod2-leaf-201	prod	pod2-leaf-201-202	srv_10g	201	[eth1/17]
static	pod1-leaf-102	nexusDashboard	pod1-leaf-101-102	ND-Data	102	[eth1/26]
static	pod1-leaf-102	nexusDashboard	pod1-leaf-101-102	ND-Data	102	[eth1/25]
static	pod1-leaf-101	nexusDashboard	pod1-leaf-101-102	ND-Data	101	[eth1/25]
static	pod1-leaf-102	nexusDashboard	pod1-leaf-101-102	ND-Data	102	[eth1/27]
static	pod1-leaf-101	nexusDashboard	pod1-leaf-101-102	ND-Data	101	[eth1/27]
static	pod1-leaf-103	nexusDashboard	pod1-leaf-103-104	ND-Mgmt	103	[eth1/27]
static	pod1-leaf-104	nexusDashboard	pod1-leaf-103-104	ND-Mgmt	104	[eth1/27]
static	pod1-leaf-103	nexusDashboard	pod1-leaf-103-104	ND-Mgmt	103	[eth1/25]
static	pod1-leaf-104	nexusDashboard	pod1-leaf-103-104	ND-Mgmt	104	[eth1/25]
static	pod1-leaf-103	nexusDashboard	pod1-leaf-103-104	ND-Mgmt	103	[eth1/26]
static	pod1-leaf-104	nexusDashboard	pod1-leaf-103-104	ND-Mgmt	104	[eth1/26]
VMware/hx-sc-dvs	pod1-leaf-103	rtp-ucsm-aep	pod1-leaf-103-104	rtp-ucsm-01-fi-b-vpc	103	[eth1/52]
VMware/hx-sc-dvs	pod1-leaf-103	rtp-ucsm-aep	pod1-leaf-103-104	rtp-ucsm-01-fi-b-vpc	103	[eth1/52]
VMware/hx-sc-dvs	pod1-leaf-103	rtp-ucsm-aep	pod1-leaf-103-104	rtp-ucsm-01-fi-b-vpc	103	[eth1/52]

Domains - Specific

EPG - ND-Mgmt

Summary | Policy | **Operational** | Stats | Health | Fa

Client Endpoints | **Configured Access Policies** | Contracts | Controlle

Domain	Switch Profile	Attachable Entity Profile	Interface Profile	PC/vPC/If Policy Group	Node	Path Endpoint
VMware/hx-sc-...	pod1-leaf-103	rtp-ucsm-aep	pod1-leaf-103-1...	rtp-ucsm-01-fi-...	103	[eth1/52]
VMware/hx-sc-...	pod1-leaf-103	rtp-ucsm-aep	pod1-leaf-103-1...	rtp-ucsm-01-fi-...	103	[eth1/52]
VMware/hx-sc-...	pod1-leaf-103	rtp-ucsm-aep	pod1-leaf-103-1...	rtp-ucsm-01-fi-...	103	[eth1/52]
VMware/hx-sc-...	pod1-leaf-103	rtp-ucsm-aep	pod1-leaf-103-1...	rtp-ucsm-01-fi-...	103	[eth1/52]
VMware/hx-sc-...	pod1-leaf-104	rtp-ucsm-aep	pod1-leaf-103-1...	rtp-ucsm-01-fi-...	104	[eth1/52]
VMware/hx-sc-...	pod2-leaf-202	rtp-ucsm-aep	pod2-leaf-201-...	rtp-ucsm-02-fi-...	202	[eth1/51]
VMware/hx-sc-...	pod2-leaf-201	rtp-ucsm-aep	pod2-leaf-201-...	rtp-ucsm-02-fi-...	201	[eth1/51]
VMware/hx-sc-...	pod1-leaf-102	rtp-ucsm-aep	pod1-leaf-101-1...	rtp-ucsm-01-fi-...	102	[eth1/52]
VMware/hx-sc-...	pod1-leaf-101	rtp-ucsm-aep	pod1-leaf-101-1...	rtp-ucsm-01-fi-...	101	[eth1/52]
VMware/hx-sc-...	pod2-leaf-202	rtp-ucsm-aep	pod2-leaf-201-...	rtp-ucsm-02-fi-...	202	[eth1/52]
VMware/hx-sc-...	pod2-leaf-201	rtp-ucsm-aep	pod2-leaf-201-...	rtp-ucsm-02-fi-...	201	[eth1/52]
nexusDashboard	pod1-leaf-102	nexusDashboard	pod1-leaf-101-1...	ND-Data	102	[eth1/26]
nexusDashboard	pod1-leaf-102	nexusDashboard	pod1-leaf-101-1...	ND-Data	102	[eth1/26]
nexusDashboard	pod1-leaf-101	nexusDashboard	pod1-leaf-101-1...	ND-Data	101	[eth1/26]
nexusDashboard	pod1-leaf-102	nexusDashboard	pod1-leaf-101-1...	ND-Data	102	[eth1/25]
nexusDashboard	pod1-leaf-101	nexusDashboard	pod1-leaf-101-1...	ND-Data	101	[eth1/25]
nexusDashboard	pod1-leaf-102	nexusDashboard	pod1-leaf-101-1...	ND-Data	102	[eth1/27]
nexusDashboard	pod1-leaf-101	nexusDashboard	pod1-leaf-101-1...	ND-Data	101	[eth1/27]
nexusDashboard	pod1-leaf-103	nexusDashboard	pod1-leaf-103-1...	ND-Mgmt	103	[eth1/27]
nexusDashboard	pod1-leaf-104	nexusDashboard	pod1-leaf-103-1...	ND-Mgmt	104	[eth1/27]
nexusDashboard	pod1-leaf-103	nexusDashboard	pod1-leaf-103-1...	ND-Mgmt	103	[eth1/25]
nexusDashboard	pod1-leaf-104	nexusDashboard	pod1-leaf-103-1...	ND-Mgmt	104	[eth1/25]
nexusDashboard	pod1-leaf-103	nexusDashboard	pod1-leaf-103-1...	ND-Mgmt	103	[eth1/26]
nexusDashboard	pod1-leaf-104	nexusDashboard	pod1-leaf-103-1...	ND-Mgmt	104	[eth1/26]

VLAN Pools

A range of VLANs that are available to be programed to switchports on a leaf.

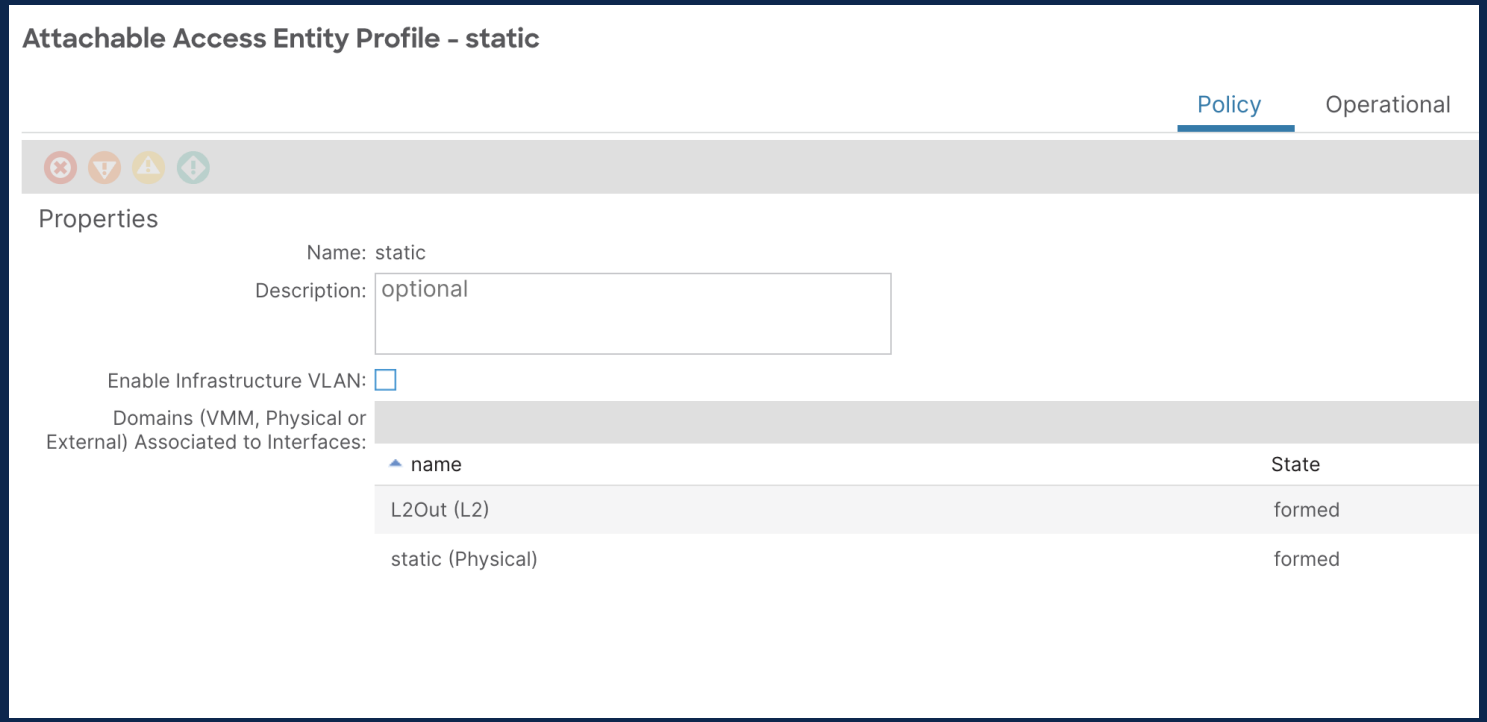
- In general, do not have the same VLAN(s) in multiple pools
- Fewer VLAN pools the better
- Do not create a large Encap Block of VLANs in a pool. A range can be only created or deleted. It cannot be modified.
- Remember, EPGs can support multiple Encaps!
- Sub Interfaces – No VLAN pool needed

AAEP/AEP

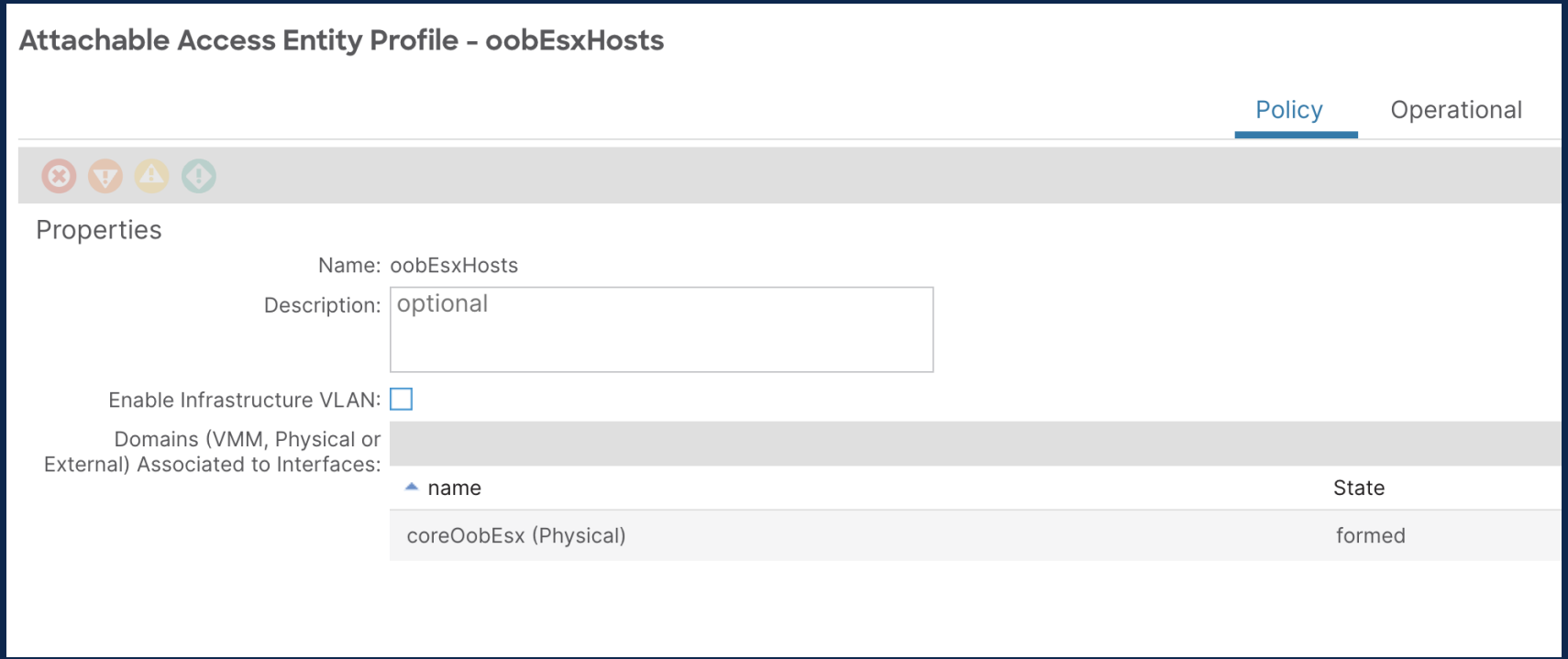
The Attachable Access Entity Profile (AAEP or AEP) is used to map domains (physical or virtual) to one or more physical interfaces, with the end goal of mapping VLANs to interfaces.

- Build AEPs based on a group of ports doing a common function – think VM, SQL or big data cluster
- This allows the use of AEPs to be mapped to EPGs desired for a set of ports. Even if AEP/EPG mapping is not used, this makes the configuration self-documenting
- If the AEP is too broad it will apply EPGs to many more ports than intended.

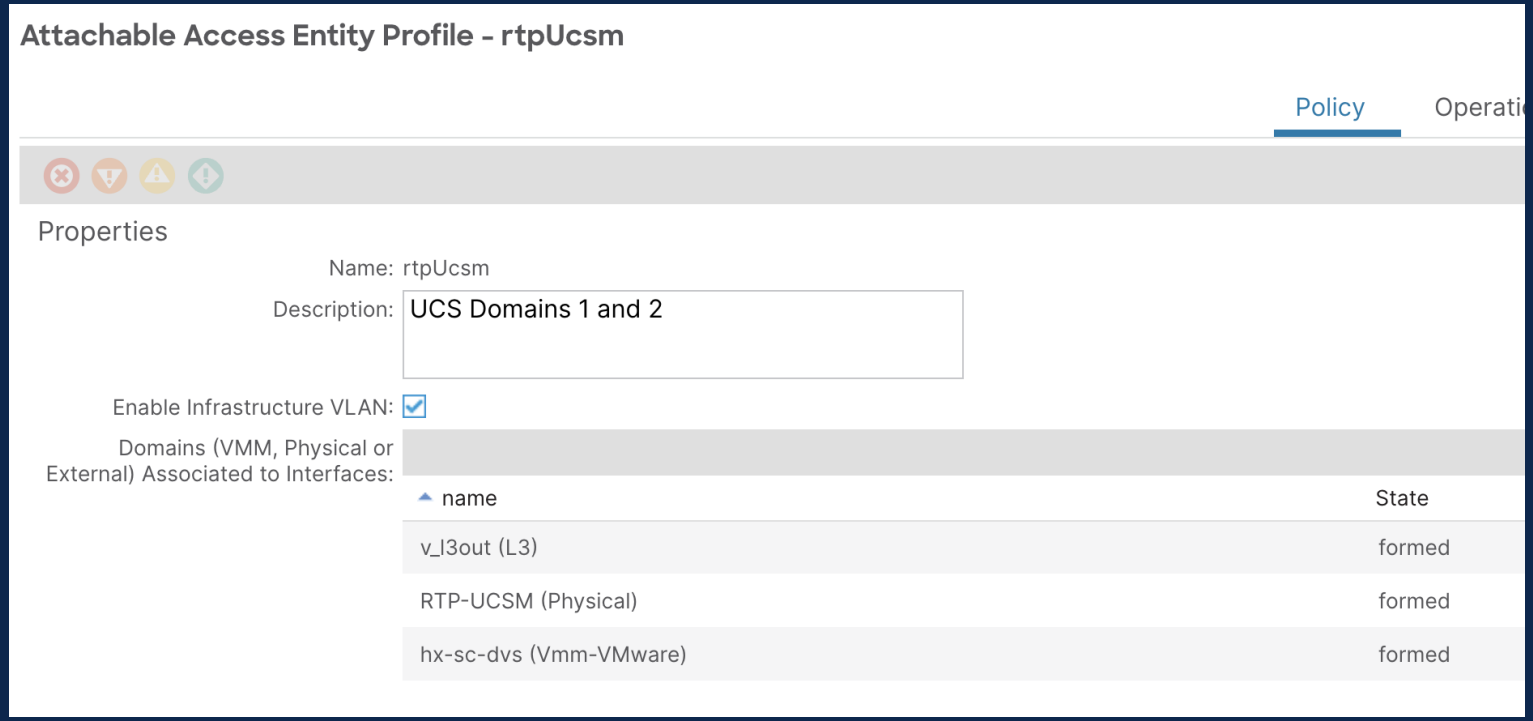
AAEP - Generic



AAEP - Specific



AAEP – UCS Manager



AAEP/AEP/Domain Design Exercise – 75 Rack Servers

20 Misc Bare Metal

10 Big Data

10 Backup

<u>35 VM Hosts</u>
3x Prod Clusters – 7 hosts ea.
1x Dev Cluster – 7 hosts
1x DMZ Cluster – 7 hosts

AAEP/AEP/Domain Design Exercise – 75 Rack Servers

20 Misc Bare Metal
General AAEP
General Phys Domain

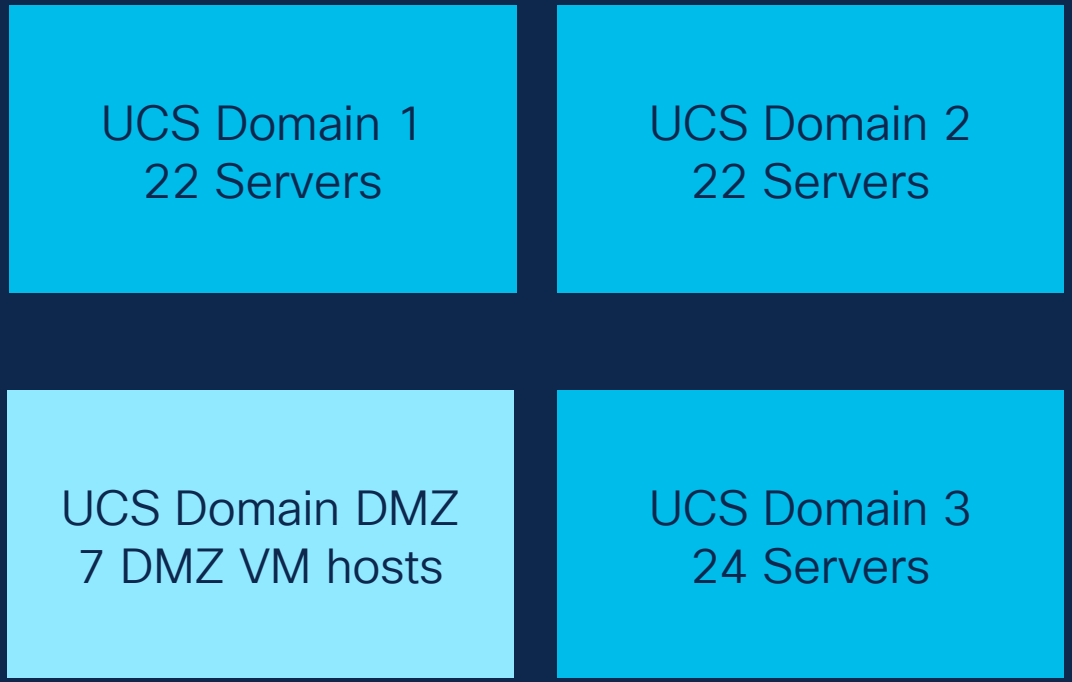
10 Big Data*
Big Data AAEP
Big Data Phys Domain

10 Backup*
Backup AAEP
Backup Phys Domain

35 VM Hosts*
3x Prod Clusters – 7 hosts ea.
Prod VM AAEP
Prod Vmm Domain
1x Dev Cluster – 7 hosts
Dev AAEP
Dev Vmm Domain
1x DMZ Cluster – 7 hosts
DMZ AAEP
Dmz Vmm Domain

*Mgmt connectivity not addressed here, depending on design it would be contained in 1 or more AAEPs

AAEP/AEP/Domain Design Exercise – UCS – 75 Servers



AAEP/AEP/Domain Design Exercise – UCS – 75 Servers

UCS Domain 1
22 Servers
Prod AAEP
Static and VMM Domain

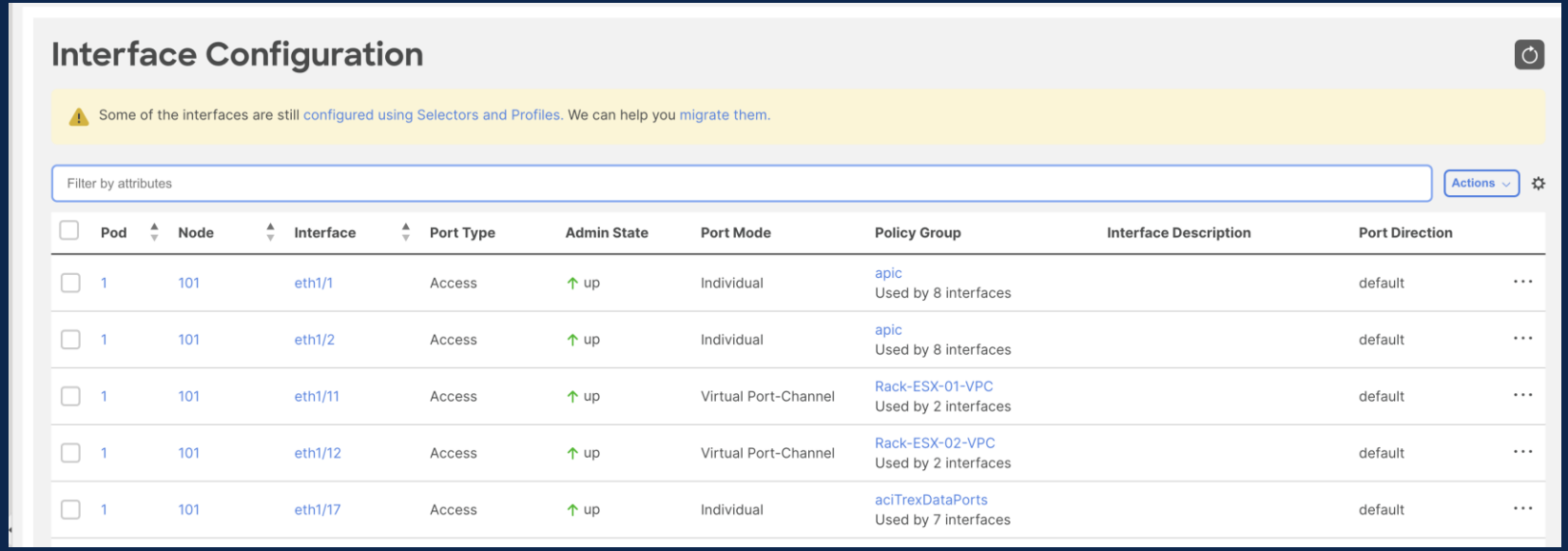
UCS Domain 2
22 Servers
Prod AAEP
Static and VMM Domain

UCS Domain DMZ
7 DMZ VM hosts
DMZ AAEP
DMZ Static and VMM Domain

UCS Domain 3
24 Servers
Prod AAEP
Static and VMM Domain

Interface Configuration

ACI 5.2(7) and ACI 6.0 have a simplified interface configuration model that streamlines much of the following topics.



Interface Policies and Groups

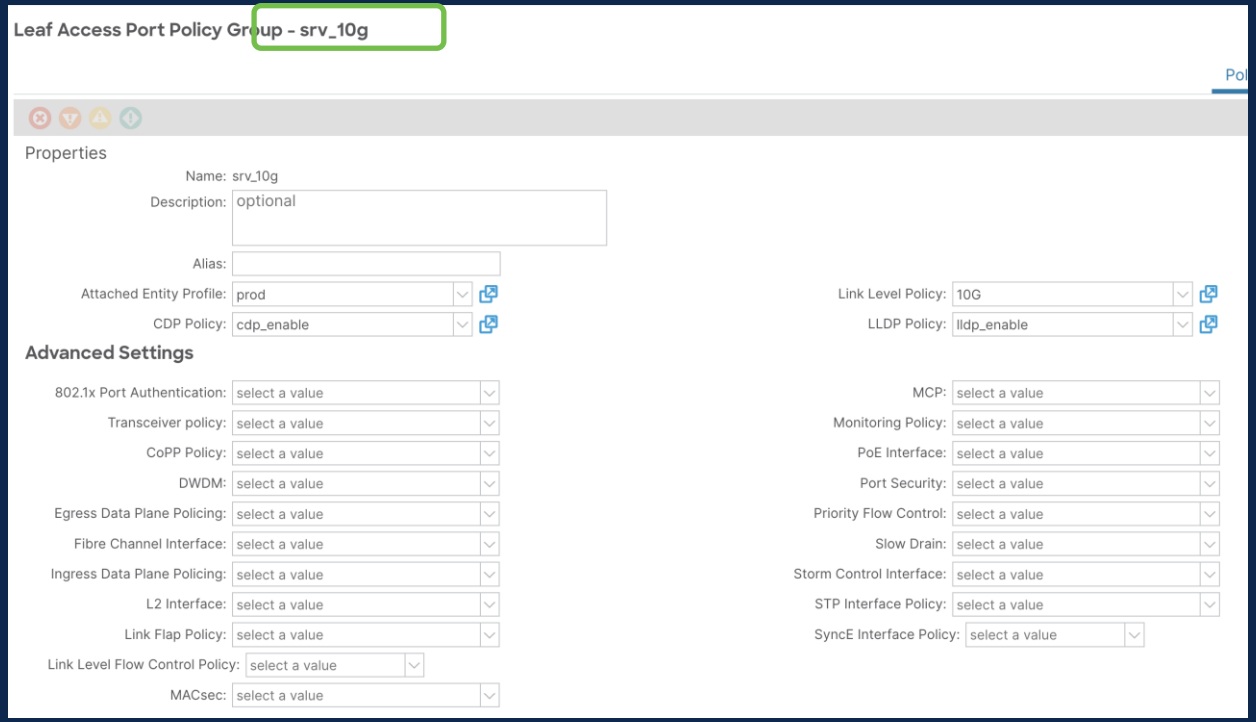
The primary purpose of the Interface Policy Group (IPG) is to define properties to the port. These properties are Interface Policies (link speed, LLDP, CDP, Storm Control, etc).

- Interface Policies can be re-used over an over and usually are only defined once. Many are generated automatically in newer versions. ie: cdpEnabled, lldpDisabled
- IPGs can be created based on the cluster of devices it will be supporting. This methodology encourages optimal re-use. ie: ndData, ndMgmt, esxCl01Dvs
- Port-Channels and VPCs are an exception to the re-use

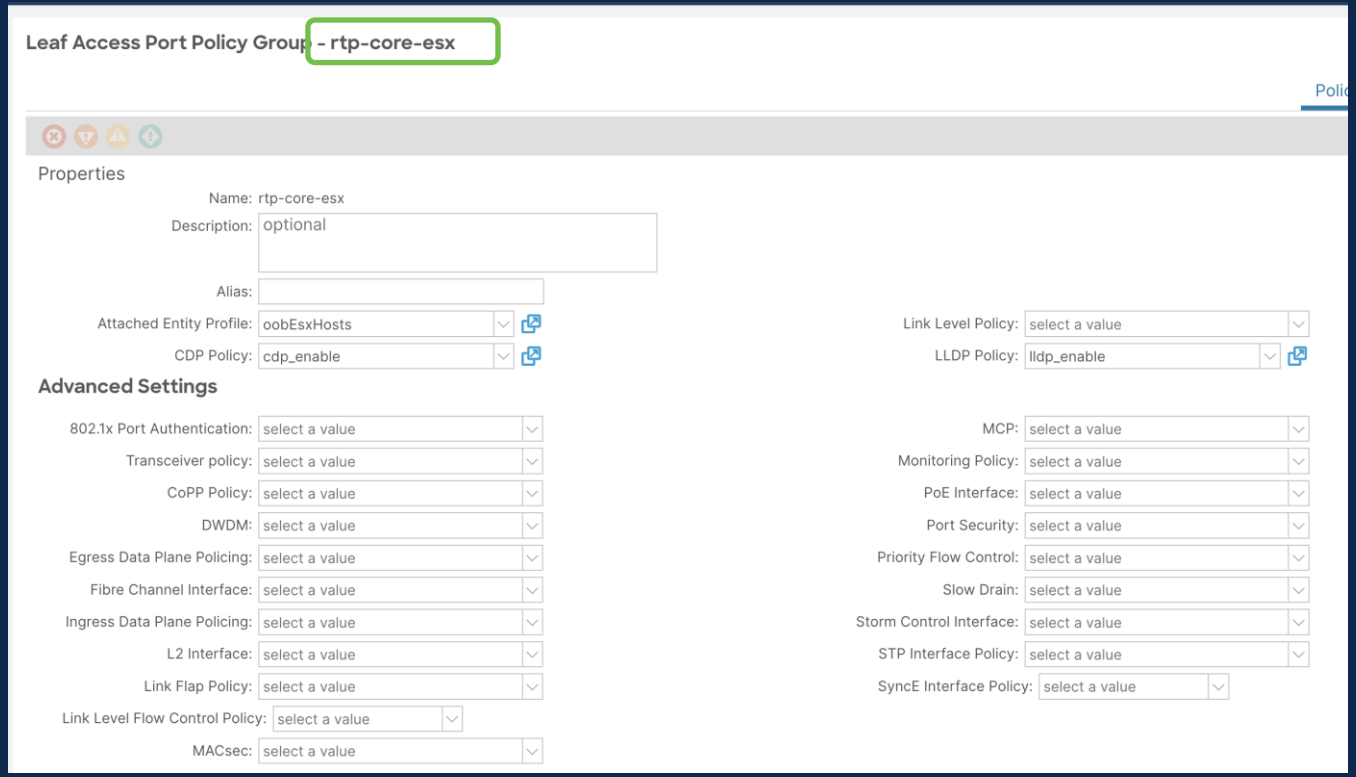
Interface Policies and Groups

- Since the IPG references the AAEP, this reinforces the need for it to have the proper port scope, especially if EPGs are applied via the AAEP.
- For critical connections consider dedicated, unshared IPG for the respective ports of each server.

IPG - Generic



IPG – Self Documenting

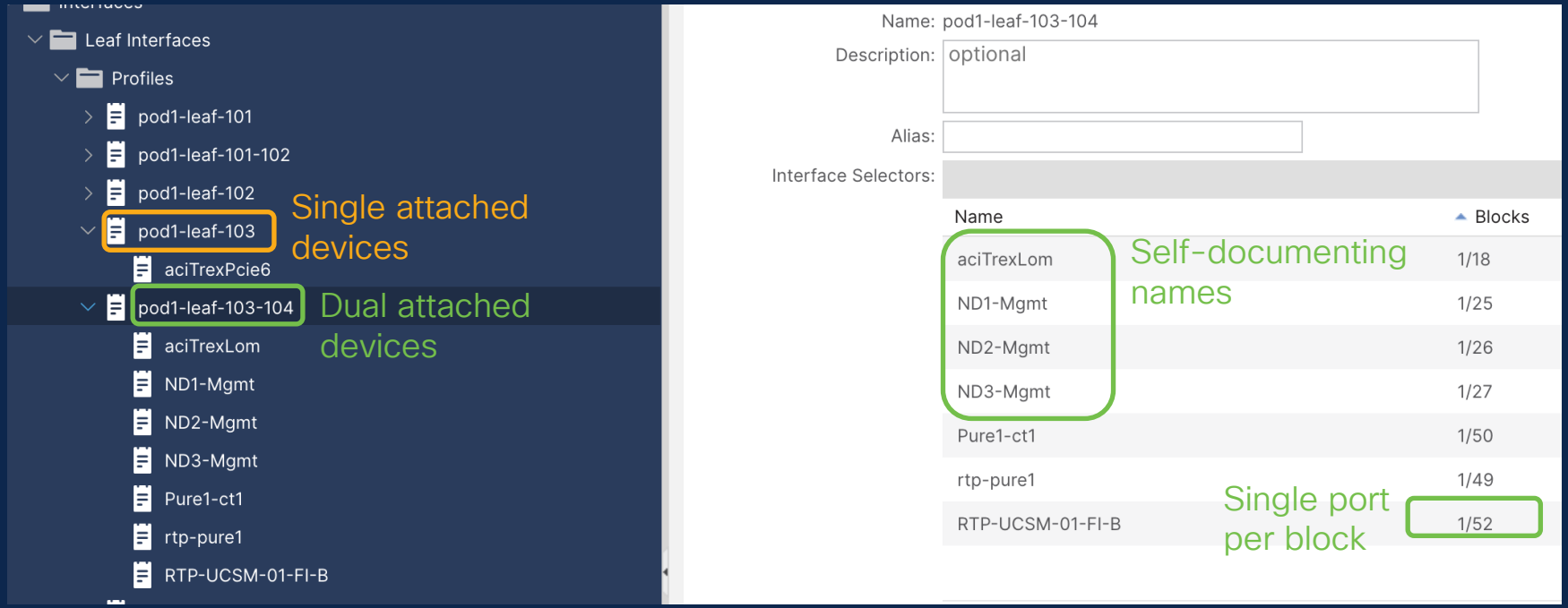


Leaf Interface Profiles and Interface Selectors

A Leaf Interface Profile is used to bind interface configuration to the physical switch via the Switch Profile. Under the Leaf Interface Profile interfaces are referenced via Interface selectors to define the port-by-port configuration with IPGs

- A single Leaf Interface Profile per Switch Profile and possibly one Leaf Interface Profile per VPC pair is all that is needed.
- For Interface Selectors only reference one port per selector and use a self-documenting name for the selector.

Leaf Interface Profiles and Interface Selectors

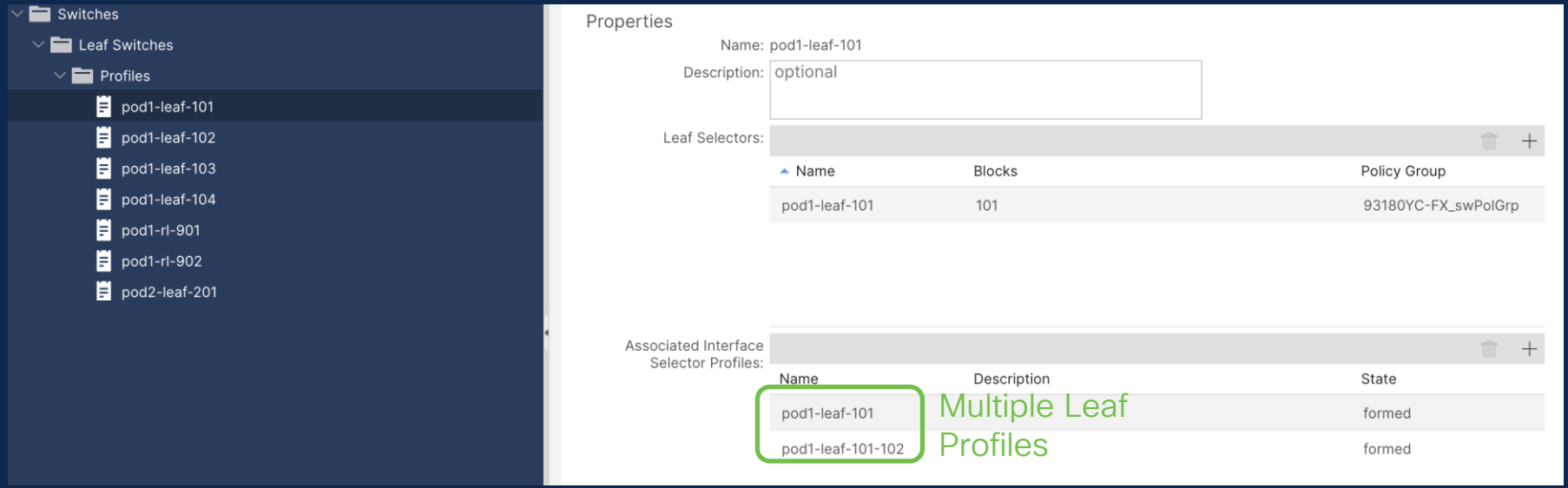


Switch Profiles

A Switch Profile binds configuration to the physical switch. This includes interface configuration and switch specific configuration like forwarding scale via Switch policy groups.

- A single switch profile per switch is all that is needed, no need for pairs for VPC. Multiple Leaf Profiles can be mapped under a single Switch Profile

Switch Profiles

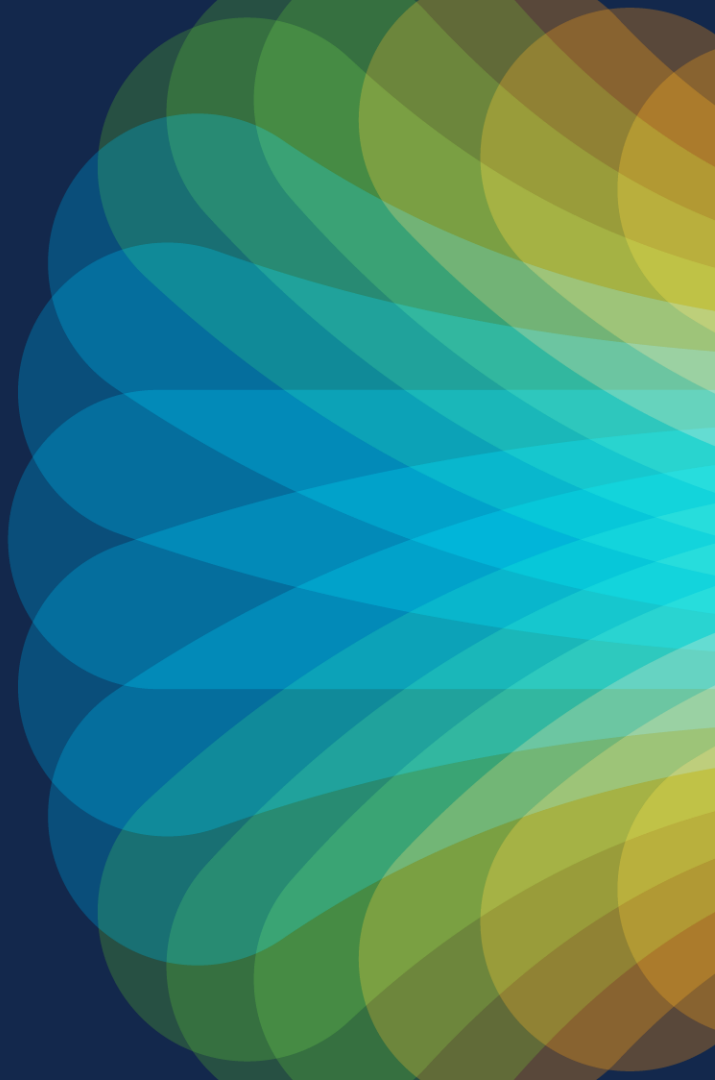


Avoid if you can!

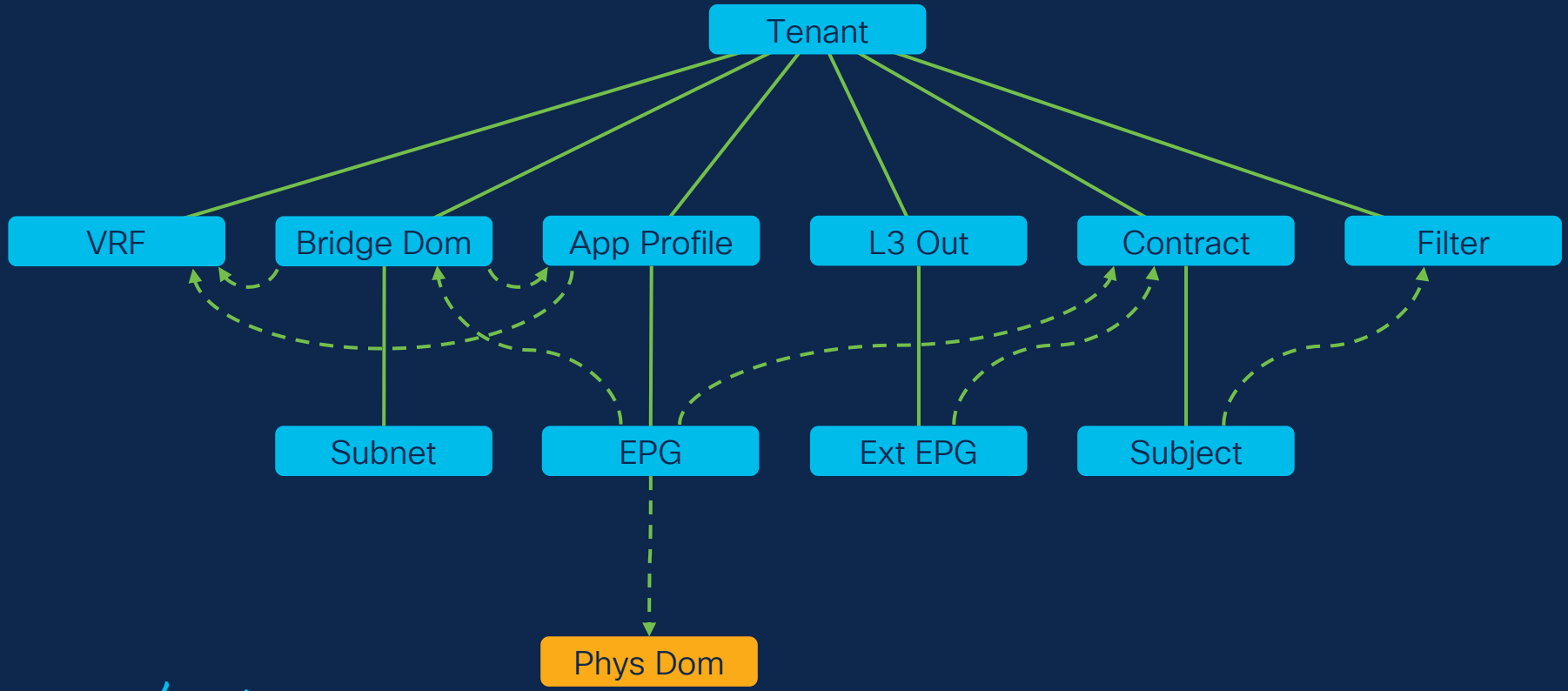
Interface Overrides

While possible, it is difficult to follow, makes troubleshooting much more complicated, creates exceptions. one must understand

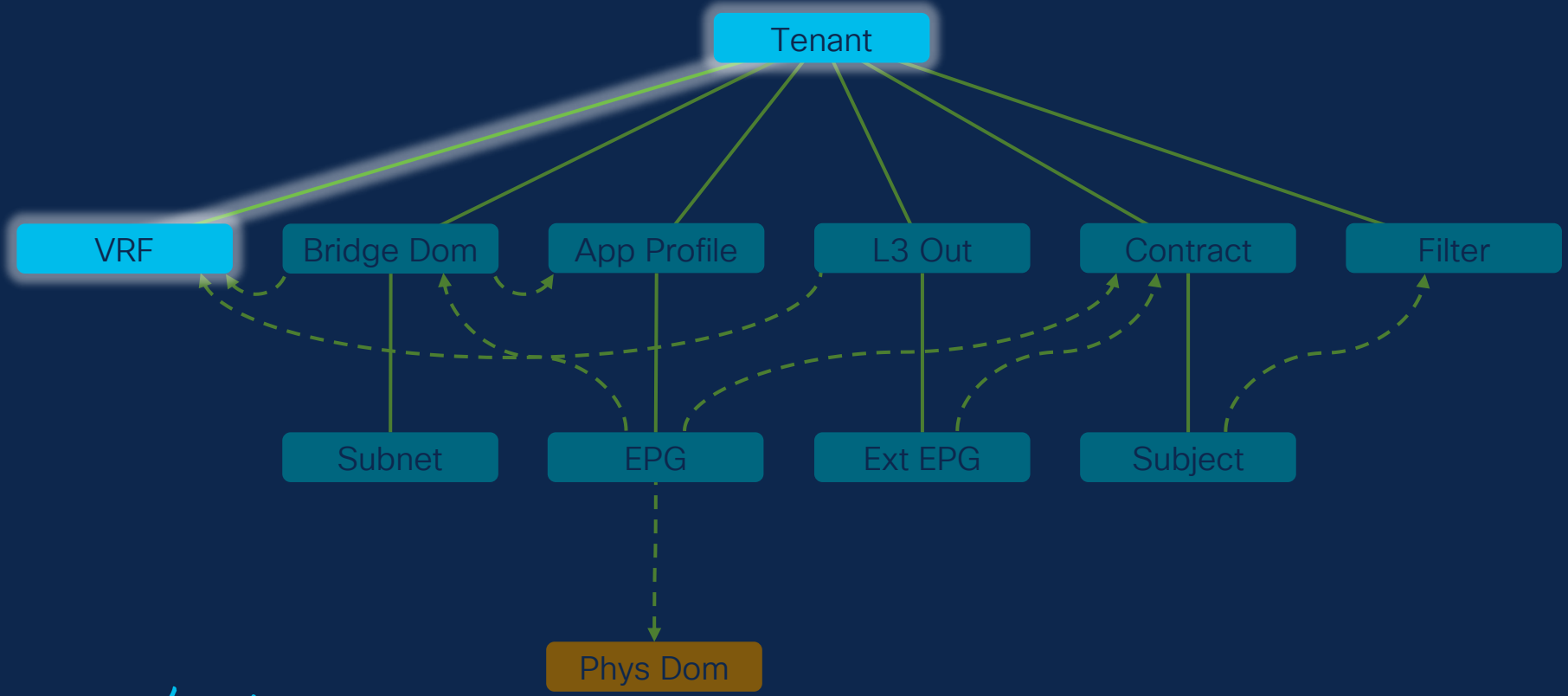
Tenant Objects



Tenant Object Model



Tenant Object Model



For your Reference

The Tenant Object

A tenant is a logical container for application policies that enable an administrator to exercise domain-based access control. A tenant represents a unit of isolation from a policy perspective. Tenants can represent a customer in a service provider setting, an organization or domain in an enterprise setting, or just a convenient grouping of policies.

Tenant	
Class:	fvTenant
DN Prefix:	tn-
Parent Object:	uni (Universe)

Example DN: uni/tn-BRKDCN-2647

For your Reference

The VRF Object

A Virtual Routing and Forwarding (VRF) or "context" defines a Layer 3 address domain. One or more bridge domains are associated with a VRF. All the endpoints within the Layer 3 domain must have unique IP addresses because it is possible to forward packets directly between these devices if the policy allows it. A tenant can contain multiple VRFs.

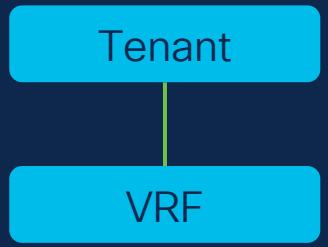
VRF	
Class:	fvCtx
DN Prefix:	ctx-
Parent Object:	Tenant

Example DN: uni/tn-BRKDCN-2647/ctx-Live

Tenant Object Model – Tenant and VRF

Preferred Design

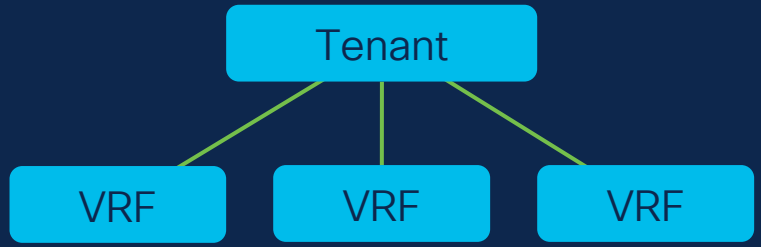
Single VRF per Tenant



- Object relationships easier to track
- Simplified L3Out
- No route-leaking

Not Recommended

Multiple VRFs per Tenant



- Confuses object relationships
- Complicates L3Outs
- May encourage route–leaking

Avoid if you can!

Route-leaking

While technically possible, it is difficult to follow (even in traditional networks), makes troubleshooting much more complicated, and requires complex contract rules.

Exception:

The common tenant is made for route-leaking and facilitates consumption of common services across a multi-tenant fabric.

Default Tenants

infra	Infrastructure tenant: Networking for Inter-Pod Network (IPN), Inter-Site Network (ISN), and Remote Leaf. Recommendation: Don't use for other networks
mgmt	Management tenant: Out-of-band and in-band management for ACI controllers, leaves, and spines. Recommendation: Don't use for other networks
common	Common tenant: Common services to be consumed across multiple tenants. Recommendation: Use for shared services

Default Tenants – Common

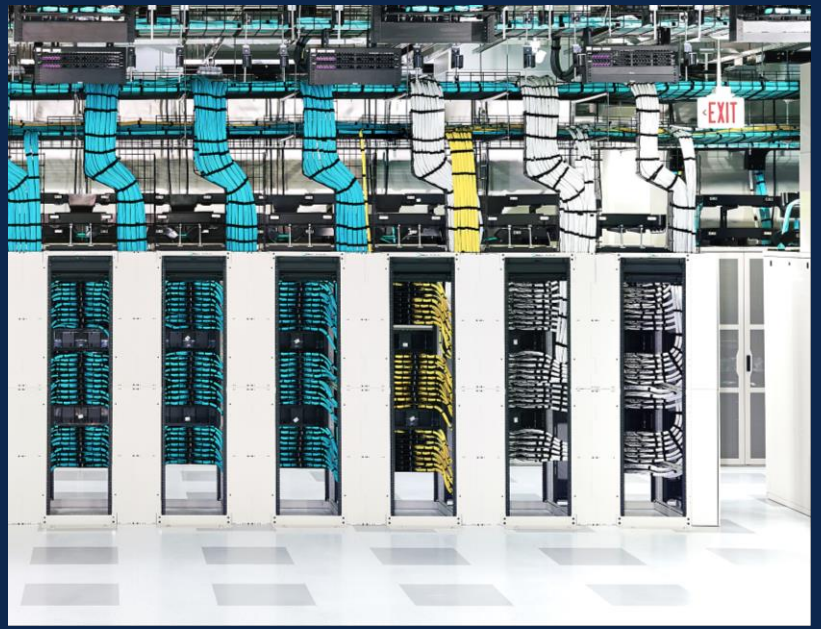
Common Tenant – Dedicated tenant for common services and shared objects.

- For shared services hosted in Common and consumed in other tenants.
- Route-leaking is simplified, but still required.
- Not recommended to split the object model between Tenants. (e.g. using a VRF or an L3Out out of Common from another tenant.)
  - Exception: It is a normal practice to use Filters from the Common tenant across the fabric

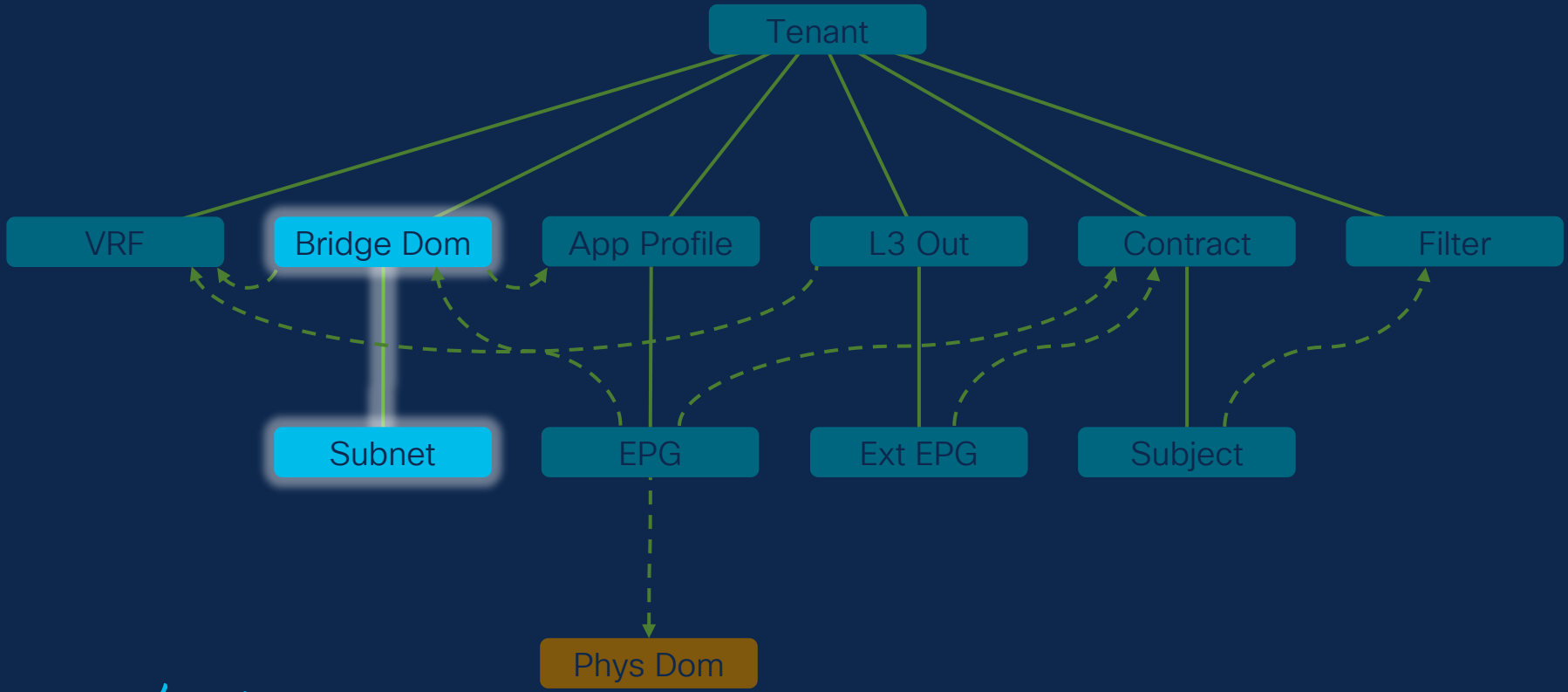
Example: Active Directory services hosted in Common and consumed in “Prod” and “Dev” tenants.

Tenant / VRF Object Scaling Decision

- Primarily used to isolate logical network domains (business units, customers, etc.).
- One VRF per Tenant.
- Consider default-deny policy model before adding a Tenant.
- Avoid splitting the network model across multiple tenants.



Tenant Object Model



For your Reference

The Bridge Domain Object

A bridge domain defines the unique Layer 2 MAC address space and a Layer 2 flood domain. With connections to Subnets and L3Outs, the BD provides a critical connection point between Endpoints and all things at Layer 3.

Uses:

- Define L2 flooding domain
- Connect to ACI L3 Object (Subnet, L3Out, etc.)

Bridge Domain	
Class:	fvBD
DN Prefix:	BD-
Parent Object:	Tenant

Example DN: uni/tn-BRKDCN-2647/BD-demo

For your Reference

The Subnet Object

The subnet object brings the layer 3 address space to a bridge domain where it can ultimately be consumed by endpoints in an endpoint group (EPG). The address defined in a subnet will be programmed into leaves as an anycast gateway for the associated endpoints.

Uses:

- Subnet and Anycast Gateway for endpoints

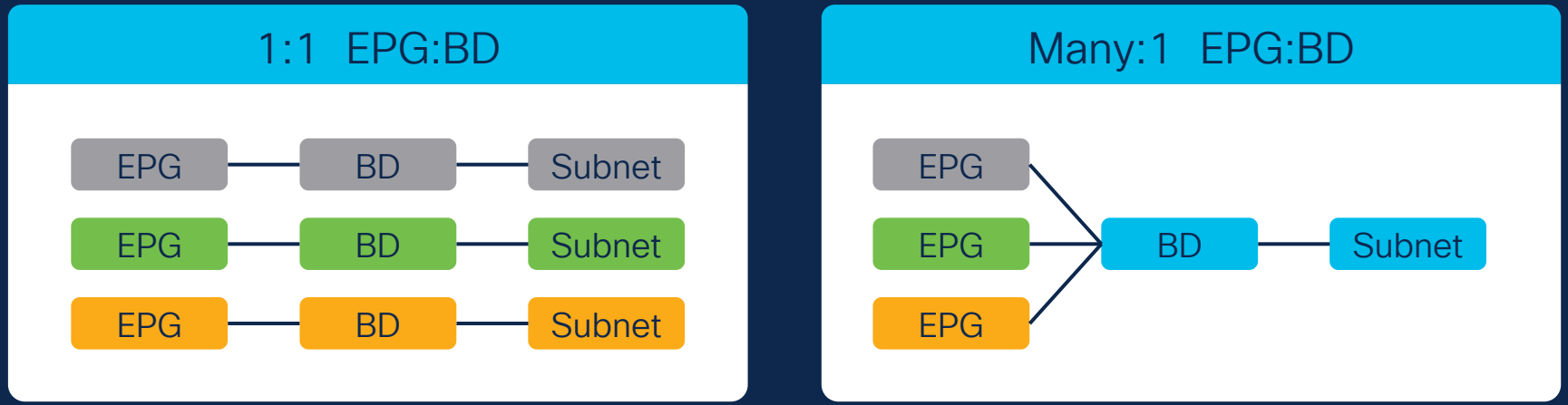
Subnet	
Class:	fvSubnet
DN Prefix:	subnet-
Parent Object:	Bridge Domain

Example DN: uni/tn-BRKDCN-2647/BD-demo/subnet-[10.20.30.1/24]

Bridge Domain – Subnet Recommendation

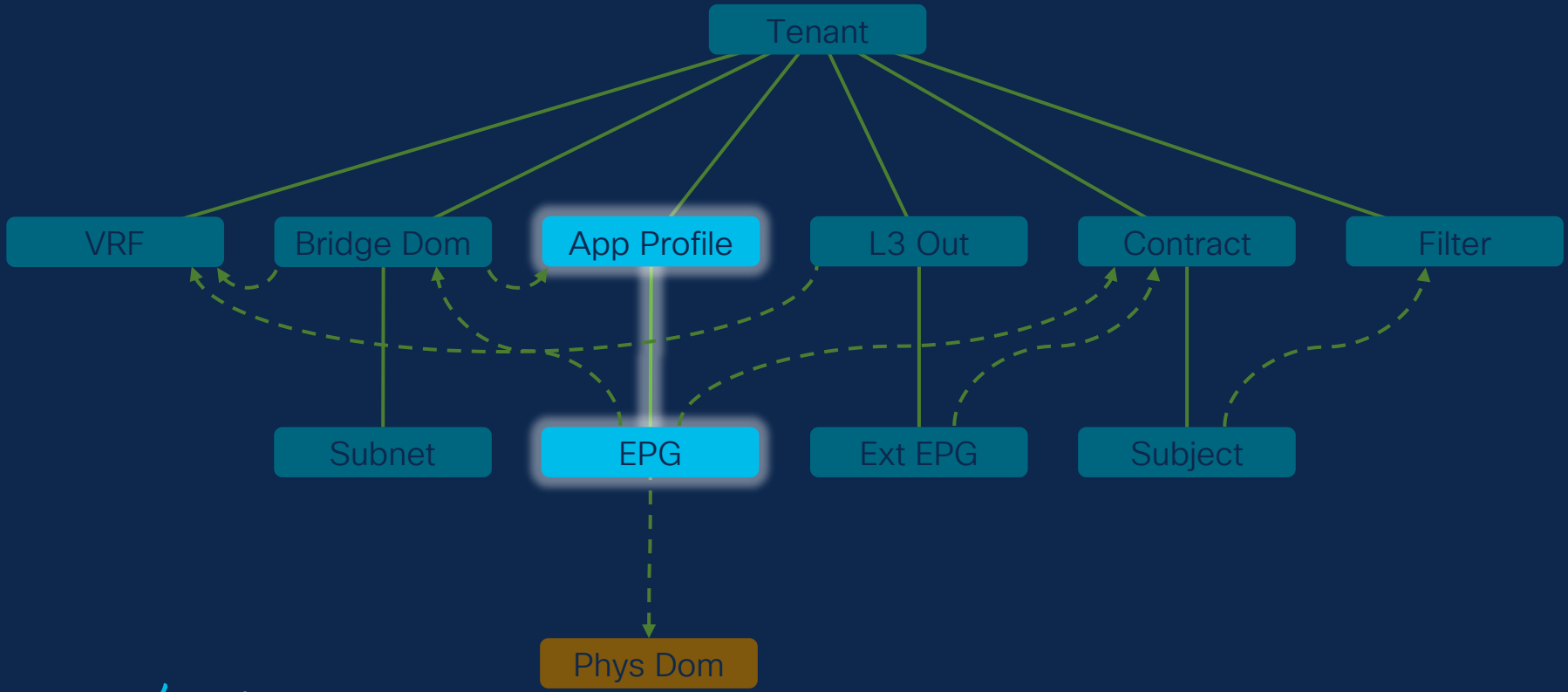
One Subnet per Bridge Domain

Bridge Domain “Overload”



Not necessary to use the subnet as a segmentation boundary

Tenant Object Model



For your Reference

The Application Profile Object

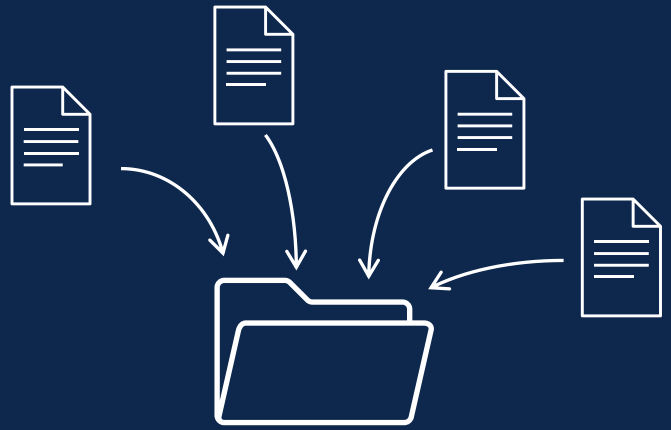
The application profile contains as many (or as few) EPGs as necessary that are logically related to providing the capabilities of an application.

Uses:
- Organization of EPGs
- Contract scoping boundary (not recommended)

Application Profile	
Class:	fvAp
DN Prefix:	ap-
Parent Object:	Tenant

Example DN: uni/tn-BRKDCN-2647/ap-app1

Application Profile General



Generally, just an EPG container

For your Reference

The Endpoint Group Object

An EPG is a managed object that contains a collection of endpoints. Endpoints are devices that are connected to the network directly or indirectly. EPGs are fully decoupled from the physical and logical topology.

Uses:
- Segmentation
- Association to BD/Subnet
- Define endpoint connectivity

Endpoint Group	
Class:	fvAEpg
DN Prefix:	epg-
Parent Object:	App Profile

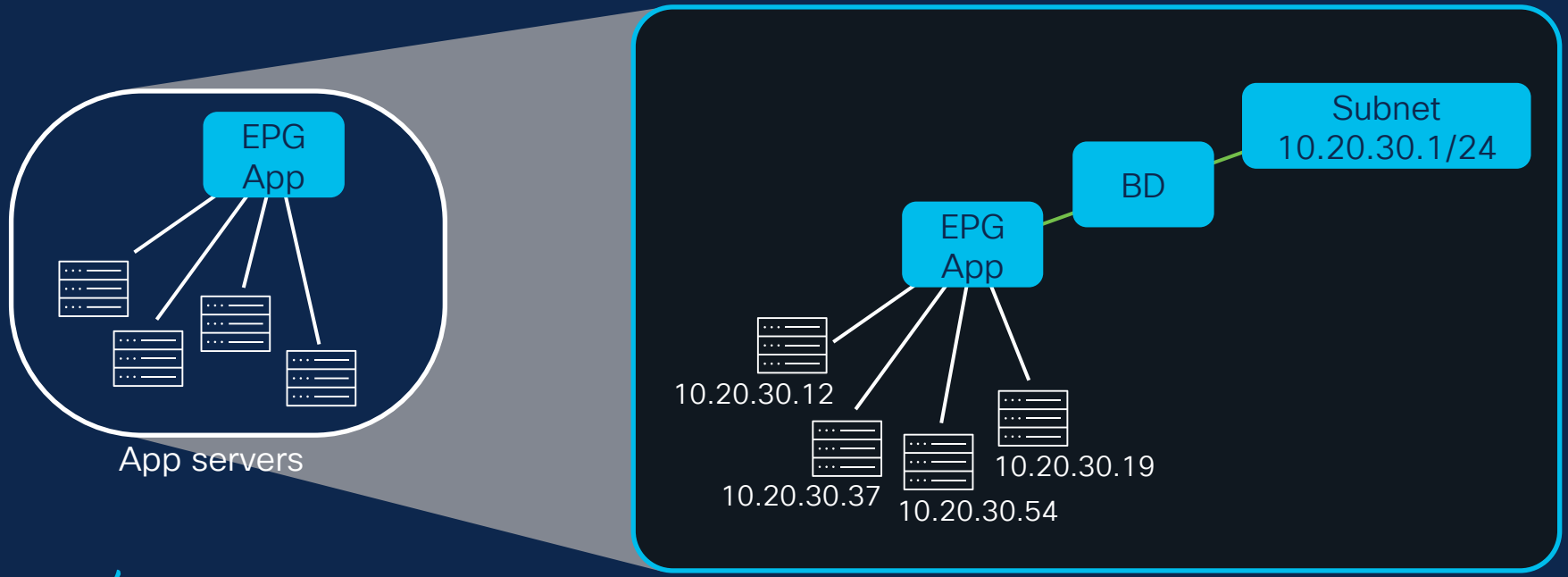
Example DN: uni/tn-BRKDCN-2647/ap-app1/epg-web

Endpoint Group

Primary Role: Group Endpoints with like policy requirements

Endpoint Group

Secondary Role: Connect Endpoints to a BD and Subnet.

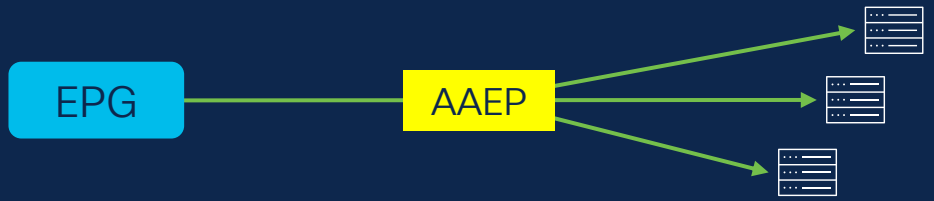


Endpoint Group

Tertiary Role: Connect endpoints to the fabric.

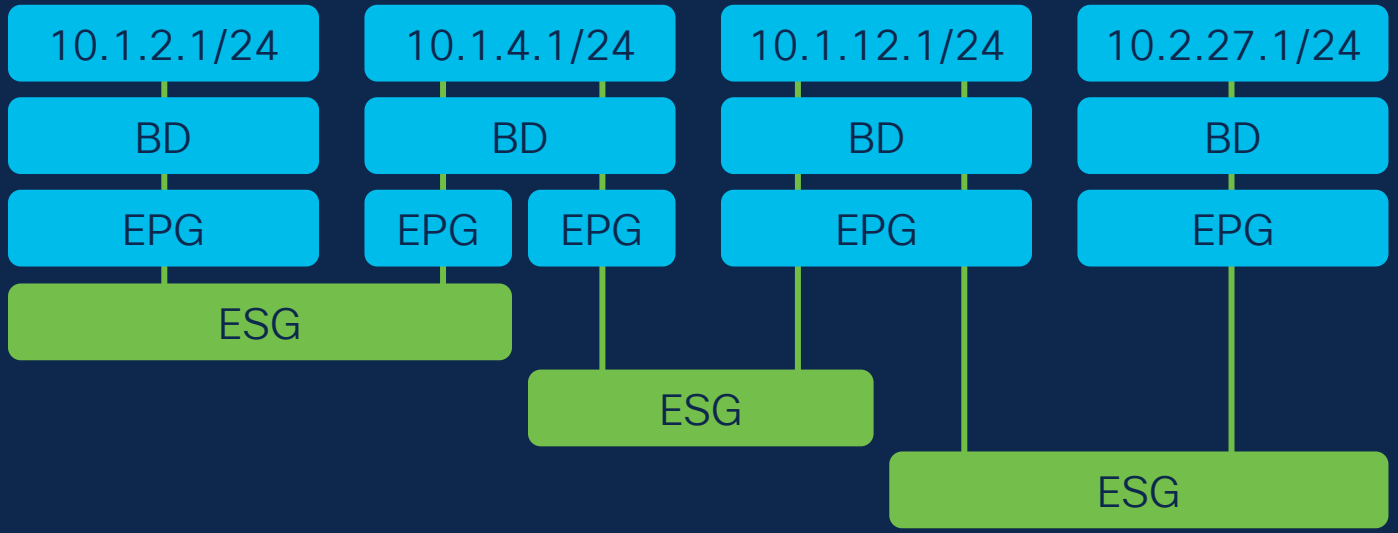
1. Static Port Binding

2. AAEP to EPG mapping



3. VMM Integration

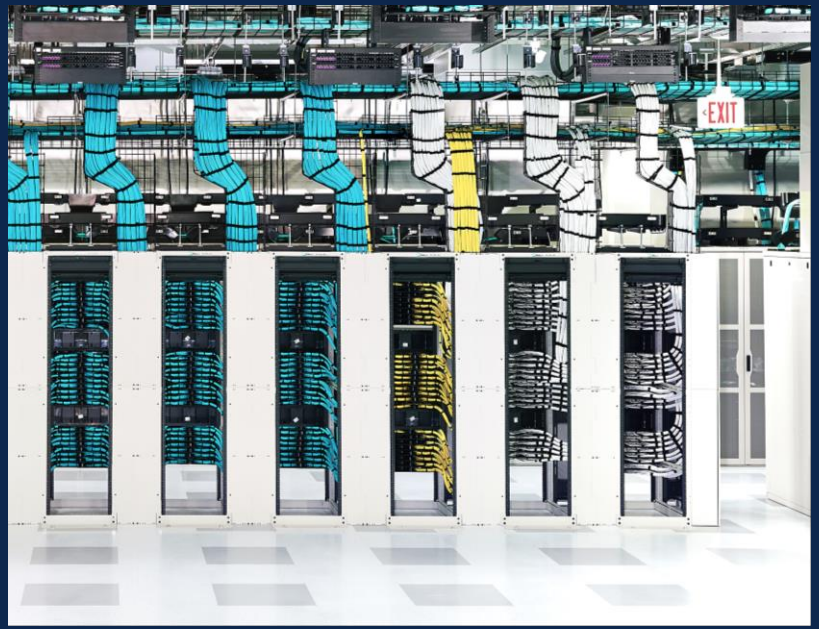
ESG – Endpoint Security Group



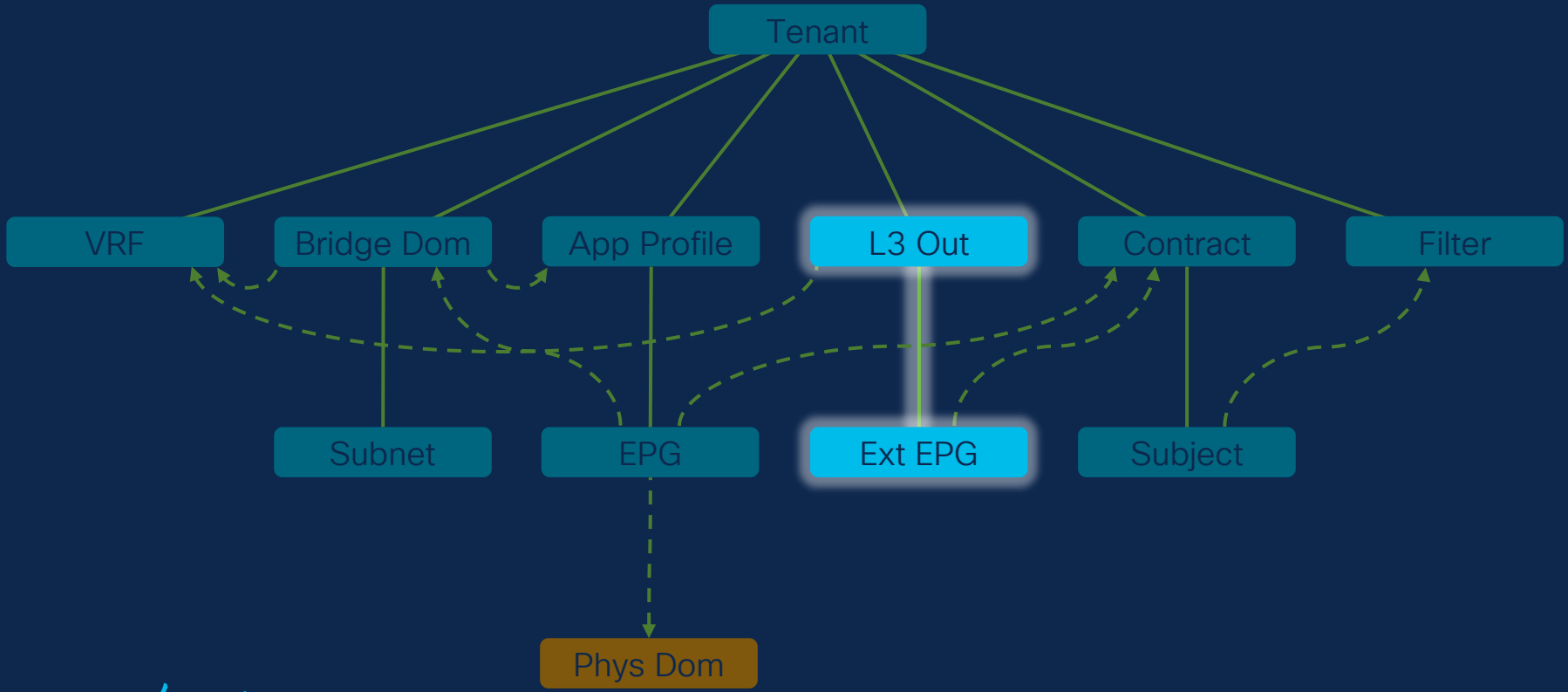
For more info on ESGs review these sessions:
BRKDCN-2984 “ACI: the foundation of an internal private cloud”
BRKDCN-2658 “Application Centric Design: How to get there with Cisco”

EPG Object Scaling Decision

- Primarily decision should focus on segmentation.
- Consider breaking up a BD/Subnet into multiple EPGs as needed.
- Primary limiting factor for EPGs is VLAN limit per leaf.
  (Each EPG uses one VLAN.)



Tenant Object Model



For your Reference

The L3Out Object

A Layer 3 connection between border leaves in the fabric and a set out external routing devices (router, firewall, L3 Switch, etc.). This connection provides a routed path out of the fabric; designed to reach a specific set of networks (defined by External EPG).

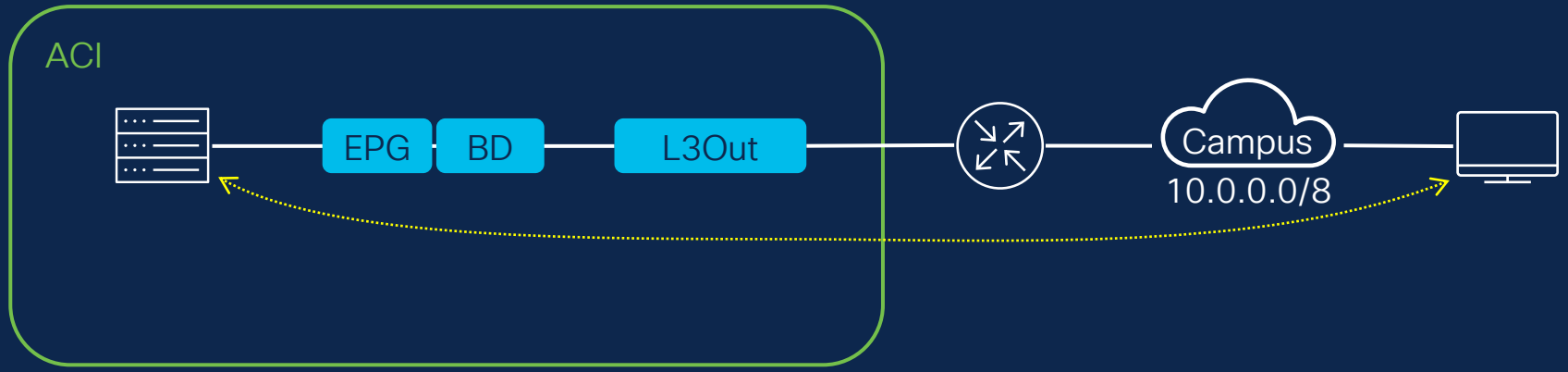
L3Out	
Class:	l3extOut
DN Prefix:	out-
Parent Object:	Tenant

Uses:

- Define a routed path out of a Tenant
- A configuration point for routing protocols or static routes.

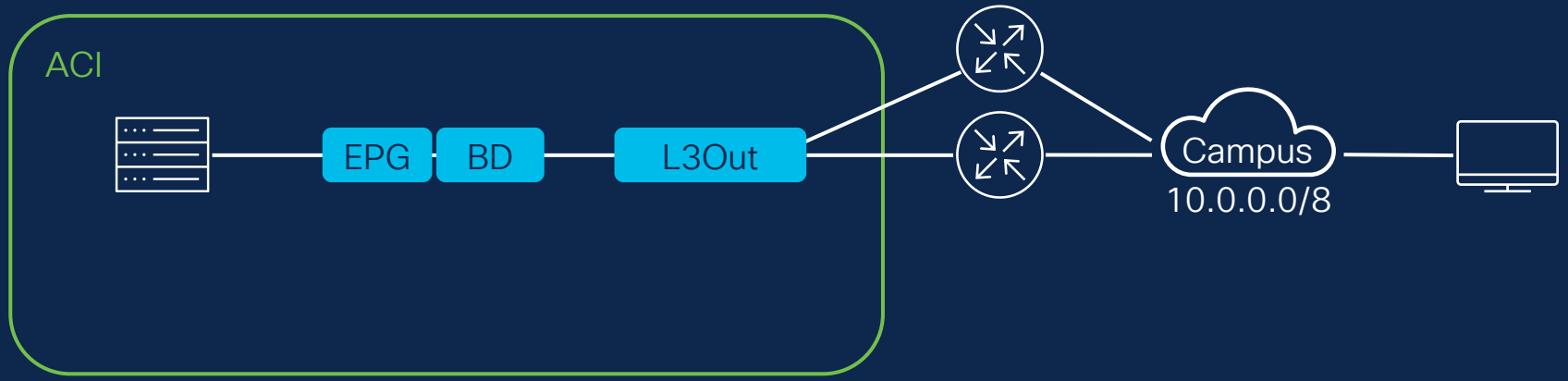
Example DN: uni/tn-BRKDCN-2647/out-core

L3Out Object



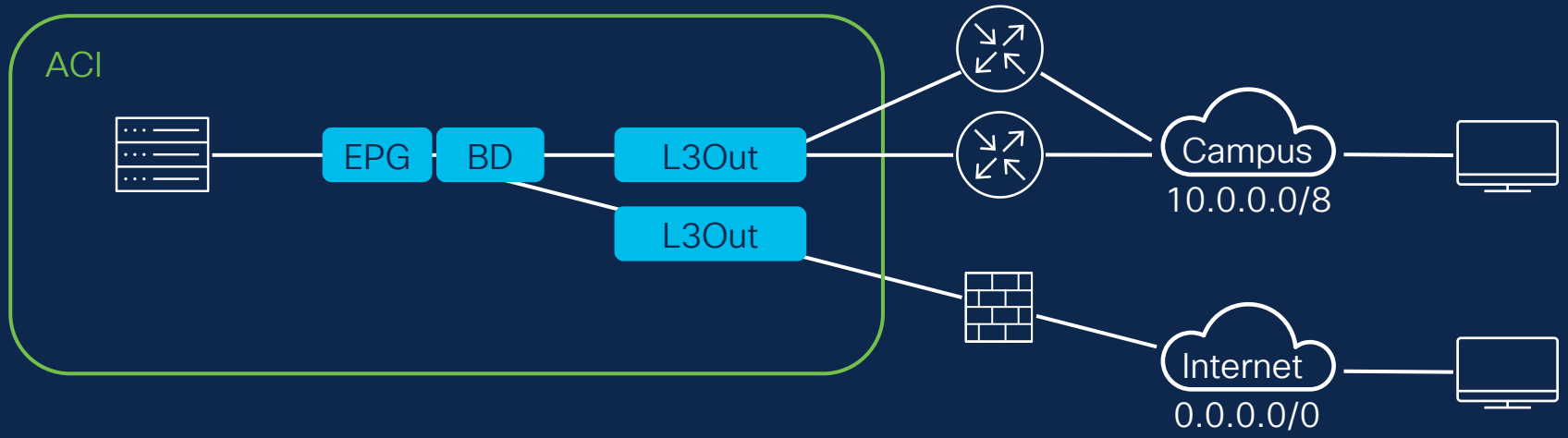
The primary purpose of the L3Out is to provide a routed path between EPGs and a specific set of external networks.

L3Out Object



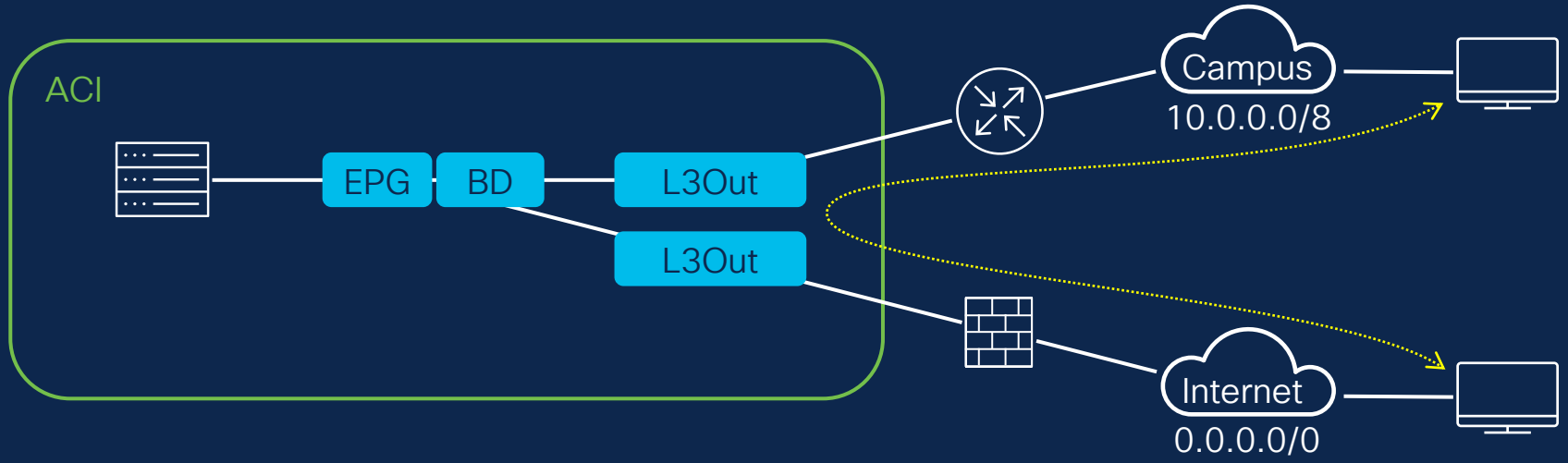
The primary purpose of the L3Out is to provide a routed path between EPGs and a specific set of external networks.

L3Out Object



The primary purpose of the L3Out is to provide a routed path between EPGs and a specific set of external networks.

Transit Routing



Disabled by default, Transit Routing allows traffic to pass through ACI between external endpoints connected through different L3Outs.

Avoid if you can!

Transit Routing

While possible, ACI is not generally intended as a step in the path between different external networks. This makes transit routing a non-standard configuration.

Exception:

Some resources in the DC, like load balancers, may be L3 connected to the fabric.

For your Reference

The External EPG

The External EPG defines networks external to the L3Out's context where EPG communications may be needed. For most use-cases the Ext EPG is not used for L3, but rather as an anchor point for security policy.

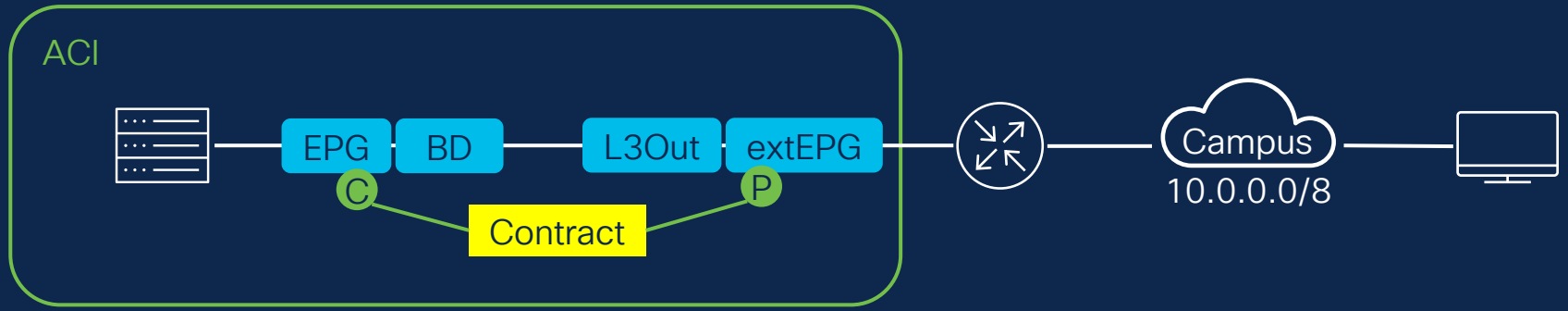
Uses:

- Define an external subnet for policy enforcement.
- Define subnets for route control.

External EPG	
Class:	l3extInstP
DN Prefix:	instP
Parent Object:	L3Out

Example DN: uni/tn-BRKDCN-2647/out-core/instP-default

Ext EPG Primary Use - Security



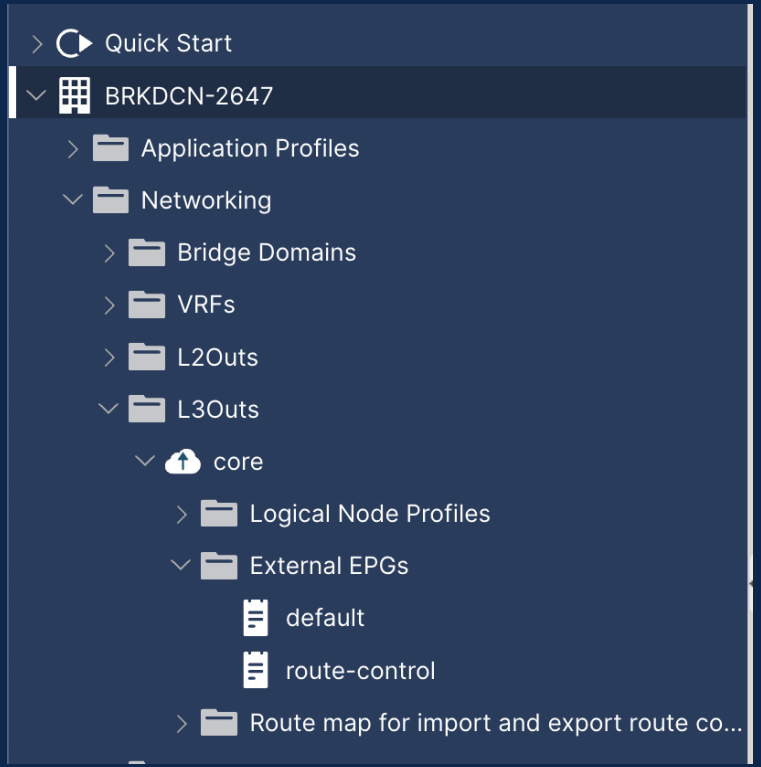
ACI Forwarding Decision

1. Is there a path? (L2/L3, Route/Switch)
2. Is this communication allowed? (Policy-Model / Zero Trust)

Ext EPG Other Use – Route Control

Route Control with External EPGs for things like route-summarization.

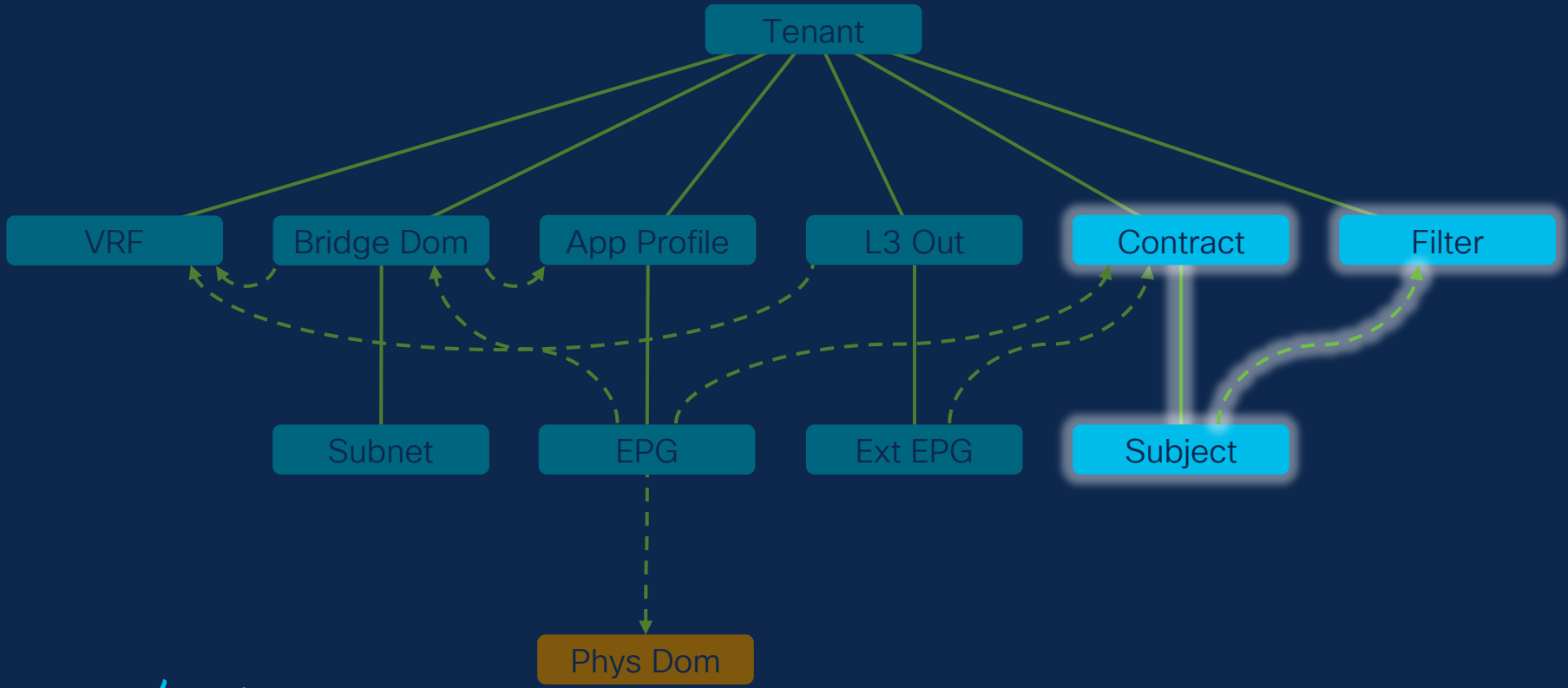
Ext EPG Other Use – Route Control



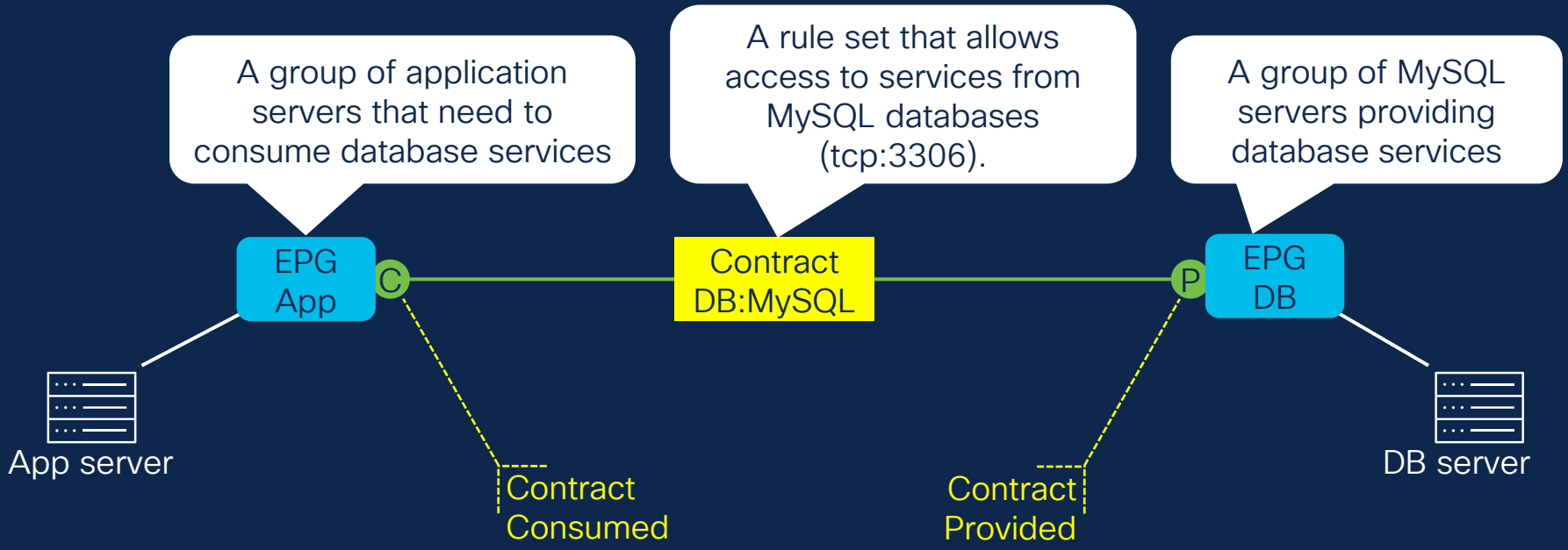
Including the route-control subnet in an existing External EPG can be confusing and misleading.

Creating a dedicated External EPG for route-control subnets will greatly improve the first-glance understanding of your ACI design.

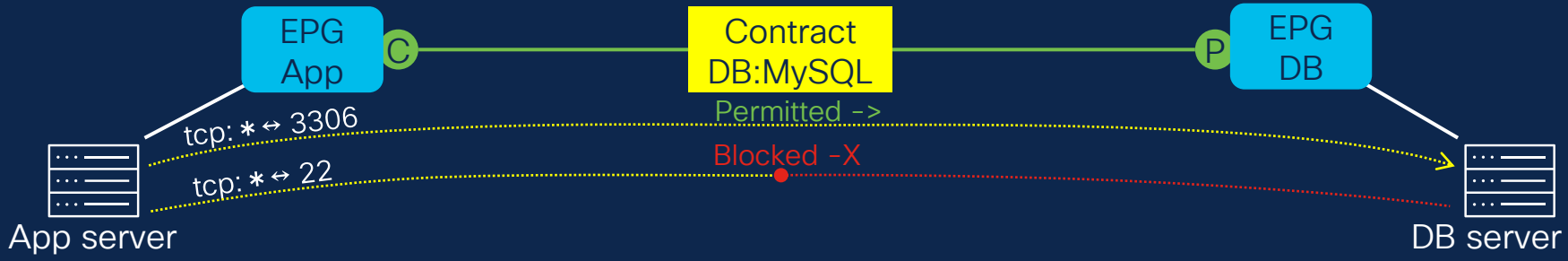
Tenant Object Model



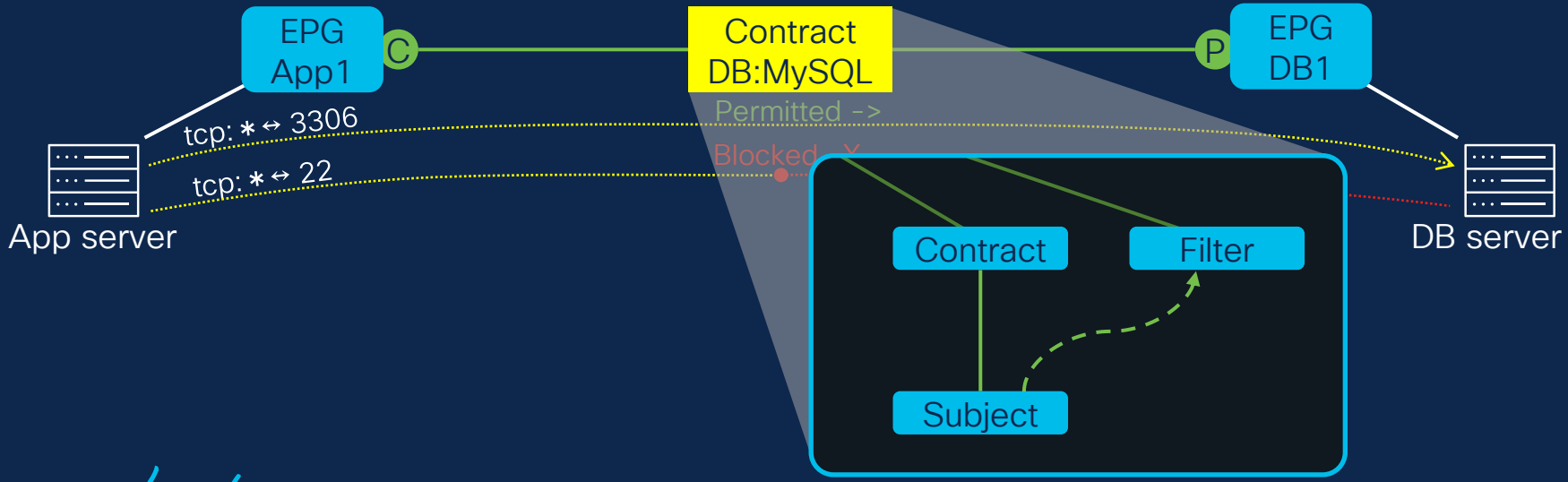
How Contracts Work



How Contracts Work



How Contracts Work



For your Reference

The Contract Object

A contract is a policy construct used to define communication between EPGs. Without a contract between EPGs, no unicast communication is possible unless the VRF is configured in unenforced mode, or those EPGs are in the preferred group.

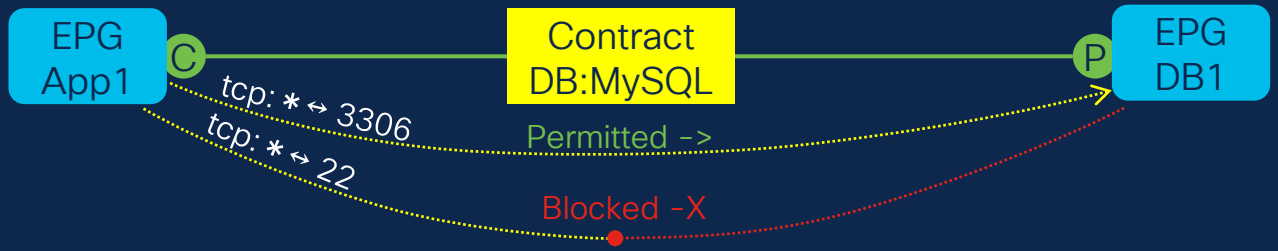
Uses:

- Define allowed communication between EPGs
- Route-leaking

Contract	
Class:	fvBrCP
DN Prefix:	brc-
Parent Object:	Tenant

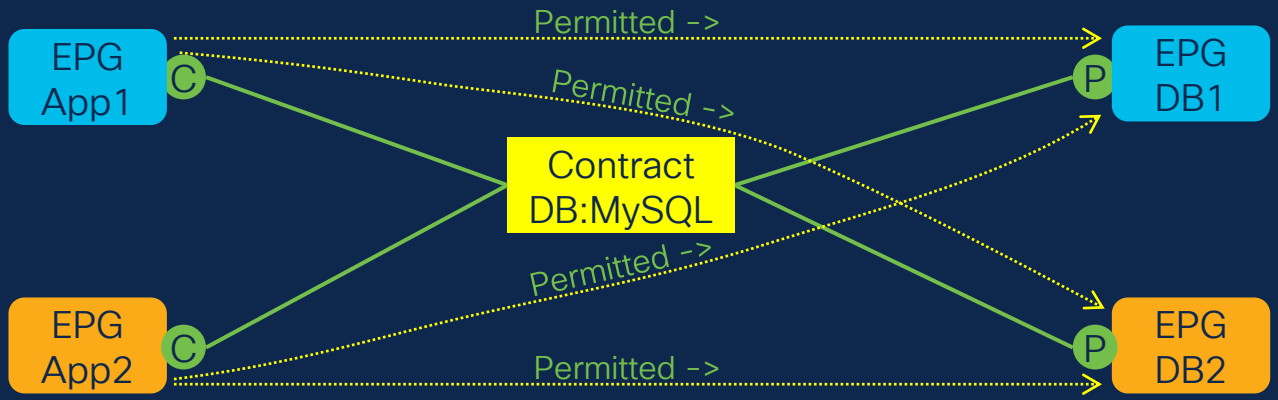
Example DN: uni/tn-BRKDCN-2647/brc-demo-db:mysql

How Contracts Work



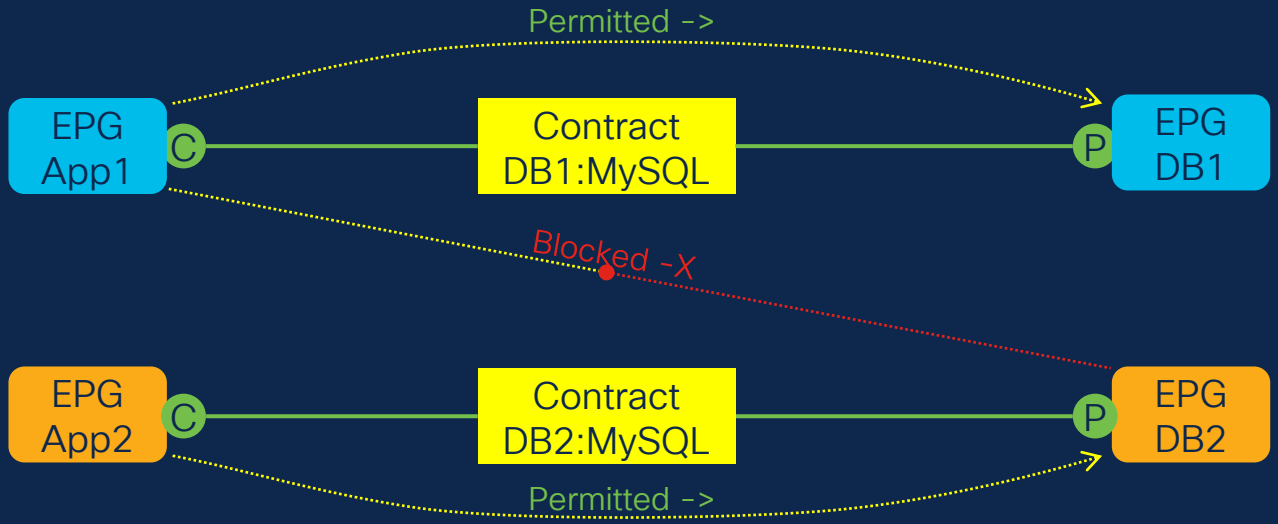
Reuse of Contracts

Reuse of Contracts

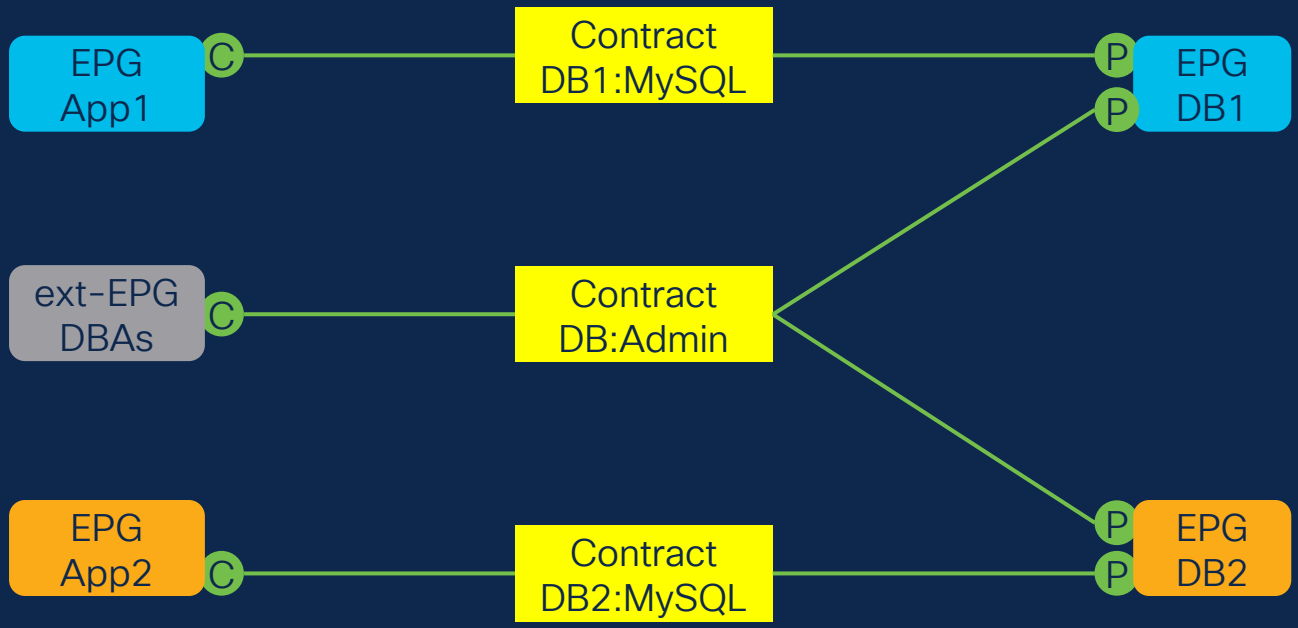


Reuse of contracts may permit traffic that was not intended!

Reuse of Contracts



Contract with multiple providers



For your Reference

The Subject Object

A Subject is a contract attachment point for filters and other contract related policies. This attachment point allows for different policies to be associated with specific filter sets within a single contract.

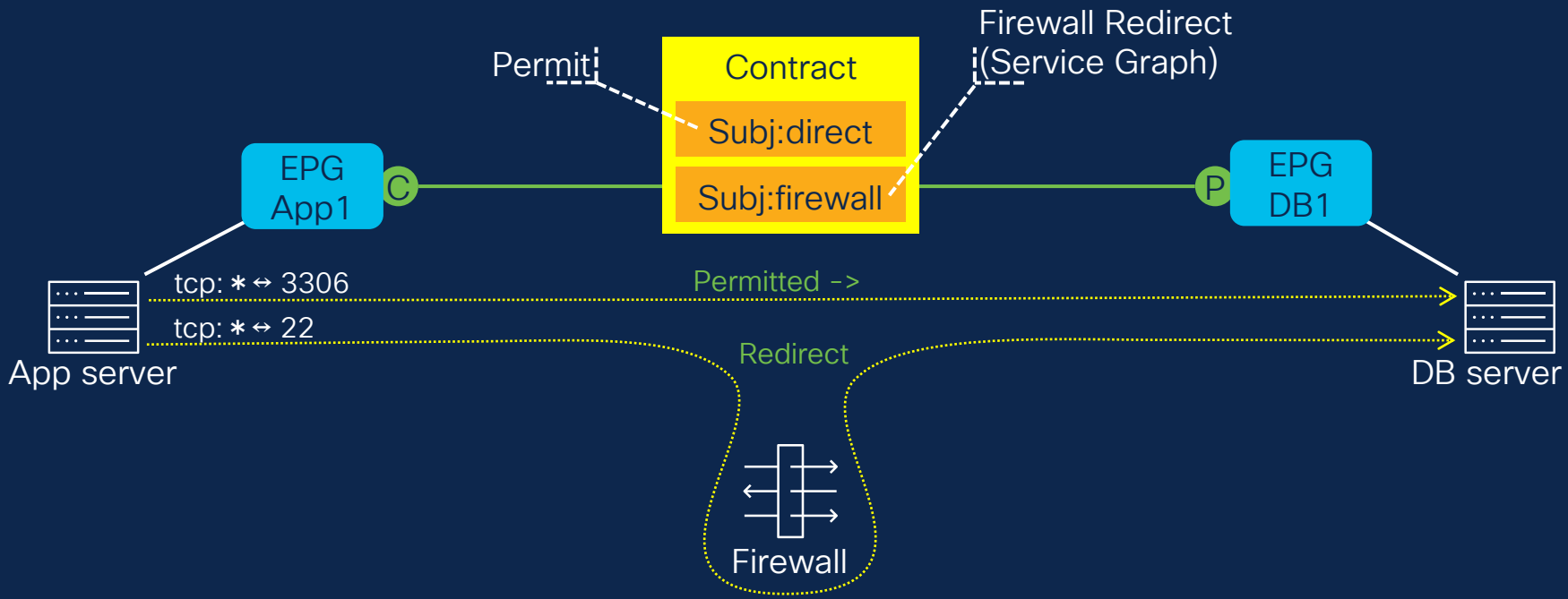
Uses:

- Control how a filter is applied to a contract
- Associate contract policy with filter sets
  - e.g. L4-L7 Service Graph or QoS priority

Subject	
Class:	vzSubj
DN Prefix:	subj-
Parent Object:	Contract

Example DN: uni/tn-BRKDCN-2647/brc-demo-db:mysql/subj-default

Subject



For your Reference

The Filter Object

A Filter is a list of matching rules that define communications to associate with a contract. These rules include things like ethertype, protocol, and source and destination port(s).

Uses:

- Classify traffic for policy enforcement.

Filter	
Class:	vzFilter
DN Prefix:	flt-
Parent Object:	Tenant

Example DN: uni/tn-common/flt-mysql

Filters

Filter: MySQL
tcp: ∗ ↔ 3306

Filter: FTP
tcp: ∗ ↔ 20
tcp: ∗ ↔ 21

Filter: DNS
udp: ∗ ↔ 53
tcp: ∗ ↔ 53

Filter: HTTP
tcp: ∗ ↔ 80

Filter: HTTPalt
tcp: ∗ ↔ 8080

Filter: PKI
tcp: ∗ ↔ 80
tcp: ∗ ↔ 389
tcp: ∗ ↔ 636
tcp: ∗ ↔ 9389

Filter: HTTPS
tcp: ∗ ↔ 443

Filter: HTTPSalt
tcp: ∗ ↔ 8443

Filters

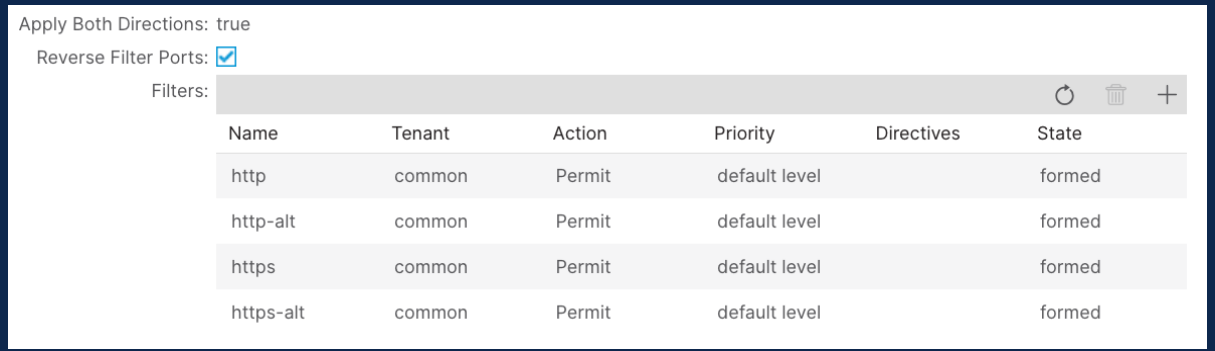
Filter: web-all
tcp: * ↔ 80
tcp: * ↔ 443
tcp: * ↔ 8080
tcp: * ↔ 8443

Filter: http
tcp: * ↔ 80

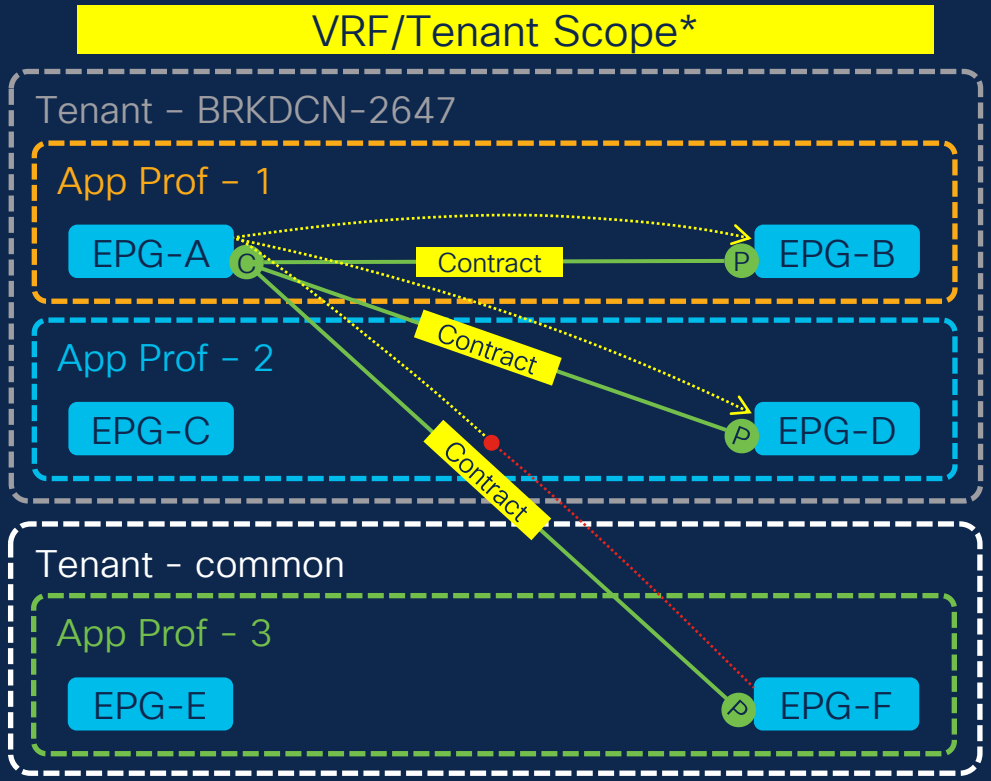
Filter: https
tcp: * ↔ 443

Filter: https-alt
tcp: * ↔ 8443

Filter: http-alt
tcp: * ↔ 8080



Contract Scope

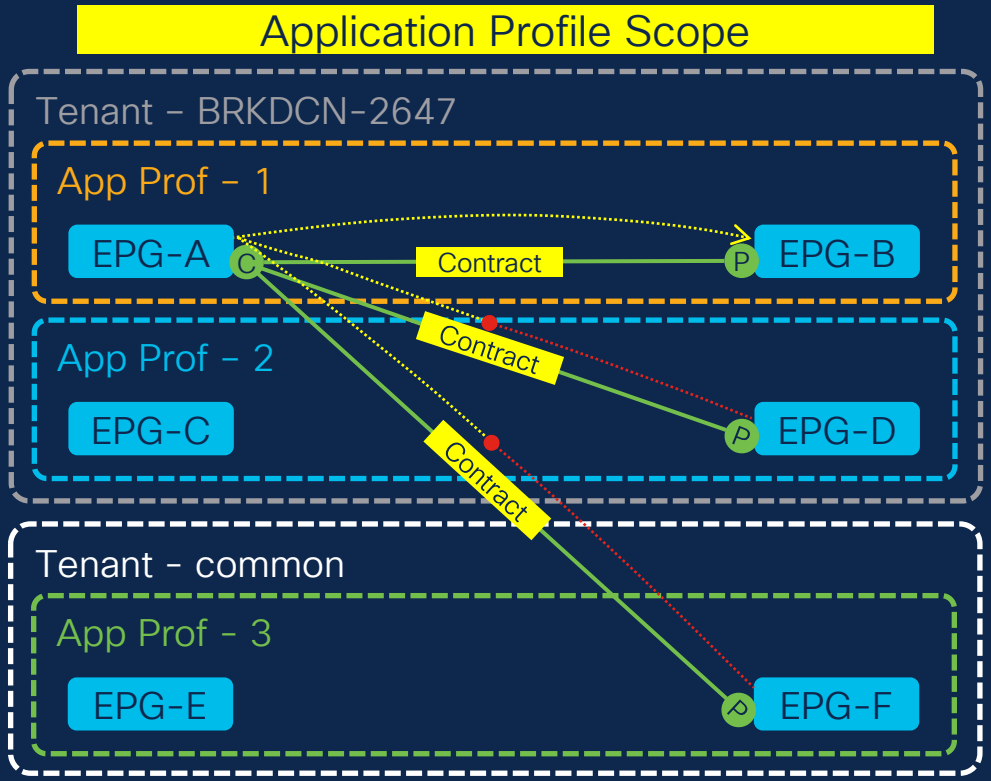


Default Contract Scope – VRF*

Contracts only permit EPGs to communicate within the same the VRF.

* 1:1 Tenant:VRF makes VRF scope functionally the same as Tenant scope.

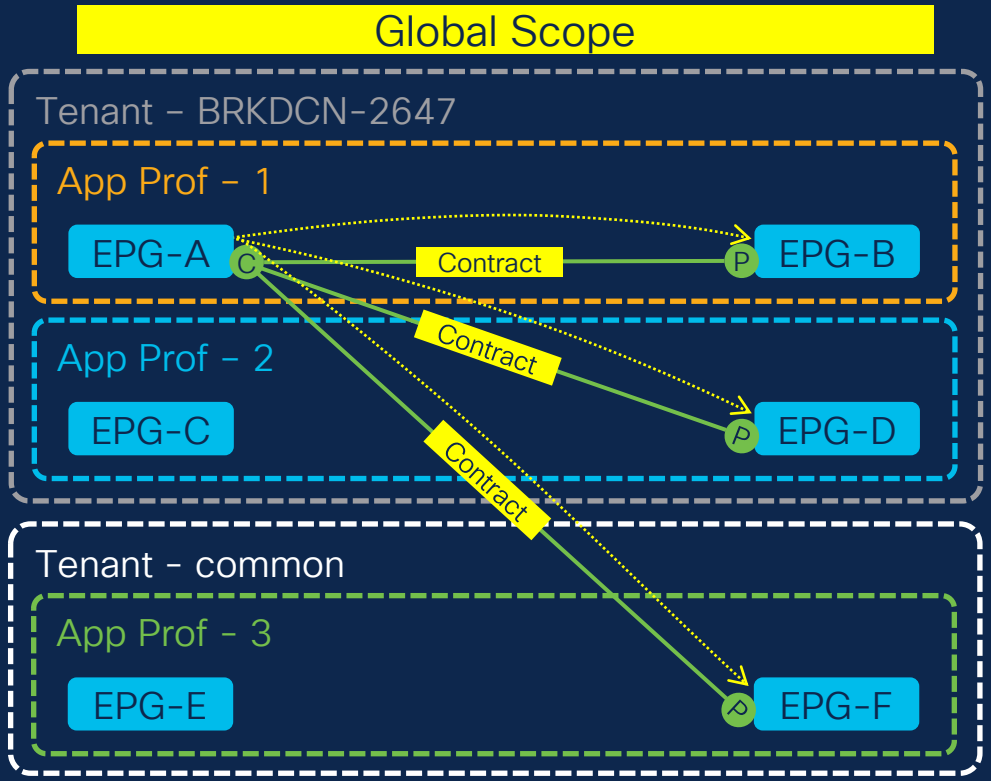
Contract Scope



Contract Scope – Application Profile

Contracts only permit EPGs to communicate within the same the same Application Profile.

Contract Scope



Default Contract Scope – Global

Contracts permit EPGs to communicate across Tenants and VRFs. Used for route-leaking.

Fill out your session surveys!

Attendees who fill out a minimum of four session surveys and the overall event survey will get **Cisco Live-branded socks** (while supplies last)!

Attendees will also earn 100 points in the **Cisco Live Challenge** for every survey completed.

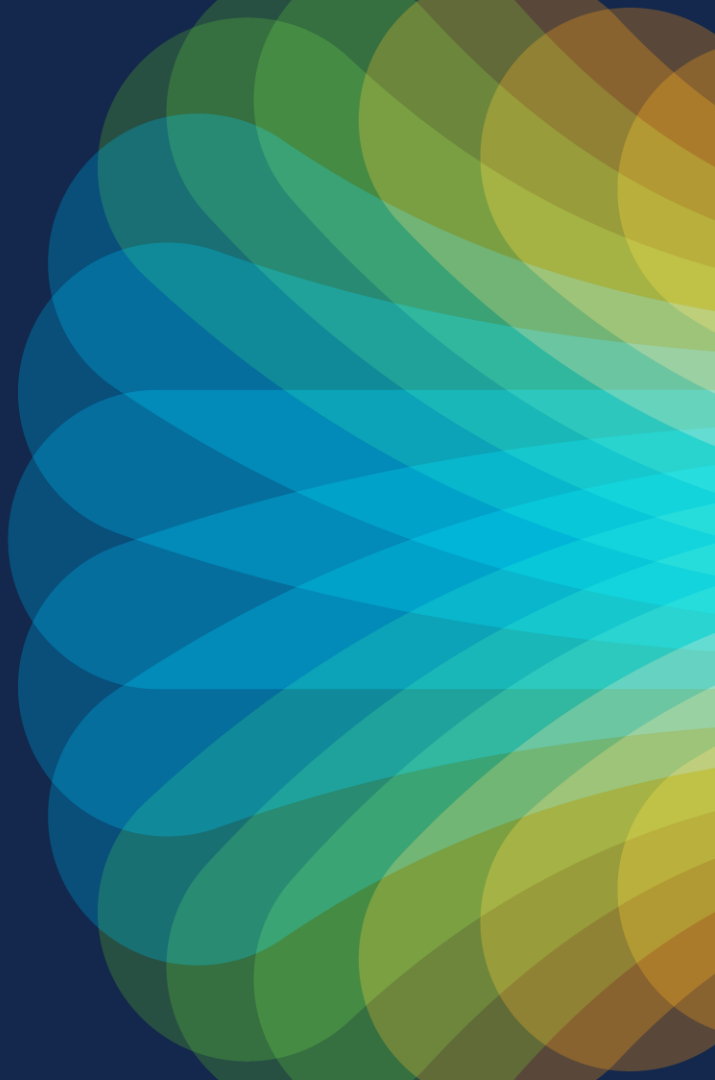
These points help you get on the leaderboard and increase your chances of winning daily and grand prizes

Continue your education

- Visit the Cisco Showcase for related demos
- Book your one-on-one Meet the Engineer meeting
- Attend the interactive education with DevNet, Capture the Flag, and Walk-in Labs
- Visit the On-Demand Library for more sessions at www.CiscoLive.com/on-demand

The bridge to possible

Thank you




cisco Live!
Let's go
#CiscoLive