Alias, Annotations, and Tags

Alias, Annotations, and Tags

To simplify the identifying, addressing, and grouping of objects, ACI provides several methods for the user to add label metadata to objects. These methods are summarized in the list below:

- **Name Alias**: A cosmetic substitute for a GUI entity.
- **Global Alias**: A label, unique within the fabric, that can serve as a substitute for an object's Distinguished Name (DN).
- **Tag Instance / Annotation**: A simple note or description.
- **Policy Tag**: A label for grouping of objects, which need not be of the same class.

Alias

In the ACI object model, every object has a unique Distinguished Name (DN), which is an often lengthy identifier that includes the names of its parent object hierarchy and itself. For example, consider a tenant named **Tenant2468** that contains an application profile named **ap13**, which contains an application endpoint group named **aepg35**. The DN of that application endpoint group, generated by APIC, is **uni/tn-Tenant2468/ap-ap13/epg-aepg35**. After each of these objects is created, ACI typically does not allow their names to be changed, as that would cause a change in the DNs of all descendant objects of the renamed object. To overcome this inconvenience, ACI provides two alias functions — Name Alias for the GUI and Global Alias for the API.

Name Alias

The Name Alias feature (or simply "Alias" where the setting appears in the GUI) changes the displayed name of objects in the APIC GUI. While the underlying object name cannot be changed, the administrator can override the displayed name by entering the desired name in the **Alias** field of the object properties menu. In the GUI, the alias name then appears along with the actual object name in parentheses, as *name_alias (object_name)*. Many object types, such as tenants, application profiles, bridge domains, and EPGs, support the alias property. In the object model, the name alias property is `objectClass.nameAlias`. The property for a tenant object, for example, is `fvTenant.nameAlias`.

Using the preceding example of a tenant, suppose the administrator prefers to see the tenant name "AcmeManufacturing" instead of "Tenant2468." By entering the preferred name in the **Alias** field of the Tenant2468 tenant properties, the GUI would now display **AcmeManufacturing (Tenant2468)**.

The name alias property is purely cosmetic for the APIC GUI. The alias need not be unique in any scope, and the same value can be used as name alias for other objects.

Global Alias

The Global Alias feature simplifies querying a specific object in the API. When querying an object, you must specify a unique object identifier, which is typically the object's DN. As an alternative, this feature allows you to assign to an object a label that is unique within the fabric. Using the preceding example, without a global alias, you would query the application endpoint by its DN using this API request:

```
GET: https://APIC_IP/api/mo/uni/tn-Tenant2468/ap-ap13/epg-aepg35.json
```

By configuring a simpler yet unique name in the **Global Alias** field of the object properties menu, you can use the global alias along with a different API command to query the object:

```
GET: https://APIC_IP/api/alias/global_alias.json
```

Using the preceding example, by entering "AcmeEPG35" in the **Global Alias** field of the application endpoint group's configuration properties, the query URL would now be:

```
GET: https://APIC_IP/api/alias/AcmeEPG35.json
```

In the APIC object model, the global alias is a child object (tagAliasInst) attached to the object that is being aliased. In the preceding example, the global alias object would be a child object of the application endpoint group object.

For additional information, see the "Tags and Aliases" chapter of the *APIC REST API Configuration Guide*.

Creating a Name Alias or Global Alias

This example procedure shows you how to create a name alias and a global alias for an application profile of a tenant. Many other objects support these alias features using the same procedure after navigating to the object.

Step 1 On the menu bar, choose **Tenants** and select the applicable Tenant.

Step 2 In the **Navigation** pane, expand ***tenant_name*** > **Application Profiles** > ***application_profile_name***.

Step 3 In the **Work** pane, click the **Policy** tab.

The **Properties** page of the application profile appears.

Step 4 In the **Alias** field, enter a name alias.

The alias need not be unique in any scope.

Step 5 In the **Global Alias** field, enter an alias for the distinguished name (DN) of the application profile.

The global alias must be unique within the fabric.

Step 6 Click **Submit**.

If you configured a name alias, the application profile is now identified in the **Navigation** pane as *alias* (*name*). For example, if the **Name** is **ap1234** and you configured an **Alias** as **SanJose**, the application profile appears as **SanJose** (**ap1234**).

If you configured a global alias, you can now substitute that value for the distinguished name (DN) of the application profile in API commands that support the global alias.

Annotations

You can add arbitrary key:value pairs of metadata to an object as annotations (`tagAnnotation`). Annotations are provided for the user's custom purposes, such as descriptions, markers for personal scripting or API calls, or flags for monitoring tools or orchestration applications such as Cisco Multi-Site Orchestrator (MSO). Because APIC ignores these annotations and merely stores them with other object data, there are no format or content restrictions imposed by APIC.

Evolution of Annotations

APIC support for user-defined annotation information has changed over time in the following steps:

- Prior to Cisco APIC Release 4.2(4), APIC supported tag instances (`tagInst`), which stored a simple string. In APIC GUI menus, these were labeled as "Tags."
- In Cisco APIC Release 4.2(4), because many modern systems use a key and value pair as a label, changes were made to move to key:value annotations (`tagAnnotation`) as the main label option for API. The shortcut API to query objects via tag instances (**/api/tag/***your_tag***.json**) was deprecated. The APIC GUI continued to use the simple string tag instances (`tagInst`), labeled as "Tags.".
- In Cisco APIC Release 5.1(1), tag instances (`tagInst`) were deprecated in the GUI. GUI menus still used the term "Tags," but actually configured annotations (`tagAnnotation`). Also beginning with this release, a list of all annotations can be viewed from **Fabric** > **Fabric policies** > **Tags**.
- In Cisco APIC release 5.2(1), GUI menu labels were changed from "Tags" to "Annotations." This change was made to avoid confusion with Policy Tags.

Creating an Annotation

This example procedure shows you how to create an annotation for a tenant. Many other objects support the annotation feature using the same procedure after navigating to the object.

Step 1 On the menu bar, choose **Tenants** and select the applicable Tenant.

Step 2 In the **Navigation** pane, click the ***tenant_name***.

Step 3 In the **Work** pane, click the **Policy** tab.

The properties menu of the tenant appears.

Step 4 Next to **Annotations**, click the **+** symbol to add a new annotation.

Step 5 In the annotation key box, choose an existing key or type a new key.

Step 6 In the annotation value box, type a value.

Allowed characters for key and value are a-z, A-Z, 0-9, period, colon, dash, or underscore.

Step 7 Click the ✓ symbol to save the annotation.

You can add more annotations by repeating these steps.

Policy Tags

Policy tags (`tagTag`), or simply tags, are user-definable key and value pairs for use by ACI features. You can configure multiple tags on a single object, and you can apply the same tag on multiple objects. Because many object classes support policy tags, you can use policy tags to group disparate objects. For example, a policy tag can be used to group endpoints, subnets, and VMs together as one Endpoint Security Group (ESG) using ESG tag selectors in Cisco APIC Release 5.2(1).

ACI features using policy tags include:

- Endpoint Security Group (ESG)

Creating a Policy Tag

This example procedure shows you how to create a policy tag for a static endpoint. Several other objects support policy tags using the same procedure after navigating to the object.

Step 1 On the menu bar, choose **Tenants** and select the applicable Tenant.

Step 2 In the **Navigation** pane, expand ***tenant_name*** > **Application Profiles** > ***application_profile_name*** > **Application EPGs** > ***application_epg_name*** > **Static Endpoint**.

Step 3 In the **Work** pane, double-click the static endpoint to be tagged.

The Static Endpoint properties dialog box appears.

Step 4 Next to **Policy Tags**, click the **+** symbol to add a new policy tag.

Step 5 In the tag key box, choose an existing key or type a new key.

Step 6 In the tag value box, type a tag value.

Allowed characters for key and value are a-z, A-Z, 0-9, period, colon, dash, or underscore.

Step 7 Click the ✓ symbol to save the tag.