CA Performance Center

Web Services API Guide

2.4.1

CA Technologies Product References

This document references the following CA Technologies products:

- CA Performance Center
- CA Infrastructure Management
- CA Application Delivery Analysis
- UC Monitor
- CA Network Flow Analysis

Contact CA Technologies

Contact CA Support

For your convenience, CA Technologies provides one site where you can access the information that you need for your Home Office, Small Business, and Enterprise CA Technologies products. At http://ca.com/support, you can access the following resources:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

Providing Feedback About Product Documentation

If you have comments or questions about CA Technologies product documentation, you can send a message to techpubs@ca.com.

To provide feedback about CA Technologies product documentation, complete our short customer survey which is available on the CA Support website at http://ca.com/docs.

Contents

Chapter 1: CA Performance Center API Overview

This section contains the following topics:

CA Performance Center Web Services

CA Performance Center offers a set of APIs that let you automate provisioning and configuration tasks. The most frequently repeated or time-consuming tasks are exposed to you by means of web services.

Some of these APIs consist of RESTful web services. *REST*, or Representational State Transfer, refers to a method of structuring software for the World Wide Web or other applications that conform to the requirements of HTTP in client-server networks. The REST model lets you access a set of resources by means of a fixed set of operations. This model takes advantage of widely deployed HTTP features that are supported by common hardware, such as gateway devices.

The CA Performance Center RESTful web services can programmatically perform the following tasks:

- Create and update user accounts
- Create containers for MSP customer sites ("tenants")
- Create and manage groups
- Import and export dashboards
- Manage data sources
- Create, edit, and delete SNMP profiles
- Create IP domain definitions and associate them with tenants
- Provide lists of all configuration items, such as custom user accounts, roles, or groups, that are already in the system

Additional APIs let you create custom data views and run custom database queries to gain access to precise data sets and build new dashboards. For more information, see the *RIB API User Guide*. And a SOAP API lets you change settings in the Single Sign-On authentication component.

Endpoints and the XML Schema

All endpoints act on a common set of data. The data can be represented in different data formats (i.e. MIME types), depending on the endpoint that consumes or produces the data. An XML schema describes the data and other supported data formats, such as JSON.

This guide describes the basic terms and parameters of the XML schema that you need to know to create scripts for the CA Performance Center RESTful web services. Data can be grouped by namespace. A schema describes the types and elements of each namespace. *Types* define the structure of the data, while *elements* are instances of a type. For example, elements are usually produced or consumed by a REST endpoint, and the structure of each element is described by its type.

Basic Operations in REST Web Services

The REST specification leaves room for some flexibility. As a result, REST web services can use basic HTTP syntax to perform different tasks. All of the CA Performance Center REST web services require user and password authorization in the HTTP request header. You can use the same credentials that you use to log in from the CA Performance Center user interface.

In this implementation, the basic REST commands are used as follows:

- GET - Log in or retrieve information from a server database. Only requires a browser session.
- POST - Create an object. Often requires XML input to supply parameters.
- PUT - Edit an existing object. Occasionally also used to create an object. Usually does not require XML input to supply parameters.
- DELETE - Delete an existing object.

Here is an example of a simple PUT operation that updates the description parameter of a tenant:

`http://`*`CA Performance Center Server IP Address`*`:8181/pc/center/webservice/tenants/tenantName/`*`tenantName`*`/`*`tenantDescription`*`/`*`NewDescription`*

You would substitute the desired values for the italicized terms. Some parameters are required:

tenantName

The name of the tenant that you want to edit.

tenantDescription

The new description to identify this tenant.

Here is an example of a simple GET operation that returns a list of tenant IDs and names using the tenants web service:

```
http://CA Performance Center Server IP Address:8181/pc
/center/webservice/tenants/idNames
```

The following XML is returned:

```
<?xml version="1.0" encoding="UTF-8"?>

<idNames>

  <idName value="tenantAccountId" />

  <idName value="tenantItemId" />

  <idName value="tenantName" />

</idNames>
```

HTTP requests always return a response and a status code, even when successful. The response text is either the expected result or an error message to indicate a problem. The status code is 200 for a successful response, or a numeric error indicator. The following HTTP response code ranges are used:

- 200 - Command status is 'OK'.
- 400 - A user error has occurred. Errors in this range indicate a problem with the input text (400) or the user credentials (403) and can usually be easily corrected.
- 500 - A system error occurred. Errors in this range typically indicate a system fault. Such errors can require assistance from CA Technical Support to resolve them.

For more information about HTTP status codes, see the following IETF website:

http://www.ietf.org/rfc/rfc2616.txt

Accessing the API

API components are automatically installed with the CA Performance Center software. You can run the web services from a web browser. The launch page includes a list of the available web services, endpoint addresses, and WADL and WSDL URIs.

Access the launch page using the following URL syntax:

`http://`*`CA Performance Center server IP address`*`:8181/pc/center`

If you use a testing utility to run web service calls, you receive feedback that is useful for debugging purposes. For example, you can test your scripts using the soapUI open source testing utility. Using a testing utility is also a timesaver. You can supply username and password parameters as service endpoints for automatic authentication of all service calls.

Such utilities require a WSDL file (an XML file that conforms to the Web Services Description Language) that describes the service being tested. In the REST format, the simpler Web Application Description Language (WADL) is used instead. The CA Performance Center API launch page gives you access to a WADL file for each web service that you can use for testing. A link to the WSDL is provided for the SOAP web services.

Chapter 2: Administration Web Services

This section contains the following topics:

About Administration Web Services

The API provides web services to let you perform common administration tasks. For example, you can create tenant and IP domain definitions, SNMP profiles, user accounts, and custom menus.

Check the CA Performance Center Documentation Bookshelf for use cases that describe the workflows to use web services to automate administration tasks. The section titled "Use Cases" contains multiple use cases for administrators who plan to deploy web services.

Businesshours Web Service

The CA Performance Center API provides web services to let you perform common management tasks.

Use the businesshours RESTful web service to create and manage business-hour definitions. Use business-hour definitions to filter views and highlight critical data. CA Performance Center administrators can create sets of business-hour definitions to enhance reporting. They can use the businesshours web service to add and delete multiple business-hour definitions and assign them to site groups.

A use case document is available on the CA Performance Center Documentation Bookshelf to help you use the businesshours web service.

Businesshours Web Service Example Syntax

The following URL shows all of the available operations against the businesshours end point and the required parameters:

`http://CA Performance Center Server IP Address:8181/pc/center/rest/businesshours/documentation`

The following URL returns the parameters for the businesshours web service:

`http://CA Performance Center Server IP Address:8181/pc/center/webservice/businesshours`

A GET operation to the root /businesshours/ returns a list of all business-hour definitions for the tenant of the currently logged-in administrator:

- If you are logged in as a global administrator, you see a list of business hours for the Default Tenant only.
- If you are logged in as a tenant administrator, you see a list of business-hour definitions for your tenant.

Available Businesshours Web Service Methods

Retrieve the list of all the business hours that are associated with the tenant for the logged-in user:

`http://CA Performance Center Server IP Address:8181/pc/center/webservice/businesshours`

Note: The global administrator sees only the business-hour definitions that fall within the Default Tenant. A tenant administrator sees only the definitions within that tenant.

Retrieve the list of all of the business hours that are associated with the specified tenant ID:

`http://CA Performance Center Server IP Address:8181/pc/center/webservice/businesshours/tenantId/tenantId`

Note: You must be logged in as either the global administrator or the tenant administrator of the specified tenant.

Retrieve the business-hour definition that is associated with the specified identifier (the internally assigned businesshour ID):

`http://CA Performance Center Server IP Address:8181/pc/center/webservice/businesshours/id/id`

Delete a business-hour definition:

```
http://CA Performance Center Server IP Address:8181/pc
/center/webservice/businesshours/id/id
```

Note: The user who is running the DELETE method must be logged in as either the global administrator or the tenant administrator of the tenant that owns the business hour ID.

Available PUT Methods for the Businesshours Web Service

Associate a business hour definition with a specified site group:

```
http://CA Performance Center Server IP Address:8181/pc
/center/webservice/businesshours/assign/businesshour
/businessHour/site/siteGroupId
```

Remove the association of the business hour definition currently assigned to the specified site group:

```
http://CA Performance Center Server IP Address:8181/pc
/center/webservice/businesshours/unassign/businesshour
/site/siteGroupId
```

Associating Time Zones

To avoid errors associated with slash (/) characters in time zones, the web service call accepts either the integer ID (ID column of the timezones table) or the language-agnostic ID (the TimeZoneID column of the timezones table). If you want to use the language-agnostic ID, simply replace any slashes with %2f.

Example: If you have a site group with a siteId of 118, and you want to set the time zone to be America/Denver (TimeZone ID of 93), perform a PUT operation with either of the following URLs:

```
http://CA Performance Center Server IP Address:8181/pc
/center/webservice/businesshours/assign/timezone/93/site/118
```

or:

```
http://CA Performance Center Server IP Address:8181/pc
/center/webservice/businesshours/assign/timezone/America%2fDenver
/site/118
```

To get a list of all available time zones, use the following syntax:

`http://CA Performance Center Server IP Address:8181/pc/center/webservice/consoleinfo/allTimezones/cultureID`

The results that are returned include the localized name of the time zone based on the specified culture ID, as well as the language-agnostic TimeZone ID.

Site groups are required to deploy business hours. For more information about site group syntax, see Syntax for Site Group Management (see page 32).

Dashboards Web Service

The CA Performance Center API provides web services to let you perform dashboard management tasks.

Use the dashboards RESTful web service to create and manage dashboard pages to display data views. You can create new dashboards. You can also use this web service to import and export dashboards. The import feature lets you build the dashboards once and deploy them on a second CA Performance Center server or in an additional tenant without having to manually recreate these dashboards.

You can also use the web service to test the results of an import operation. The automated test imports a custom page file and verifies that it exists in the main CA Performance Center database.

Dashboards Web Service Example Syntax

Issue the following call to see the parameters for the dashboards web service:

`http://`*`CA Performance Center Server IP Address`*`:8181/pc`
`/center/rest/dashboards/documentation`

GET Method (Export)

Perform a GET to export a dashboard to an XML file that is suitable for importing into another instance of CA Performance Center.

To export a dashboard, the internally assigned page ID is required. Use the product user interface to find the ID, as described in the following steps:

1. Navigate to the dashboard to export.
2. In the browser window, find the page ID in the URL, such as, for example:

   http://*CA Performance Center Server Hostname*:8181/pc/desktop/page?pg=2000040

   The page ID is 2000040.
3. In a REST client, set the URL to the following:

   `http://`*`CA Performance Center Server IP address`*`:8181/pc/center/webservice/dashboards/`*`pageId`*
4. Perform a GET operation.

   An error or success message appears in the response.

POST Method (Import)

You can import a dashboard from an XML file.

1. Set the URL to the following:

   `http://`*`CA Performance Center Server IP Address`*`:8181/pc/center/webservice/dashboards/import`
2. Either browse to select the XML file that represents a dashboard that you have exported, or paste the exported text of the XML file into the Body field.

   Checking is performed for the unique MenuItem and DashboardTitle.
3. Verify the following properties at the top of the file:

   ```
   <menuItem>Custom Dashboards</menuItem>
   <dashboardTitle>MyCustomDashboard_Example</dashboardTitle>
   ```

   These properties must be unique per tenant.
4. Perform a POST operation.

   An error or success message appears in the response.

Note: The web service assigns a page ID to the imported dashboard.

Test the import of a dashboard before you actually import it. You can view the results without performing the import operation.

Issue the following call:

`http://CA Performance Center Server IP Address:8181/pc/center/webservice/dashboards/test/`

The results include the page ID for the imported dashboard.

PUT Method (Update Dashboard)

You can update a dashboard based on a page ID. Edit the XML to modify the dashboard.

1. Set the URL to the following:

   `http://hostname:8181/pc/center/webservice/dashboards/pageId`

2. Paste the XML file that represents the dashboard into the Body field.
3. Perform a PUT operation.

   An error or success message appears in the response.

Datasources Web Service

The API provides the datasources web service to let you get a current list of registered data sources, add (register) new data sources, modify data source settings, such as the authentication method to use, and remove (unregister) data sources. You can also synchronize data sources and manage the data source log file.

Issue the following call to see the available operations and parameters for the datasources web service:

`http://CA Performance Center Server IP Address:8181/pc/center/rest/datasources/documentation`

Basic Datasource Parameters

The current values for the following data source settings are available from the GET command:

id

An internally assigned identifier for the data source.

enabled

Indicates whether the data source is enabled.

Default: true

name

The hostname of the data source.

authtype

The type of authentication to use for this data source. One of the following:

NONE

BASIC

type

The data source type. One of the following:

REPORTER: CA Network Flow Analysis (formerly CA ReporterAnalyzer)

SUPER_AGENT: CA Application Delivery Analysis (formerly CA SuperAgent)

E_HEALTH: CA eHealth

VOIP_MONITOR: CA Unified Communications Monitor

ANOMALY_DETECTOR: CA Anomaly Detector

ALLOCATE: NetQoS Allocate (not supported by CA Performance Center)

F5: F5 BIG-IP load balancer (not supported by CA Performance Center)

EVENT_MANAGER: CA Event Manager

PATH_LATENCY: CA Trade Monitor (not supported by CA Performance Center)

TRADE_MONITOR: CA Trade Monitor (not supported by CA Performance Center)

NETQOS_API: The NetQoS reporting API for Microsoft Excel (not supported by CA Performance Center)

NEXUS: A Cisco Wide-Area Application Services (WAAS) data source (not supported by CA Performance Center).

SPECTRUM_IM: CA Spectrum

APM: CA Application Performance Management

DATA_AGGREGATOR: CA Infrastructure Management Data Aggregator

CATALYST_CONNECTOR: The CA Catalyst Connector for CA Performance Center

SERVICE_OPERATIONS_INSIGHT: CA Service Operations Insight

RESERVED_CUSTOMER_N: An enum that has been reserved for unspecified custom uses.

consoleSameAsDataSource

Indicates whether the data source Web console address is the same as the hostname. Use this parameter in cases where network address translation is deployed.

Default: true

consoleAddress

Is the IP address of the data source console; the address where the data source should be contacted by CA Performance Center.

Remove a Data Source

Typically, you can unregister a data source by removing it on the Manage Data Sources page in the CA Performance Center Console. However, in certain circumstances, the unregister operation cannot be completed. For example, the data source server might be offline or inoperable. In such cases, use web services to remove the data source so that it can later be registered to another CA Performance Center instance.

To get the data source ID, issue a GET to the following URL:

CA Performance Center Server IP Address:8181/pc
/center/webservice/datasources

The XML that is returned contains a list of all data sources that are registered with CA Performance Center and their properties.

Follow these steps:

1. Enter a URL for the CA Performance Center RESTful web services API in the REST client. Use the following format:

   http://*CA Performance Center Server IP Address*:8181/pc
   /center/webservice/datasources/*idName*/*idValue*

   Note: The data source ID can be found by running the GET operation that is discussed above.

2. Select DELETE for "HTTP" Method.
3. Provide a valid Username and Password for a user account that has administrator access to CA Performance Center.
4. Select 'application/xml' as the 'Body Content-type' in the Body settings.
5. Run the method.

Resynchronize a Data Source

Use the datasources web service to resynchronize data source databases with CA Performance Center.

The following URL shows the syntax:

CA Performance Center Server IP Address:8181/pc
/center/webservice/datasources/resyncDataSources

Follow these steps:

1. Enter the following URL in the REST client:

   http://*CA Performance Center Server IP Address*:8181/pc
   /center/webservice/datasources/resyncDataSources

2. Select PUT for "HTTP" Method.
3. Provide a valid Username and Password for a user account that has global administrator access to CA Performance Center.
4. Select 'application/xml' as the 'Body Content-type' in the Body settings.
5. Run the method.

   The XML that is returned includes a timestamp that shows when the last synchronization request completed. You can pass this value to the 'wait for resync' method, which accepts a requestedOn parameter.

Devices Web Service

The API provides the devices web service to let you get a current list of devices that belong to a tenant. You can also retrieve a list of interfaces from a device (a router or switch) in the inventory of managed items.

All devices are associated with a tenant. If you are not deploying multi-tenancy, devices are associated with the Default Tenant. Run the getlist method to the root /devices/ to retrieve a list of devices that are associated with your tenant.

Issue the following call to see the parameters for the devices web service:

http://*CA Performance Center Server IP Address*:8181/pc/center/webservice/devices

Issue the following call to see a list of supported operations:

http://*CA Performance Center Server IP Address*:8181/pc
/center/rest/devices/documentation

Available GET Methods

get list

Gets a list of devices that belong to the tenant of the logged-in user.

`http://CA Performance Center Server IP Address:8181/pc/center/webservice/devices`

get interfaces

Gets a list of all interfaces that belong to a specified device.

`http://CA Performance Center Server IP Address:8181/pc/center/webservice/devices/idName/idValue/interfaces`

The following URL lets you retrieve a list of internally assigned identifiers that can be used in other methods to identify devices:

`http://CA Performance Center Server IP Address:8181/pc/center/webservice/devices/idNames`

Or you can retrieve a list that is filtered by tenant and by device subtype as follows:

`http://CA Performance Center Server IP Address:8181/pc/center/webservice/devices/subtype/tenant/tenantIdName/tenantIdValue`

The subtype further describes the device. For example, the server, switch, and router subtypes identify devices, while an interface can have subtype "physical" or "virtual".

Note: A GET to the following URL for the groups web service returns a list of group members that includes their subtype:

`http://CA Performance Center Server IP Address:8181/pc/center/webservice/groups/idName/idValue/items`

Set the Alias Name for a Device Example

You can set an alias name for a single monitored device. The alias name appears in the inventory list of devices and in the inventory list of interfaces.

Note: An alias that is set using REST web services takes precedence over the alias that you can set by importing a CSV file. For information on importing a CSV file, see the *CA Performance Center Administrator Guide*.

To set an alias name for a single monitored device, run a PUT to the following URL:

`http://CA Performance Center Server IP Address:8181/pc/center/webservice/devices/deviceItemId/device_id/aliasName/alias_name`

device_id

Is the device item identification number for the monitored device for which you are setting an alias name.

alias_name

Is the alias name for the monitored device.

Example

Run a PUT to the following URL to set 'Alias name for 96.24' as the alias name for a device with a device item ID of 119:

```
http://My_Server_Name:8181/pc/center/webservice/devices/
deviceItemId/119/aliasName/Alias name for 96.24
```

Note: To set the aliases for multiple device items, use the update_alias_name.sh script. For information on importing a CSV file, see the *CA Performance Center Administrator Guide*.

Set Interface Name Aliases

As an administrator, you can set aliases for interface component names per device item. An interface alias is an administrator-configured name that is applied to the associated interface component in CA Performance Center. Users see the interface aliases in their dashboards and views depending on the role rights you assign.

Note: An alias that is set using REST web services takes precedence over the alias that you can set by importing a CSV file. For information on importing a CSV file, see the *CA Performance Center Administrator Guide*.

The following URL shows the syntax:

http://*CA Performance Center Server IP Address*:8181/pc/center/webservice/devices

Follow these steps:

1. Determine which interfaces you want to set an alias for:
   a. To return a list of all of the interfaces on a device item, enter the following URL in the REST client:

      `http://CA Performance Center Server IP Address:8181/pc/center/webservice/devices/deviceItemId/device_id/interfaces`

device_id

Is the device item identification number for the monitored device that interface components are associated with.

b. Select GET for the "HTTP" Method.

c. Determine which interfaces you want to set an alias for.

2. Enter one of the following URLs in the REST client:

   - http://*CA Performance Center Server IP Address*:8181/pc/center/webservice/devices/deviceItemId/*device_id*/setInterfaceNameAlias
   - http://*CA Performance Center Server IP Address*:8181/pc/center/webservice/devices/domainItemId/*domain_id*/*device_IP*/setInterfaceNameAlias

   domain_id

   Is the domain identification number for the domain that interface components are associated with.

3. Select PUT for the "HTTP" Method.

4. To set the interface name aliases, enter the following information in the Body tab of the HTTP Request pane:

```
<interfaces>
    <interface>
        <itemId>interface_item_id</itemId>
        <nameAlias>alias_for_interface</nameAlias>
    </interface>
    ...
</interfaces>
```

Example

To set 'Et0 - alias' and 'Se0 - alias' as the interface name alias for two interfaces, enter the following information in the Body tab of the HTTP Request pane:

```
<interfaces>

    <interface>

         <itemId>164</itemId>

         <nameAlias>Et0 - alias</nameAlias>

    </interface>

    <interface>

         <itemId>165</itemId>

         <nameAlias>Se0 - alias</nameAlias>

    </interface>

</interfaces>
```

Note: To set the aliases for interface component names across multiple device items, use the update_alias_name.sh script. For information on using the script to set alias names, see the *CA Performance Center Administrator Guide*.

Set Component Name Aliases

As an administrator, you can set aliases for component names per device item. A component alias is an administrator-configured name that is applied to the associated component in CA Performance Center. Users see the component aliases in their dashboards and views depending on the role rights you assign.

Note: An alias that is set using REST web services takes precedence over the alias that you can set by importing a CSV file. For information on importing a CSV file, see the *CA Performance Center Administrator Guide*.

The following URL shows the syntax:

http://*CA Performance Center Server IP Address*:8181/pc/center/webservice/devices

Follow these steps:

1. Determine which components you want to set an alias for:
   a. To return a list of all of the components on a device item, enter one of the following URLs in the REST client:
   - http://*CA Performance Center Server IP Address*:8181/pc/center/webservice/devices/deviceItemId/*device_id*/components

     device_id

     Is the device item identification number for the monitored device that components are associated with.

   - http://*CA Performance Center Server IP Address*:*8181/pc/center/webservice/devices/*domainItemId*/domain_id/device_IP/components*

   domain_id

   Is the domain identification number for the domain that components are associated with.

   a. Select GET for the "HTTP" Method.

   b. All components on the device are returned. Determine which components you want to set an alias for.

2. Enter one of the following URLs in the REST client:
   - http://*CA Performance Center Server IP Address*:*8181*/pc/center/webservice/devices/deviceItemId/*device_id*/setComponentNameAlias
   - http://*CA Performance Center Server IP Address*:*8181*/pc/center/webservice/devices/domainItemId/*domain_id*/*device_IP*/setComponentNameAlias
3. Select PUT for the "HTTP" Method.

4. To set the component name aliases, enter the following information in the Body tab of the HTTP Request pane:

```
<deviceComponents>
    <deviceComponent>
        <itemId>interface_ item_ id</itemId>
        <nameAlias> alias_for_component</nameAlias>
    </<deviceComponents>
    ...
</deviceComponents>
```

Note: To set the aliases for for component names across multiple device items, use the update_alias_name.sh script. For information on importing a CSV file, see the *CA Performance Center Administrator Guide*.

Domains Web Service

The API provides the domains web service to let you get a current list of IP domain definitions, create new IP domain definitions, and modify those definitions.

Managed items are associated with IP domains by the data sources during data collection. Consult the *Administrator Guide* for each registered data source to determine the steps to take to associate items with domains.

Domains Web Service Example Syntax

Issue the following call to see the parameters for the domains web service:

```
http://CA Performance Center Server IP
Address:8181/pc/center/webservice/domains
```

For more information, see Basic IP Domain Parameters (see page 28).

Issue the following call to see a list of supported operations:

```
http://CA Performance Center Server IP Address:8181/pc
/center/rest/domains/documentation
```

Available GET Methods

get list

Gets a list of all of the IP domains that belong to the tenant for the logged-in user.

Run the getList method to the root /domains/ to retrieve a list of all IP domain IDs for the tenant of the currently logged-in administrator:

- If you are logged in as a global administrator, you see a list of domain IDs for the Default Tenant only.
- If you are logged in as a tenant administrator, you see a list of domain IDs for your tenant.

get

Retrieves information about a specified IP domain.

```
http://CA Performance Center Server IP Address:8181/pc
center/webservice/domains/idName/idValue
```

Note: The global administrator sees only the IP domains that fall within the Default Tenant. A tenant administrator sees only the IP domains within that tenant.

get domain for group

Gets the IP domain associated with a specified group.

```
http://CA Performance Center Server IP Address:8181/pc
/center/webservice/domains/group/groupIdName/groupIdValue
```

For more information, see Groups Web Service Example Syntax (see page 29).

get id names

Retrieves a list of identifiers that can be used to identify IP domains in other web service operations.

```
http://CA Performance Center Server IP Address:8181/pc
/center/webservice/domains/idNames
```

get list

Gets all of the IP domains that belong to the tenant for the logged-in user.

`http://CA Performance Center Server IP Address:8181/pc /center/webservice/domains`

get list by tenant

Retrieves the list of all of the IP domains that are associated with the specified tenant ID:

`http://CA Performance Center Server IP Address:8181/pc /center/webservice/domains/tenantItemId/tenantId`

get list with translation

Gets all of the IP domains that belong to the tenant for the logged-in user. Any localized text is translated to the specified language.

`http://CA Performance Center Server IP Address:8181/pc /center/webservice/domains/cultureId`

Available POST Method

create

Creates an IP domain.

`http://CA Performance Center Server IP Address:8181/pc /center/webservice/domains`

Available PUT Method

update

Updates a specified IP domain.

`http://CA Performance Center Server IP Address:8181/pc /center/webservice/domains/idName/idValue`

Available DELETE Method

delete

Deletes an IP domain definition.

`http://CA Performance Center Server IP Address:8181/pc /center/webservice/domains/{idName}/{idValue}`

Basic IP Domain Parameters

The current values for the following domain parameters are available from the GET command:

cultureID

Specifies a language (locale). Supply a language identifier from the following list:

- en-US (English, United States)
- ja-JP (Japanese)
- zh-CN (Simplified Chinese)
- fr-FR (French, France)

dnsProxyAddress

Is the IP address of the DNS proxy server.

domainItemID

Is an internal (database) identifier for a tenant definition.

groupItemID

Is an internal (database) identifier for the group definition associated with a domain.

itemDesc

Describes this domain namespace, such as naming the enterprise that owns it.

itemName

Identifies the domain.

tenantID

Is an internal (database) identifier for a tenant definition. Identifies the tenant associated with this domain.

primaryDNSAddress

Is the IP address of the primary name server for this domain.

primaryDNSPort

Is the port number that the primary name server uses.

secondaryDNSAddress

(Optional) Is the IP address of the secondary name server for this domain.

secondaryDNSPort

(Optional) Is the port number that the secondary name server uses.

isDNSProxyEnabled

Indicates whether the proxy address is enabled for this IP domain.

deviceAlias

Indicates the alias to use for a managed item. A device alias is a user-configured name that is applied to the associated managed item in CA Performance Center.

deviceAliasList

Identifies a CSV or TXT file of alternate interface descriptions. A comma-separated list of values that include the device IP address, interface name, interface description, and alternate interface description (alias) mappings.

interfaceDescriptionOverride

Indicates the alternate description to use for an interface. Overrides the interface descriptions that appear in CA Performance Center by default.

Groups Web Service

The CA Performance Center API provides web services to let you perform common group management tasks.

Use the groups RESTful web service to create and manage groups of monitored items. You can create new groups and add items to them manually. You can also write rules to create and populate groups based on item attributes.

A use case document is available on the CA Performance Center Documentation Bookshelf to help you use the groups web service.

Groups Web Service Example Syntax

Issue the following call to see the parameters for the groups web service:

```
http://CA Performance Center Server IP
Address:8181/pc/center/webservice/groups
```

Issue the following call to see a list of supported operations:

```
http://CA Performance Center Server IP
Address:8181/pc/center/rest/groups/documentation
```

To get a list of all groups under the highest-level group in the Groups tree (the default, 'All Groups') you can use the groupPath parameter or the groupItemId parameter.

Issue the following call to use the groupPath parameter to identify the default group:

```
http://CA Performance Center Server IP
Address:8181/pc/center/webservice/groups/
groupPath/All%20Groups
```

Note: When using some REST clients, the 'All Groups' syntax is required rather than 'All%20Groups'. But in general, blank spaces are not valid in URLs.

Issue the following call to use the groupItemId parameter to identify the default group (whose groupItemId value is 1):

```
http://CA Performance Center Server IP
Address:8181/pc/center/webservice/
groups/groupItemId/1
```

Subgroup Syntax

To get a list of all subgroups under a group that you specify, you have two options:

- Use the groupPath parameter.
- Use the groupItemId parameter.

Issue the following call to use the groupPath parameter to identify the group whose subgroups are listed in the XML that is returned:

```
http://CA Performance Center Server IP
address:8181/pc/center/webservice/
groups/groupPath/All%20Groups%2FInventory
```

Issue the following call to use the groupItemId parameter to identify the group whose subgroups are listed in the XML that is returned:

Note: The groupItemId of the default 'Inventory' group is 5.

```
http://CA Performance Center Server IP
address:8181/pc/center/webservice/
groups/groupItemId/5
```

The XML that is returned includes the syntax of any group rules that are applied to that group. Therefore, we recommend testing with various rules that you create in the user interface and examining the syntax that is generated.

Site Group Syntax

The following XML in a POST command creates a site group (replace the example values with the desired values for the new site group):

```
<GroupTree path="/All Groups">
   <Group name="East Coast USA" desc="This is a site group"
   inherit="true" type="site group" location="North America"
   bHourID="99990" timeZone="EST"/>
</GroupTree>
```

inherit

Indicates whether the group includes child items of group members. For example, if the "inherit" attribute is set to true, device interfaces are group members if the device has been added to the group.

type

Indicates the type of group. Accepts the following values:

- custom group: A group that a user has created.
- site group: A user-created group that represents a physical site.
- system group: A predefined group that cannot be modified or deleted.
- automatic group: A predefined group of items from a data source other than Data Aggregator that cannot be modified or deleted.

bHourID

(Optional) The internally assigned identifier of the business-hour definition that you want to associate with this site group.

timeZone

(Optional) The time zone to associate with this site group. Time zones can only be associated with site groups, not with custom groups.

Group Rules

Group rules support multiple comparisons, in addition to regular expressions. For example, use the following syntax in the XML to post a group rule that adds devices whose name begins with the word 'Cisco':

```
<Match>
   <Compare readOnly="true" using="MEMBER_OF">
       <Property name="ItemID" type="device"/>
           <Value reference="/All Groups">1</Value>
   </Compare>
   <Compare readOnly="false" using="STARTS_WITH">
      <Property name="DisplayName" type="device"/>
           <Value>Cisco*</Value>
   </Compare>
</Match>
```

AllowDeletes

Group deletion requires the 'allowDeletes' parameter to be set to 'true'. Apply this parameter to a container group when you want to delete a subgroup. For example, this XML deletes the following subgroup: All Groups\Texas\Austin:

```
<GroupTree path="/All Groups/Texas">
   Group name="USA" desc="Group to represent the entire
   United States" allowDeletes="true" type="custom group"/>
</GroupTree>
```

For group path syntax, forward-slash characters are appropriate for the XML documents that you post. This example assumes that you already have a group structure of “All Groups\Texas\Austin”:

```
<GroupTree inheritDefault="true" path="/All Groups/Texas/Austin">
    <Group desc="" inherit="true" location="" name="CA Office"
type="custom group">
        <Group desc="" inherit="true" location="" name="Austin Lab"
type="custom group"/>
    </Group>
    <Group desc="" inherit="true" location="" name="Austin Data
Center" type="custom group"/>
</GroupTree>
```

In the URL for the web service request, however, use a backslash character for a group path. Forward slashes are misinterpreted by the browser.

Syntax for Site Group Management

To get a list of all groups under the highest-level group in the Groups tree (the default, 'All Groups') you can use the groupPath parameter or the groupItemId parameter.

Issue the following call to use the groupPath parameter to identify the default group:

```
http://CA Performance Center Server IP
Address:8181/pc/center/webservice/groups/
groupPath/All%20Groups
```

Note: When using some REST clients, the ‘All Groups’ syntax is required rather than ‘All%20Groups’. But in general, blank spaces are not valid in URLs.

Issue the following call to get the identifier (siteId) for a site group:

```
http://CA Performance Center Server IP
Address:8181/pc/center/webservice/
groups/groupItemId/siteId
```

In the XML that is returned, the site ID looks something like this:

```
<GroupTree siteId="118" inheritDefault="true" path="Austin, TX">
```

To get a list of all subgroups under a group that you specify, you have two options:

- Use the groupPath parameter.
- Use the groupItemId parameter.

Issue the following call to use the groupPath parameter to identify the group whose subgroups are listed in the XML that is returned:

```
http://CA Performance Center Server IP
address:8181/pc/center/webservice/
groups/groupPath/All%20Groups%5CInventory
```

Issue the following call to use the groupItemId parameter to identify the group whose subgroups are listed in the XML that is returned:

Note: The groupItemId of the default 'Inventory' group is 5.

```
http://CA Performance Center Server IP
address:8181/pc/center/webservice/
groups/groupItemId/5
```

The XML that is returned includes the syntax of any group rules that are applied to that group. Therefore, we recommend testing with various rules that you create in the user interface and examining the syntax that is generated.

Site Groups and Rules

Group rules support multiple comparisons, in addition to regular expressions. For example, use the following syntax in the XML to post a group rule that adds devices whose name begins with the word 'Cisco':

```
<Match>
   <Compare readOnly="true" using="MEMBER_OF">
       <Property name="ItemID" type="device"/>
           <Value reference="/All Groups">1</Value>
   </Compare>
   <Compare readOnly="false" using="STARTS_WITH">
      <Property name="DisplayName" type="device"/>
           <Value>Cisco*</Value>
   </Compare>
</Match>
```

For group path syntax, forward-slash characters are appropriate for the XML documents that you post. This example assumes that you already have a group structure of “All Groups\Texas\Austin”:

```
<GroupTree inheritDefault="true" path="/All Groups/Texas/Austin">
    <Group desc="" inherit="true" location="" name="CA
Officetype="custom group">
        <Group desc="" inherit="true" location="" name="Austin Lab"
type="custom group"/>
    </Group>
    <Group desc="" inherit="true" location="" name="Austin Data
Center" type="custom group"/>
</GroupTree>
```

In the URL for the web service request, however, use a backslash character for a group path. Forward slashes are misinterpreted by the browser.

Roles Web Service

The API provides the roles web service to let you get a current list of user account roles, create custom roles, and modify those roles.

The *role* is a parameter assigned to a user account that controls user access to product features and dashboard pages. Based on user job functions, the role grants administrative access to product configuration using *role rights*. Roles let users access data and product features that they require to perform their duties and restrict access to features that they do not require. To assign custom or factory roles to user accounts, use the users RESTful web service (see page 39).

Basic Role Parameters

The current values for the following user account role settings are available from the GET command:

accessRights

Are the role rights that are allocated to this role. Multiple role rights are available for allocation, and some are data source-specific. To get a list of all available role rights, the **categoryId** is required. The categoryId with value 1 corresponds to both CA Performance Center and Data Aggregator.

For more information, see Roles Web Service Example Syntax (see page 36).

culture

Specifies a language (locale). Supply a language identifier from the following list:

- en-US (English, United States)
- ja-JP (Japanese)
- zh-CN (Simplified Chinese)
- fr-FR (French, France)

description

(Optional) Describes the role to help you identify it.

enabled

Determines whether the role is enabled for use (activated). Values are true or false.

name

Is a name for the role. The name is limited to 50 characters.

selections

Provides sets of access rights that you can selectively grant to this role, organized into categories.

userCount

Is the number of users who have this role assigned to their user account.

userID

Is an internally assigned value for the role.

Roles Web Service Example Syntax

Issue the following call to see the available operations and parameters for the roles web service:

`http://CA Performance Center Server IP Address:8181/pc/center/rest/roles/documentation`

For more information, see Basic Role Parameters (see page 35).

Available GET Methods

get access rights

Retrieves a list of the role rights that are assigned to a specified role. Use the following syntax:

`http://CA Performance Center Server IP Address:8181/pc center/webservice/roles/idName/idValue/rights/cultureId`

get access rights by category

Retrieves a list of all available role rights for a specified category. The category is either CA Performance Center or a data source. A categoryId of 1 applies to both CA Performance Center and Data Aggregator.

`http://CA Performance Center Server IP Address:8181/pc center/webservice/roles/rights/categoryid/cultureId`

get category

Retrieves an XML document that includes the categoryId:

`http://CA Performance Center Server IP Address:8181/pc /center/webservice/roles/rights/categories/en-US`

get by tenant

Retrieves a list of all roles that are associated with the tenant for the logged-in user. Use the following syntax:

`http://CA Performance Center Server IP Address:8181/pc center/webservice/roles/idName/idValue/rights/tenant/tenantIdName/tenantIdValue/cultureId`

get categoryId

Retrieves XML that shows the available category IDs.

`http://CA Performance Center Server IP Address:8181/pc center/webservice/roles/rights/categories/cultureId`

A value of 1 corresponds to CA Performance Center and Data Aggregator. Role rights in that category only apply to a Data Aggregator data source.

get id names

Retrieves a list of identifiers that can be used to identify roles in other web service methods. Use the following syntax:

`http://CA Performance Center Server IP Address:8181/pc center/webservice/roles/idNames`

Available PUT Methods

copy

Copies a role. This method creates a copy of a specified role and associates it with the tenant for the logged-in user. Use the following syntax:

`http://CA Performance Center Server IP Address:8181/pc /center/webservice/roles/idName/idValue/copy/roleName/description/enabled/cultureId`

For more information, see Basic Role Parameters (see page 35).

update

Modifies a specified role. The role name and tenant ID are required elements of the role parameters.

`http://CA Performance Center Server IP Address:8181/pc center/webservice/roles/idName/idValue/cultureId`

Available POST Methods

create

Creates a role. The new role is associated with the tenant for the logged-in user:

`http://CA Performance Center Server IP Address:8181/pc /center/webservice/roles/`

For more information, see Create a Role (see page 38).

create for tenant

Creates a role and assigns it to the specified tenant.

`http://CA Performance Center Server IP Address:8181/pc /center/webservice/roles/tenant/tenantIdName/tenantIdValue`

For more information, see Tenants Web Service Parameters (see page 46).

Create a Role

Use the roles web service to create user account roles.

Issue the following call to see the parameters for the roles web service:

CA Performance Center Server IP Address:8181/pc
/center/webservice/roles

Follow these steps:

1. Enter a URL for the CA Performance Center RESTful web services API in the REST client. Use the following format:

   http://*CA Performance Center Server IP Address*:8181/pc
   /center/webservice/roles

2. Select POST for "HTTP" Method.
3. Provide a valid Username and Password for a user account that has global administrator access to CA Performance Center.
4. Select 'application/xml' as the 'Body Content-type' in the Body settings.
5. Paste XML in the Body field that resembles the following example:

```
<role>
  <name>TestRoleName</name>
  <description>Test Role Description</description>
  <enabled>true</enabled>
  <accessRights>
  <accessRight>
  <accessRightName>ViewToS</accessRightName>
  <categoryId>1</categoryId>
  <enabled>true</enabled>
  </accessRight>
  </accessRights>
 </role>
```

accessRights

Correspond to role rights.

accessRightName

The name of the role right. For example, the administerGroups role right lets the user with this role manage a limited section of the Groups tree.

categoryId

Identifies the category of role rights, such as CA Performance Center role rights or data-source-specific role rights.

For more information about these parameters, see Basic Role Parameters (see page 35).

6. Run the method.

Users Web Service

The API provides a users web service to let you retrieve a current list of user accounts, create new user accounts, modify user accounts, and delete user accounts. The following topics provide information about related parameters:

- Basic User Account Parameters (see page 39)
- User Account Product Privilege Settings (see page 41)
- Basic Role Parameters (see page 35)

In addition, a use case document is available on the CA Performance Center Documentation Bookshelf to help you use the tenants and users web services.

Basic User Account Parameters

The current values for the following user account settings are available from a GET operation:

`http://CA Performance Center Server IP Address:8181/pc/center/webservice/users/cultureId`.

Note: This URL returns information about users associated with the same tenant as the account that is used to execute this command.

For *cultureId*, supply the language code for the language in which you would like to view the output, such as 'en-US'. See Basic User Account Parameters (see page 39) for the syntax.

userID

Is an internally assigned value for the user account.

name

Is a login name for the user account. The name is limited to 50 characters.

description

(Optional) Describes the user account to help you identify it.

enabled

Determines whether the user account is enabled for use (activated).

removable

States whether the item can be deleted (removed from the database).

Values: true or false.

Note: You cannot delete the two predefined user accounts (**admin** and **user**).

timezone (tz)

Corresponds to the time zone in which the user will view report data.

Default: UTC (Coordinated Universal Time).

userLevel

Identifies the product privilege assigned to this user account. For more information, see User Account Product Privilege Settings (see page 41).

role

Specifies the role assigned to the user account.

tenantId

Is an internal (database) identifier for the tenant with which the user account is associated.

culture

Specifies a language (locale). Supply a language identifier from the following list:

- en-US (English, United States)
- ja-JP (Japanese)
- zh-CN (Simplified Chinese)
- fr-FR (French, France)

Note: The GET method does not return password information. When you create a new user account, the password is automatically set to be the same as the user name.

A separate PUT method lets you update the password of a specified user account. The password is sent in cleartext to avoid publicizing the encryption key for the web service to use. As a result, the method for changing the password must only be used on the server where CA Performance Center is installed to protect the password privacy.

User Account Product Privilege Settings

Use the CA Performance Center user interface to determine the product privilege of a specified user account. The *product privilege* is a type of permission set associated with a user account. The product privilege grants user access to features in selected data sources and does not apply to CA Performance Center functionality.

A user account has one of the following product privilege assignments:

NONE

Indicates that this user has no access to a specified data source.

ADMINISTRATOR

Indicates that this user has the administrator product privilege for the indicated data source and can perform administrative tasks.

POWER_USER

Indicates that this user has the power user product privilege for the indicated data source and can perform some administrative tasks associated with user accounts and dashboards.

USER

Indicates that this user has the user product privilege for the indicated data source and has no access to administrative features.

Users Web Service Example Syntax

Issue the following call to see the parameters and available operations for the users web service:

```
http://CA Performance Center Server IP
Address:8181/pc/center/rest/users/documentation
```

Available GET Methods

get groups owned by user

Retrieves a list of groups for the specified user. The groups that are returned are groups that are owned by the specified user, meaning that this user can modify or delete these groups:

```
http://CA Performance Center server IP address:8181/pc/
center/webservice/users/idName/idValue/groupsOwnedByUser
```

get groups

Retrieves a list of groups to which the specified user has view access. The groups that are returned are groups that are within the permission set of the specified user. The user cannot modify or delete these groups:

`http://CA Performance Center server IP address:8181/pc/center/webservice/users/idName/idValue/groups`

get id names

Retrieves a list of identifiers that can be used to identify users in other web service methods.

`http://CA Performance Center Server IP Address:8181/pc/center/webservice/users/idNames`

idValue

Is the value for the identifying category. For example, if *idName* is userID, provide the user ID. If *idName* is userName, provide the user name.

get authentication types

Returns a list of identifiers that can be used to assign authentication types to users.

`http://CA Performance Center Server IP Address:8181/pc/center/webservice/users/authenticationTypes`

Available PUT Methods

update role

Updates the role assignment of a specified user account.

`http://CA Performance Center Server IP Address:8181/pc/center/webservice/users/idName/idValue/role/roleIdName/roleIdValue`

update time zone

Updates the time zone of a specified user account.

`http://CA Performance Center Server IP Address:8181/pc/center/webservice/users/idName/idValue/timeZone/newTimeZone`

set groups

Updates the permission groups that have been granted to a specified user account.

`http://CA Performance Center Server IP Address:8181/pc/center/webservice/users/idName/idValue/groups`

set administered groups

Updates the groups of a specified user account.

`http://CA Performance Center Server IP Address:8181/pc /center/webservice/users/idName/idValue/administeredGroup`

To add administered groups to a user account, supply XML in the following format (for example):

```
<groups>
  <group ID="5245"/>
  <group ID="5246"/>
  <group ID="5247"/>
...
 </groups>
```

update product privilege per datasource

Updates the product privilege of a specified user account to enable access to the user interface of a specific datasource.

`http://CA Performance Center Server IP Address:8181/pc /center/webservice/users/idName/idValue/ds/dsId/ productPrivilege/newProductPrivilege`

dsId

The data source identifier. For a list of available data source identifiers, see Basic Datasource Parameters (see page 16).

For more information about product privileges, see User Account Product Privilege Settings (see page 41).

Available POST Methods

create

Creates a new user account. The new user is associated with the tenant for the logged-in user. The parameters include a role assignment.

`http://CA Performance Center Server IP Address:8181/pc /center/webservice/users/role/roleIdName/roleIdValue`

For more information, see Create a User Account (see page 44).

create in tenant

Creates a new user account in the specified tenant, with the specified role assignment.

`http://CA Performance Center Server IP Address:8181/pc /center/webservice/users/tenant/tenantIdName/tenantIdValue /role/roleIdName/roleIdValue`

Create a User Account

Create a custom user account using the users web service.

Every user account is automatically associated with a tenant. If you are deploying multi-tenancy, the new user is assigned to the tenant of the authenticated user account that was used to make the REST service call. If you are not deploying multi-tenancy, this association is transparent to you; new user accounts are associated with the Default Tenant.

Follow these steps:

1. Set up a REST client with a connection to the CA Performance Center server.
2. Enter a URL for the CA Performance Center RESTful web services API in the REST client. Use the following format:

   `http://`*`CA Performance Center Server IP Address`*`:8181/pc/center/webservice/users/role/`*`roleIdName`*`/`*`roleIdValue`*`/`

   For *roleIdName*, use values that are specified in http://*CA Performance Center Server IP Address*:8181/pc/center/webservice/roles/idNames.

   Examples are 'roleName' and 'roleId'.

   For *roleIdValue*, this value depends on the *roleIdName* that you selected. For example, if 'roleName' is used, substitute a valid role name for *roleIdValue*.

   This role must be available within the tenant.
3. Select POST for "HTTP" Method.
4. Provide a valid Username and Password for a user account that has host or tenant administrator access to CA Performance Center.
5. Select 'application/xml' as the 'Body Content-type' in the Body settings.
6. Add the following parameters within the "Body" text section:

```
<user>
   <name>{UserName}</name>
   <description>{UserDescription}</description>
   <enabled>{UserEnabled}</enabled>
   <removable>{UserRemovable}</removable>
   <timezone>{UserTimeZone}</timezone>
   <culture>{UserCulture}</culture>
</user>
```

7. Replace any values with the values that you want to use for the new user account.

   For example, supply the following parameters:

```
<user>
   <name>Jane Doe</name>
   <description>User associated with the John Doe Corporation
tenant.</description>
   <enabled>true</enabled>
   <removable>true</removable>
   <timezone>CST6CDT</timezone>
   <culture>en-US</culture>
</user>
```

   For more information about user parameters, see Basic User Account Parameters (see page 39).

8. Run the method.

9. Repeat the preceding steps until you have created as many users as you require.

Tenants Web Service

The API provides the tenants web service to let you get a current list of tenant definitions, create new tenant definitions, and modify those definitions.

The basic tenant definition contains a few parameters to identify the tenant. All of the infrastructure—devices, networks, servers—and all monitoring parameters for a customer's monitored systems must be associated with the tenant definition. Each tenant must contain at least one IP domain, plus as many of the following definitions as required to manage the associated enterprise infrastructure and applications:

- User accounts
- Roles
- Custom and system groups
- Custom reports
- Custom menus

To associate definitions and monitoring parameters with an existing tenant definition, log in as the tenant administrator and use the required web services to create the required definitions. The definitions are then associated with the tenant definition and available to users logged in with this tenant's user accounts.

A use case document is available on the CA Performance Center Documentation Bookshelf to help you use the tenants and users web services.

Tenants Web Service Parameters

Issue the following call to see the available parameters and operations for the tenants web service:

```
http://CA Performance Center Server IP Address:8181/pc
/center/rest/tenants/documentation
```

Parameters

tenantDescription

(Optional) Describes the tenant.

idName

Is a name for the tenant.

status

Is the status of this tenant. Select one of the following values:

- Activated: Enables tenant user accounts for use.
- Disabled: Prevents any actions by user accounts that are associated with this tenant.

removable

States whether the item can be deleted (removed from the database).

Values: true or false.

theme

Specifies the format—the theme that controls the appearance of the page in the browser window—to use for this tenant. All operators whose user account is associated with this tenant see this same theme. Two themes are available: CA-Blue and CA-Gray.

Default: CA-Blue.

defaultCulture

Specifies a language (locale). Supply a language identifier from the following list:

- en-US (English, United States)
- ja-JP (Japanese)
- zh-CN (Simplified Chinese)
- fr-FR (French, France)

accountId

Identifies this tenant; usually corresponds to the MSP account number. If a value is supplied as input, it must be unique across all defined tenants.

tenantID

Is an internal (database) identifier for a tenant definition.

Tenants Web Service Example Syntax

The tenants web service lets you get a current list of tenant definitions, create new tenant definitions, and modify those definitions by changing their parameters.

A use case document is available on the CA Performance Center Documentation Bookshelf to help you use the tenants web service.

Operations

The following basic operations are supported by the tenants web service:

GET

Returns a list of tenant definitions sorted by name. Available on the /tenantID endpoint. Use the following syntax:

```
http://CA Performance Center Server IP Address:8181/pc
/center/webservice/tenants/
```

POST

Creates a custom tenant. Use the following syntax:

```
http://CA Performance Center Server IP Address:8181/pc
/center/webservice/tenants/
```

For more information, see Create a Tenant (see page 48).

PUT

Updates an existing tenant definition. Use the following syntax:

```
http://CA Performance Center Server IP Address:8181/pc
/center/webservice/tenants/
```

Create a Tenant

Use any REST client to create and configure a tenant using the tenants web service.

Follow these steps:

1. Set up a REST client with a connection to the CA Performance Center server.
2. Enter a URL for the CA Performance Center RESTful web services API in the REST client. Use the following format:

   ```
   http://CA Performance Center Server IP Address:8181/pc
   /center/webservice/tenants/
   ```

3. Select POST for "HTTP" Method.
4. Provide a valid Username and Password for a user account that has global administrator access to CA Performance Center.
5. Select 'application/xml' as the 'Body Content-type' in the Body settings.
6. Add the following XML within the "Body" text section:

   ```
   <tenant>
     <tenantName>Name of tenant</tenantName>
     <tenantDesc>Description of the tenant</tenantDesc>
     <accountIdentifier>unique string for this
   tenant</accountIdentifier>
     <status>{activated or disabled}</status>
     <removable>{true or false}</removable>
     <theme>{CA-Blue or CA-Gray}</theme>
     <defaultCulture>culture</defaultCulture>
   </tenant>
   ```

7. Replace any values with the values that you want to use for the new tenant.

   For example, supply the following parameters:

   ```
   <tenant>
     <tenantName>John Doe</tenantName>
     <tenantDesc>John Doe Corporation tenant</tenantDesc>
     <accountIdentifier>JD1234</accountIdentifier>
     <status>Enabled</status>
     <removable>false</removable>
     <theme>CA-Blue</theme>
     <defaultCulture>en-US</defaultCulture>
   </tenant>
   ```

   For more information about tenant parameters, see Tenants Service Example Syntax (see page 46).

8. Run the method.
9. Repeat the preceding steps until you have created as many tenants as you require.

Chapter 3: Supporting Web Services

The API also provides web services that provide supporting information to the administrative web services. Use these web services to get lists of configuration items or identifying data.

Consoleinfo Web Service

The API provides a consoleinfo web service to let you retrieve information about CA Performance Center console configuration. The information that you retrieve can be passed in to other web service methods, such as time zone information that is required to create business hour definitions.

Consoleinfo Web Service Example Syntax

Issue the following call to see the parameters for the consoleinfo web service:

```
http://CA Performance Center Server IP
Address:8181/pc/center/rest/consoleinfo/documentation
```

Available GET Methods

get all time zones

Gets a list of all of the time zones that are available for use in business hour definitions.

```
http://CA Performance Center Server IP Address:8181/pc
/center/webservice/consoleinfo/allTimezones/cultureID
```

cultureId

Specifies a language (locale). Supply a language identifier from the following list:

- en-US (English, United States)
- ja-JP (Japanese)
- zh-CN (Simplified Chinese)

  fr-FR (French, France)

get installed language packs

Gets a list of language packs that are installed on the server.

```
http://CA Performance Center Server IP Address:8181/pc
/center/webservice/consoleinfo/languagepacks/cultureID
```

get time zones assigned to sites

Retrieves a list of all of the time zones that are assigned to site groups to which the logged-in user has access.

`http://CA Performance Center Server IP Address:8181/pc/center/webservice/consoleinfo/timezonesAssignedToSites`

Event Web Service

The event web service lets you retrieve a list of events that have been raised in your environment.

Perform a GET to the following URL to see a list of events for the specified managed item:

`http://CA Performance Center Server IP Address:8181/pc/eventId/item/itemId`

Event ID

Is an internally assigned value to identify the event.

For more information, see Devices Web Service (see page 19).

Finding Out More

The full list of all available web services for CA Performance Center is available at the following URL:

`http://CA Performance Center Server IP Address:8181/pc/center`

From this page, you can also access the WADL of each web service.

Most web services provide their own documentation, including lists and descriptions of the available parameters and operations. The documentation is accessible in HTML format from the API launch page:

`http://CA Performance Center Server IP Address:8181/pc/center/rest`

where the Server IP address is the same as the CA Performance Center server.

The CA Performance Center Documentation Bookshelf provides additional documentation in the form of task-based use cases. Check the Administration and Use Cases categories for relevant documents.

Glossary

dashboards

Dashboards are dynamic report-building pages within the CA Performance Center user interface. They appear as menu items that are accessible from the Dashboards tab. Each dashboard is a collection of views that present data from registered data sources on a single web page. The layout, views, time interval, and group context of each dashboard can be customized.

data sources

Data sources are the supported products that provide performance and configuration data to CA Performance Center. Data source products that monitor, collect, and aggregate data can often function independently. However, once they are registered to an instance of CA Performance Center, they are known as data sources.

domain

IP domains are logical groupings that identify data from different devices and networks. Monitoring by domain means that IP addresses with associated interfaces or applications that belong to separate customer networks are monitored separately. When combined with appropriate permissions, IP domains are monitored from a single console, but users view data only for the domains that they monitor.

group

A *group* is a filter definition that functions as a container for managed items. Groups let you logically organize managed items in a tree structure, with each group containing subgroups or managed items. The structure is propagated to the data sources, where it enables drilldown from top-level groups into data from an increasingly narrow but related context.

menus

Menus are segments of the Dashboards tab that are used to organize dashboards by their content. By default, Administrators and Designers can customize menus and assign them to user account roles.

product privilege

The *product privilege* is a type of permission set associated with a user account. The product privilege grants user access to features in selected data sources and does not apply to CA Performance Center functionality.

REST

REST, or Representational State Transfer, refers to a method of structuring software for the World Wide Web or other applications that conform to the requirements of HTTP in client-server networks. The REST model lets you access a set of resources by means of a fixed set of operations. This model takes advantage of widely deployed HTTP features that are supported by common hardware, such as gateway devices.

role

The *role* is a parameter assigned to a user account that controls user access to product features and dashboard pages. Based on user job functions, the role grants administrative access to product configuration using *role rights*. Roles let users access data and product features that they require to perform their duties and restrict access to features that they do not require.

Single Sign-On

Single Sign-On is the term used to describe the authentication scheme used by CA data sources that CA Performance Center supports. The Single Sign-On component provides the login page that supports user authentication in CA Performance Center and in the data source products. Once users are authenticated to CA Performance Center, they can navigate among CA Performance Center and registered data sources without signing in a second time.

site groups

Site groups are custom groups that are based on physical locations, such as a city, region, office, or campus. Typically, site groups contain items and subgroups of items that are grouped by location. Adding site groups to other custom groups in your tree structure allows you to build geographically and logically organized reports. Site groups enable business-hour filtering of dashboard views.

SNMP profiles

SNMP profiles are definitions that contain the information necessary to enable secure queries of device MIBs using SNMP.

tenant

A *tenant* represents a customer environment that a managed service provider administers. Each tenant environment is independent and effectively functions as a separate instance of CA Performance Center. Each instance can contain multiple users and roles that are not shared among tenants.

Index

No index entries found.