Cisco Intercloud Services

Cisco Intercloud Services APIs

Published 01/05/2016

Copyright Notice

Document History

Topic	Date of Change	Description
Document History page	07/05/2015	Initially created.
Cisco Public	07/14/2015	Changed "Cisco Confidential" to "Cisco Public"
New document Title	07/05/2015	• Changed the document title. • Combined ordering fulfillment and remediation APIs to a user-friendly format. • Updated API parameters and examples.
Manage Tenants API	07/06/2015	Added new parameters.
Manage Users API	07/06/2015	Added new parameters.
Manage Projects API	07/06/2015	Added new parameters.
Manage Quota API	08/14/2015	New APIs.
Manage Bulk Operations	08/31/2015	New APIs.
Create a Project-V2	08/31/2015	Version 2 - new API.
Get Project Users-V2	08/31/2015	Version 2 - new API.
Get a Project by Name-V2	08/31/2015	Version 2 - new API.
Get a Project by a Project ID-V2	08/31/2015	Version 2 - new API.
Get Project Details for a Project User-V2	08/31/2015	Version 2 - new API.
Get Projects of a Specific Tenant of a Partner-V2	08/31/2015	Version 2 - new API.
Create a Tenant-V2	08/31/2015	Version 2 - new API.
Get a Tenant-V2	08/31/2015	Version 2 - new API.
Associate User to Tenant-V2	08/31/2015	Version 2 - new API.
Disassociate a User from a Tenant-V2	08/31/2015	Version 2 - new API.

Topic	Date of Change	Description
Get Active Tenants for a Specific Partner-V2	08/31/2015	Version 2 - new API.
Create a User-V2	08/31/2015	Version 2 - new API.
Get a User-V2	08/31/2015	Version 2 - new API.
Get User V2 For Input UIDs	08/31/2015	Version 2 - new API.
Exchange an Access Token for a Keystone Token	08/31/2015	Version 2 - new API.
Get Project Info by Req ID-V2	09/25/2015	Version 2 - new API.
Get Target Provider Info	08/31/2015	Version 2 - new API.
Manage User Roles	09/03/2015	New APIs.
Generate the API Key	09/24/2015	New procedure. Replaced Regen User API Key.
Regen User API Key	09/24/2015	Deprecated.
Get User V2 For Input UID	10/8/2015	Updated response example.
Get a User-V2	10/14/2015	Updated description.
Remove a Project	10/14/2015	Updated description and added a new parameter option, "verify."
Create a User	10/14/2015	Updated description.
Create a User-V2	10/14/2015	Added a notification flag.
Get Target Provider Info	10/22/2015	Updated response example.
Cisco Intercloud Services	01/04/2016	Added a section for Unsupported Characters.
Create a Tenant	01/04/2016	Added a note for unsupported characters.
Create a Tenant-V2	01/04/2016	Added a note for unsupported characters.

Contents

Copyright Notice 2
Document History 3
Introduction **9**
Audience 9
Conventions 9
Cisco Intercloud Services **10**
OpenStack REST APIs 11
Cisco Intercloud Services APIs 11
Unsupported Characters 12
REST Clients 12
Request URL 12
Authentication 12
Header 12
HTTP Status Codes 13
Generate an API Key **14**
Exchange Access Token for Keystone Token **16**
On-boarding Process **18**
Error Handling 19
Get Target Provider Info **21**
Managing Tenants **22**
Create a Tenant 24
Create a Tenant–v2 26
Get Active Tenants for a Specific Partner–v2 28
Get a Tenant–v2 30
Get a Tenant 31
Remove a Tenant 32
Resume a Tenant 34
Suspend a Tenant 36
Associate a User to a Tenant 38
Disassociate a User from a Tenant 40

Associate a User to a Tenant–v2 42
Disassociate a User from a Tenant–v2 43
Managing Users **46**
Create a User 47
Create a User–v2 49
Delete a User 52
Get a User 54
Get a User–v2 55
Get User V2 For Input UIDs 57
Get Project Details for a Project User 59
Get Project Details for a Project User–v2 59
Update a User 61
Managing User Roles **63**
Create User Roles 64
Delete User Role Association 66
Get User Accounts 67
Get Users for a Specific Account 68
Get Users for a Specific Account and Role 69
Get Roles for a Specific Account and User 70
Get Users for a Specific Tenant 71
Get Users for a Specific Tenant and Role 72
Get Roles for a Specific Tenant and User 73
Get User Roles 74
Update User Role Association 75
Managing Projects **77**
Create a Project 78
Create a Project-v2 80
Get Project Users–v2 81
Get Project Info by Req ID 83
Get Project Info by Req ID-v2 84
Remove a Project 85

Resume a Project 87
Suspend a Project 89
Associate a User to Project 91
Disassociate a User from Project 93
Managing Quota **95**
Account Quota 96
Create an Account Quota 97
Delete an Account Quota 99
Get an Account Quota Based on a Region 101
Get an Account Quota Based on Service 102
Get Account Quota Based on Service and Region 104
Get Quota for All Accounts 105
Get Quota for a Specific Account 108
Set Maximum Account Quota Pool 110
Managing Tenant Quota 113
Create Tenant Quota 114
Delete Tenant Quota 117
Get a Tenant Quota Based on a Region 120
Get a Tenant Quota Based on Service 121
Get a Tenant Quota Based on Service and Region 122
Get All Tenants Quota 123
Set Maximum Tenant Quota Pool 124
Update a Tenant Quota 127
Check Requisition Status 130
Managing Bulk Operations **131**
Associate Many Users to Many Projects 132
Disassociate Many Users from Many Projects 133
Disassociate Many Users from One Tenant 134
Get Bulk User Details with Filters 135
Get Bulk User-Tenant Disassociation Status 137
Generate Reports **139**

Generate a Report for All Tenants 140
Generate a Report for a Specific Tenant 143
Generate a Report for a Specific Tenant Project 146
Generate a Report for a Specific Project Instance 149
Get Service Requests Status 152

Introduction

This document lists API methods for on-boarding and managing tenants and projects of the provider.

Audience

The intended audience is the federated partner operations team who is responsible for operations of the Infrastructure as a Service (IaaS) based on Cisco Intercloud Services. It is assumed that these operators have basic understanding of the following:

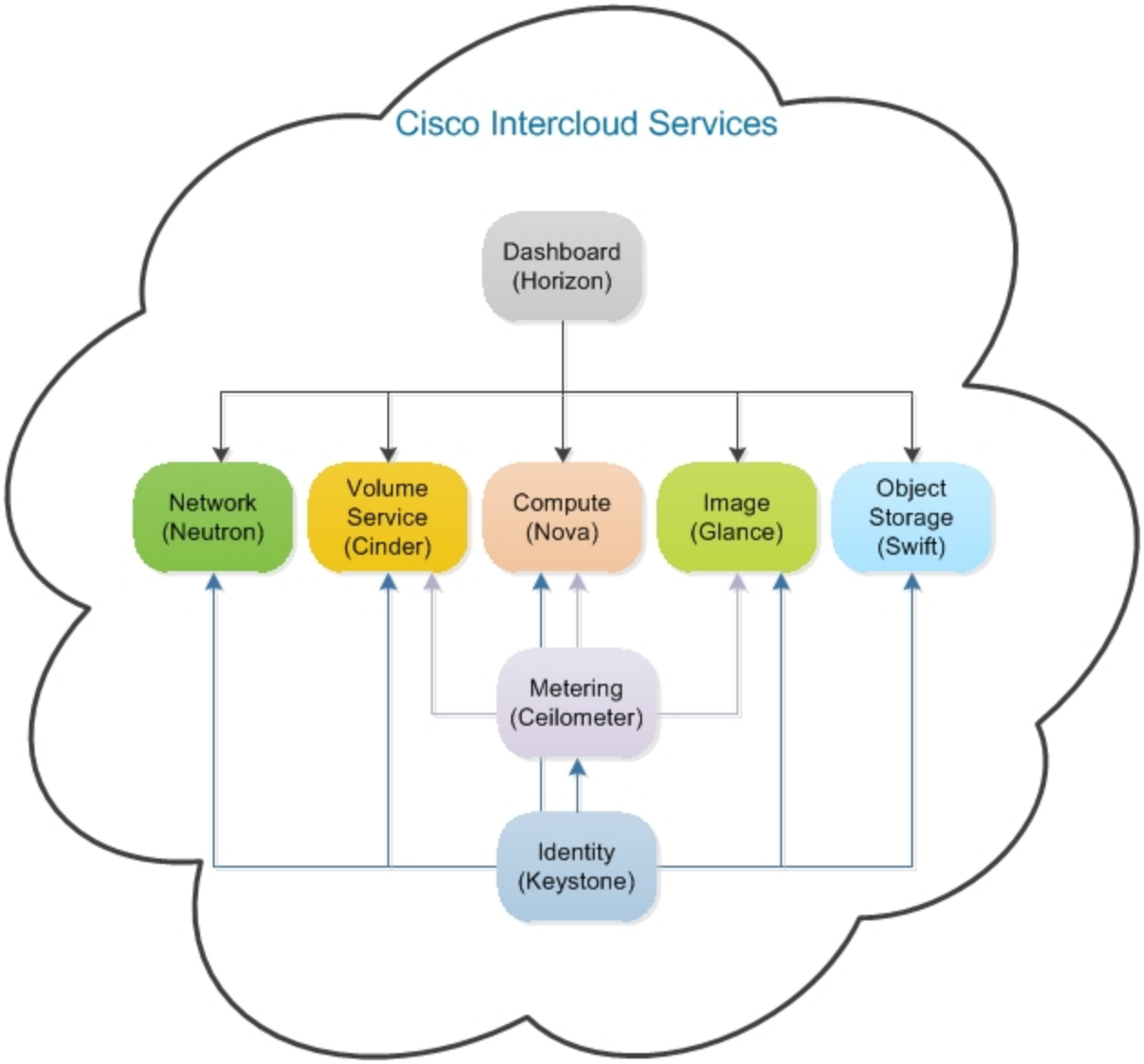
- RESTful web services
- HTTP 2.0
- JSON format

Conventions

Item	Description
bold	Menu, command.
mono-space font	Code, typed data.
	**Note**. Contains information that might be useful. Ignoring a note has no negative consequences.
	**Tip**. Includes information such as helpful hints or a shortcut that might help you complete a task.
	**Important**. Includes information that might be easily overlooked and might cause unnecessary frustration. For example, configuration changes that only apply to the current session, or services that need restarting before an update will apply.
	**Warning**. Contains information that must not be ignored. Ignoring recommendations in Warnings may result in data loss or other catastrophic issues.

Cisco Intercloud Services

Cisco Intercloud Services (CIS) is an Infrastructure-as-a-service (IaaS) model, which provides virtual machines and other resources—servers, storage and networking—on a on-demand and as-need basis. The following diagram shows the Cisco Intercloud Services architecture that is based on OpenStack and configured to be integrated with the Cisco network and authentication services.



Component	Description
Dashboard (Horizon)	A web-based interface that provides a graphical user interface for provisioning, managing, and monitoring OpenStack operations.
Network (Neutron)	Provides networking connectivity between the interfaces of other OpenStack services.
Volume (Cinder)	Provides persistent block-level storage for the instances running in Nova.
Compute (Nova)	Manages virtual machines running on nodes, and interacts with Horizon, Glance, and Keystone.
Object Storage (Swift)	Provides redundant storage system, allowing tenants to store and retrieve objects and files.
Metering (Ceilometer)	Collects metering and event data, and monitors resources for performance, such as memory, disk, and CPU.
Identity (Keystone)	Provides authentication and authorization for other OpenStack services.

OpenStack REST APIs

The OpenStack APIs are RESTful APIs that use the HTTP protocol. You need a Keystone token for calling OpenStack APIs.

Cisco Intercloud Services APIs

You use the Cisco Intercloud Services subset of OpenStack APIs to manage projects and users, and monitor tenant operations. You use the CIS APIs for CRUD (create, read, update and delete) operations. The primary methods are GET, POST, PUT, and DELETE.

CIS APIs use standard HTTP methods and authentication, and use JSON encoding for request and response data. CIS APIs provide the following services:

- Manage Tenants
- Manage Users
- Manage Projects
- Manage Quota
- Generate Reports

- Exchange an Access Token for a Keystone Token
- Manage Bulk Operations
- Manage User Roles

Unsupported Characters

CIS APIs do not support the hash character (#) and the forward slash (/). Do not include these characters as a part of an identifier, such as tenant ID and project ID.

REST Clients

You can use REST or HTTP clients to provide request and receive responses. You can also use cURL, a command line tool that lets you make HTTP calls and receive responses.

Request URL

```
http://<server-ip>:<port>/<URI>
```

Authentication

CIS APIs use HTTP basic authentication. To authenticate, provide the user name and the API key that you generate from Horizon.

Header

```
Content-Type: application/json
Accept: application/json
```

The following example shows a request header using cURL:

```
-X POST \
-H ‘Authorization: Basic <<Base64 encoded (username:password)>>' \
-H 'Content-type: application/json' \
```

```
-H 'Accept: application/json' \
-d '<requestPayload>'
```

A backslash (\) is used as an escape character that allows continuation of the command across multiple lines. However, do not include backsplash in the JSON request body within the cURL command.

HTTP Status Codes

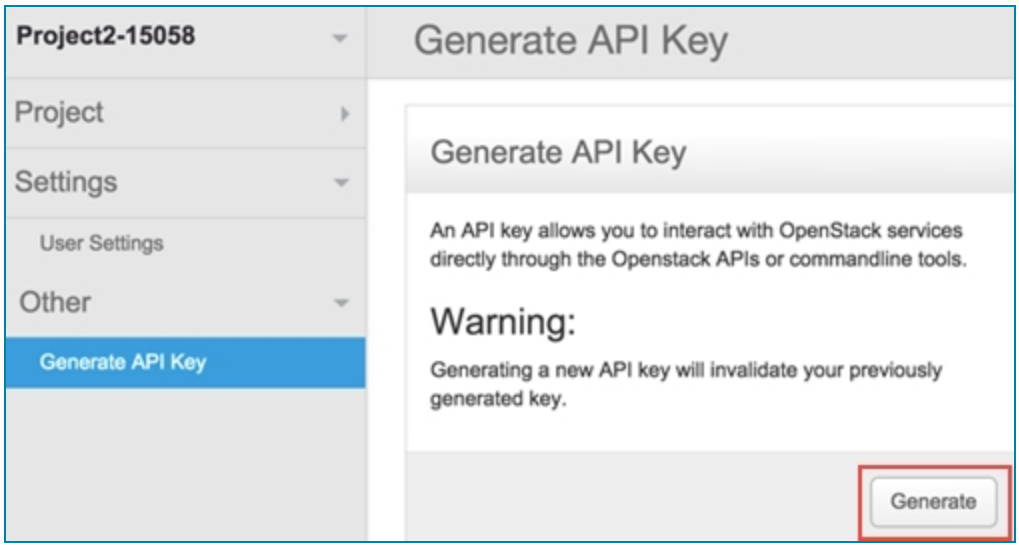
Status Codes	Description
201 OK	The operation is successful.
400 Bad Request	The request is missing one or more elements, or the values of some elements are invalid.
401 Unauthorized	You are not authorized to complete this operation because the request is submitted with an invalid authentication token.
403 Forbidden	The request is valid, but the server is refusing to respond because you do not have permission to access the requested resource.
404 Not Found	The requested resource is not found.
500 Internal Server Error	An unexpected condition is encountered.
503 Service Unavailable	Service is currently not available.

Generate an API Key

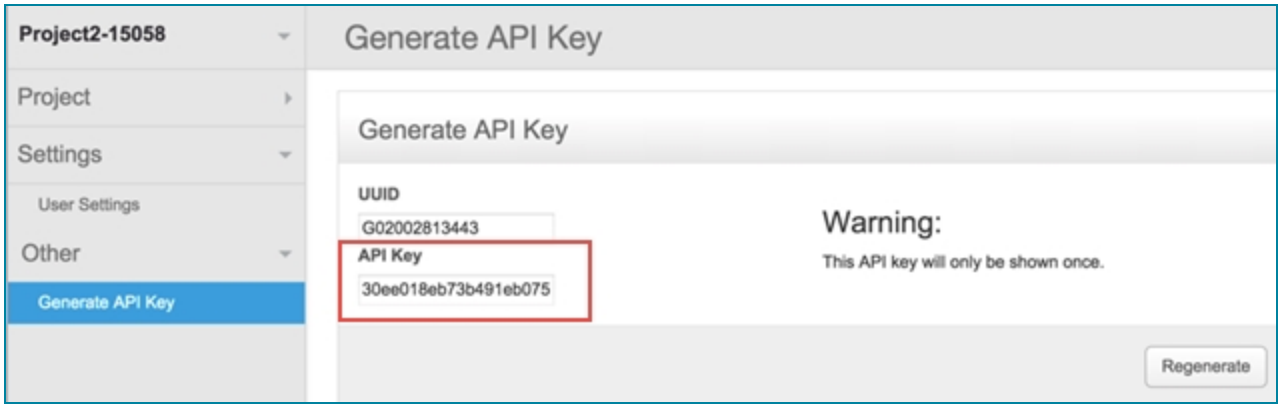
You need an API key to make OpenStack API calls.

To generate an API key:

1. Log into Horizon with your user name.
2. Click **Settings -->Other -->Generate API Key**.



3. Click **Generate**.



4. Copy the API key.
5. Call the OpenStack API using the API key to request a Keystone token that you will

use for subsequent API calls, for example,

```
curl -X POST -d '{"auth":{"passwordCredentials":
{"username":"cecuser","password":"30ee018eb73b491eb075"}}}' -H "Content-type:
application/json" http://10.207.235.25:5000/v2.0/tokens
```

Clicking **Regenerate** generates a new API key and invalidates the previously generated key.

Exchange Access Token for Keystone Token

Method	URI	Description
POST	/services/v2/keystone/tokengen	Exchange a valid OAuth2 access token for a keystone token. **Note**: This works only for access tokens issued by Cisco IDP (Identity Provider) at this time.

Request Header

```
Authorization: Bearer <<access token>>
Accept: application/json
Content-Type: application/json
```

Request Example

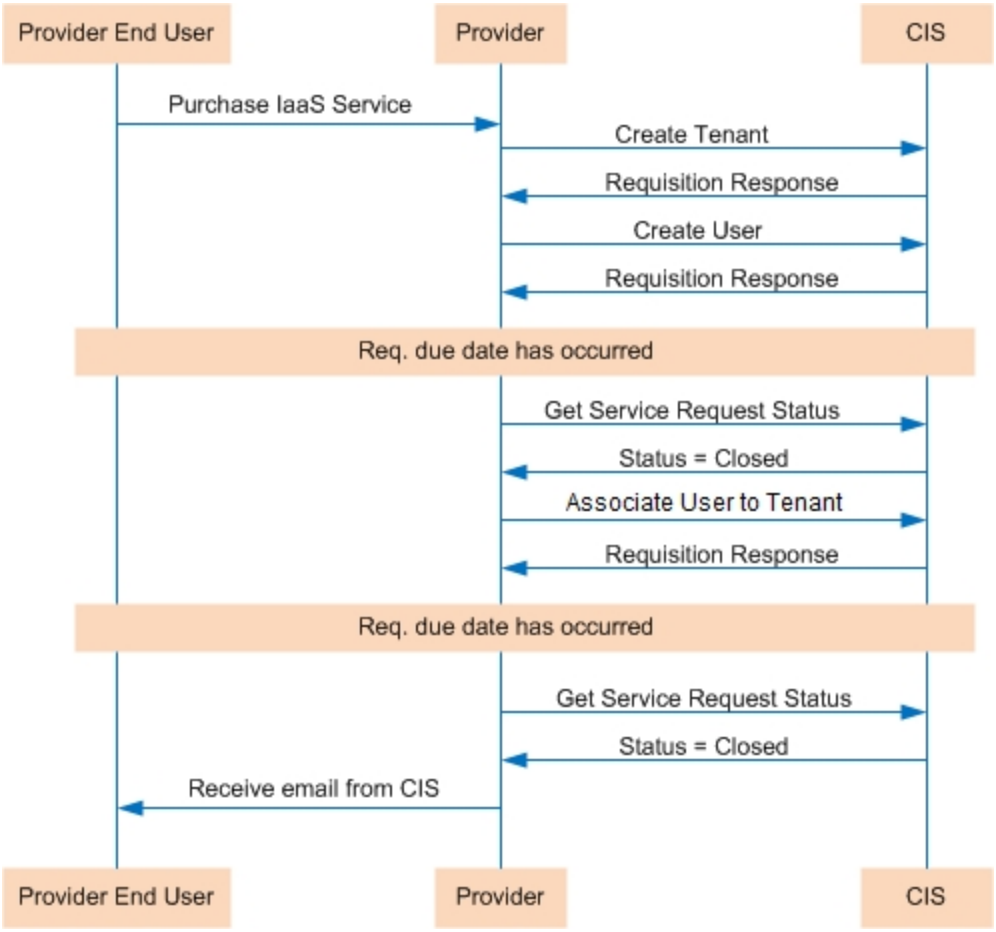
```
{
"appId":"CLIQR",
"serviceProvider":"CISCO"
}
```

Response Example


```
{"access":
             {"token":
                 {"issued_at": "2015-03-25T18:35:15.210659",
                 "expires": "2015-03-25T19:35:15Z",
                 "id": "322e09216c9b4904a21728e82c6353e2"},
             "serviceCatalog": [],
             "user":
                    {"username": "bhagyash",
                    "roles_links": [],
                    "id": "cd088ff8516c4e36aa7f418a5686a771",
                    "roles": [],
                    "name": "bhagyash"},
             "metadata":
                    {"is_admin": "0",
                    "roles": []
                    }
             }
}
```

On-boarding Process

The provider's end-user interacts with the provider to purchase IaaS services. This triggers the provider to issue a number of API calls to complete this process. The following diagram shows a typical onboarding process.



The provider issues the Create Tenant API call that contains the identifier of the tenant.

The Onboarding management interface synchronously returns a response to the Requisition.

After the tenant is created, the provider issues the Create User API. The first user being onboarded is given the role of Administrator, which is required to add new projects and users by way of the CIS Horizon.

The Onboarding management interface synchronously returns a response to the Requisition. This response includes a status of Ongoing, an ID for the Requisition and an estimate on when the processing of the Requisition should complete.

To initiate this process, CIS console provides the Administrator with a form to send an email to the prospective user. The email provides a link to the provider's portal to register themselves as a user of the platform.

After receiving confirmation of the completed Requisition, the provider's portal informs the end-user to check their email, including their Spam or Junk folder, for an email from the Cisco Intercloud.

As this point, the tenant's user can begin to create projects (Create Project) and consume CIS IaaS resources using Horizon and OpenStack APIs.

Error Handling

There are two types of errors that can occur when submitting API requests to the CIS Onboarding Management Interface: synchronous and asynchronous.

- Synchronous errors are fatal errors, where the calling application is immediately notified of the error in response to the API submittal. In the case of a synchronous error, the request does not start the fulfillment process.

Upon receipt of the synchronous error, the calling application should take action based on the error message, and resubmit the request if appropriate. The possible synchronous errors are noted for each applicable API call and the suggested action to be taken by the calling application.

- Asynchronous errors may or may not be fatal. These occur when a request has passed error conditions that would generate a synchronous error, but fail in the subsequent fulfillment of the request. If an asynchronous error is encountered, the request will not be completed and will show as "Ongoing" in response to polls until the error condition is resolved.

The recommendation for the calling application is to raise an alert when 2x the duration indicated in the request response, that is, 2x (dueDate – startedDate), has been reached. Based on this alert, submit Cisco a support case to investigate the root cause of the issue. Once the root case of the error condition has been identified and corrected, the request continues its execution, and returns “Closed” to the calling application. Therefore, requests that have asynchronous errors typically do not need to be resubmitted.

Get Target Provider Info

Method	URI	Description
GET	/services/v2/targetProviders	Returns information of all CIS active target providers.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Response Example


```
[
  {
    "KeystoneEndpoint": "https://us-rdu-2.cisco.com:5000/v2.0",
    "CustomerType": "Internal,Reseller,Marketplace",
    "Description": "",
    "DisplayOrder": "1",
    "ExternalAllowed": "Yes",
    "DCLocation": "North Carolina, United States",
    "DisplayName": "US-RDU-2",
    "City": "Research Triangle Park",
    "PostalCode": "27709-4987",
    "State": "North Carolina",
    "TrialAllowed": "Yes",
    "Address2": null,
    "Address1": "7200-10 Kit Creek Road",
    "HorizonURL": "https://us-rdu-2.cisco.com/horizon",
    "CountryCode": "020",
    "ProviderTarget": "US-RDU-2"
  }
]
```

Managing Tenants

You must create the tenant before you create projects for that tenant.

Service Name	Type	URI
Create a Tenant	POST	/services/tenant
Get a Tenant	GET	/services/tenant/<<ccs_tenant>>
Remove a Tenant	DELETE	/services/tenant/<<ccs_tenant>>?force=true/false
Resume a Tenant	PUT	/services/tenant/<<ccs_tenant>>/resume
Suspend a Tenant	PUT	/services/tenant/<<ccs_tenant>>/suspend
Associate a User to a Tenant	POST	/services/user/tenant
Disassociate a User from a Tenant	PUT	/services/user/tenant
Version 2		
Create a Tenant-v2	POST	/services/v2/tenant
Get a Tenant-v2	GET	/services/v2/tenant/<<ccs_tenant>>
Get Active Tenants for a Specific Partner-v2	GET	/services/v2/serviceProvider/<<SP>>/tenants
Get Projects of a Specific Tenant of a Partner-v2	GET	/services/v2/serviceProvider/<<SP>>/tenant/<<tenantId>>/projects?status=<<active|any>>
Associate User	POST	/services/v2/user/tenant?action=<<action>>

Service Name	Type	URI
to Tenant-v2		
Disassociate a User from a Tenant-v2	PUT	/services/v2/user/tenant?action=<<action>>

Create a Tenant

Method	URI	Description
POST	/services/tenant	Create a tenant.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Example

```
{
 "ccs_tenant": "<TenantID>",
 "description": "<Description>"
}
```

Request Parameters

Parameter	Description
ccs_tenant	Required. The tenant identifier cannot exceed 32 characters and it has to be unique. Do not include the hash character (#) or forward slash (/) as a part of an identifier.
description	Optional. A notation of the tenant.

Response Example


```
{
    "RequisitionSubmit": {
        "id": 23,
        "customer": "API User",
        "initiator": "API User",
        "dueDateRaw": 1399047699577,
        "dueDate": "05/02/2014 9:21 AM",
        "startedDateRaw": 1399047303153,
        "startedDate": "05/02/2014 9:15 AM",
        "status": "Ongoing”
    }
}
```


Response Parameters

- The date/time format and time zone are based on the preferences set in the login user's personal profile.
- The due date returned in the API response is available only when the asynchronous submission global setting is enabled.
- Formatted dates returned in the API responses are based on the API user's timezone. Raw dates returned are in UTC.

Create a Tenant–v2

Method	URI	Description
POST	/services/v2/tenant	On board a tenant.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Example

```
{
 "ccs_tenant": "<TenantID>",
 "description": "<Description>",
 "partner_uid":"<PartnerID>",
 "customer_uid":"<CustomerID>",
 "billing_uid":"<BillingID>",
 "customer_type":"Direct",
 "payment_type":"PO",
 "subscription_id":"1234",
 "has_billing":"False",
 "service_level":"Standard",
 "tenant_type":"Trial"
 }
```

Request Parameters

Do not include the hash character (#) or forward slash (/) as a part of an identifier.

Parameter	Description
ccs_tenant	Required. The tenant identifier cannot exceed 32 characters and it has to be unique.
description	Optional. A notation of the tenant.
partner_uid	Required. The ID of the partner.

Parameter	Description
customer_uid	Required. The ID of the customer.
billing_uid	Required. The ID of the billing.
customer_type	Optional.
payment_type	Optional.
subscription_id	Optional.
has_billing	Optional.
Service_level	Optional.
tenant_type	Optional. Options are: • Production (default) • Trial

Response Example


```
{
    "RequisitionSubmit": {
        "id": 23,
        "customer": "API User",
        "initiator": "API User",
        "dueDateRaw": 1399047699577,
        "dueDate": "05/02/2014 9:21 AM",
        "startedDateRaw": 1399047303153,
        "startedDate": "05/02/2014 9:15 AM",
        "status": "Ongoing"
    }
}
```


Response Parameters

- The date/time format and time zone are based on the preferences set in the login user's personal profile.
- The due date returned in the API response is available only when the asynchronous submission global setting is enabled.
- Formatted dates returned in the API responses are based on the API user's timezone. Raw dates returned are in UTC.

Get Active Tenants for a Specific Partner–v2

Method	URI	Description
GET	/services/v2/serviceProvider/<<SP>>/tenants	Get information for all active tenants for a specific partner.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Parameters

Parameter	Description
SP	Required. The identifier of the partner.

Response Example


```
[
  {
    "ccs_tenant": "<Tenant>",
    "description": "Tenant Desc",
    "partner_uid": "<PID>",
    "customer_uid": "<CID>",
    "billing_uid": "<BID>",
    "status": "Active",
    "customer_type": null,
    "payment_type": null,
    "subscription_id": null,
    "has_billing": null,
    "service_level": null,
    "project_count": "0",
    "user_count": "0",
    "tenant_type": "Production"
 },
  {
    "ccs_tenant": "<Tenant>",
    "description": "Tenant Desc",
    "partner_uid": "<PID>",
    "customer_uid": "<CID>",
    "billing_uid": "<BID>",
    "status": "Active" ,
    "customer_type": Direct,
    "payment_type": PO,
    "subscription_id": null,
    "has_billing": False,
    "service_level": Standard,
    "project_count": "0",
    "user_count": "0",
    "tenant_type": "Trial"
 }
]
```

Get a Tenant–v2

Method	URI	Description
GET	/services/v2/tenant/<<ccs_tenant>>	Return a tenant information.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Parameters

Parameter	Description
ccs_tenant	Required. The identifier of the tenant.

Response Example

This request returns the status of the tenant.


```
{
    "ccs_tenant": "CLTEST24",
    "description": "CLTEST24",
    "partner_uid": "Cisco",
    "customer_uid": "C123123",
    "billing_uid": "B123123",
    "customer_type": "Internal",
    "payment_type": "None",
    "subscription_id": "Standard",
    "has_billing": "FALSE",
    "service_level": "Standard",
    "status": "Active",
    "tenant_type": "Production"
}
```

Get a Tenant

Method	URI	Description
GET	/services/tenant/<<ccs_tenant>>	Return the tenant information.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Parameters

Parameter	Description
ccs_tenant	Required. The identifier of the tenant.

Response Example

This request returns the status of the tenant. Status includes Active, Suspended, or Inactive.

```
{
        "ccs_tenant": "<<TenantID>>",
        "description": "<<Description>>,
        "status": "<<Status>>"
    }
```

Remove a Tenant

Method	URI	Description
DELETE	/services/tenant/ <<ccs_tenant>>?force=true/false	Remove the tenant , and the behavior depends on the optional parameter, **force**.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Parameters

Parameter	Description
ccs_tenant	Required. The identifier of the tenant.
Force	Optional. • **True** indicates that the tenant is forcefully removed from CIS. If there are projects or users within the tenant, they will be automatically deleted, including any instances in the projects. • **False** (default) indicates that the tenant can only be removed from CIS if the tenant is empty. If projects or users exist within the tenant, an error will be returned from the CIS Console indicating assets still exist.

Response Example


```
{
    "RequisitionSubmit": {
        "id": 23,
        "customer": "API User",
        "initiator": "API User",
        "dueDateRaw": 1399047699577,
        "dueDate": "05/02/2014 9:21 AM",
        "startedDateRaw": 1399047303153,
        "startedDate": "05/02/2014 9:15 AM",
        "status": "Ongoing"
    }
}
```

Response Parameters

- The date/time format and time zone are based on the preferences set in the login user's personal profile.
- The due date returned in the API response is available only when the asynchronous submission global setting is enabled.
- Formatted dates returned in the API responses are based on the API user's timezone. Raw dates returned are in UTC.

Resume a Tenant

Method	URI	Description
PUT	/services/tenant/ <<ccs_tenant>>/resume	Resume a tenant from a suspended state. When resumed, access to CIS Horizon, OpenStack API, and remote access to instances are restored.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Parameters

Parameter	Description
ccs_tenant	Required. The identifier of the tenant.

Response Example


```
{
    "RequisitionSubmit": {
        "id": 23,
        "customer": "API User",
        "initiator": "API User",
        "dueDateRaw": 1399047699577,
        "dueDate": "05/02/2014 9:21 AM",
        "startedDateRaw": 1399047303153,
        "startedDate": "05/02/2014 9:15 AM",
        "status": "Ongoing"
    }
}
```


Response Parameters

- The date/time format and time zone are based on the preferences set in the login user's personal profile.

- The due date returned in the API response is available only when the asynchronous submission global setting is enabled.
- Formatted dates returned in the API responses are based on the API user's timezone. Raw dates returned are in UTC.

Suspend a Tenant

Method	URI	Description
PUT	/services/tenant/ <<ccs_tenant>>/suspend	• When a Suspend Tenant API is issued, access to all projects and instances within those projects is disabled. All users within the tenant are unable to log into the CISconsole or use the OpenStack APIs. • Remote access (such as SSH, Telnet, RDP, VNC) to all instances in all projects are blocked. • The state of all instances in all projects cannot be modified. • The tenant remains in the suspended state until an API call is received to either resume or delete the tenant. • Resource consumption remains to be collected, provided, and charged to the federation partner accordingly.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Parameters

Paramter	Description
ccs_tenant	Required. The identifier of the tenant.

Response Example


```
{
    "RequisitionSubmit": {
        "id": 23,
        "customer": "API User",
        "initiator": "API User",
        "dueDateRaw": 1399047699577,
        "dueDate": "05/02/2014 9:21 AM",
        "startedDateRaw": 1399047303153,
        "startedDate": "05/02/2014 9:15 AM",
        "status": "Ongoing”
    }
}
```


Response Parameters

- The date/time format and time zone are based on the preferences set in the login user's personal profile.
- The due date returned in the API response is available only when the asynchronous submission global setting is enabled.
- Formatted dates returned in the API responses are based on the API user's timezone. Raw dates returned are in UTC.

Associate a User to a Tenant

Method	URL	Description
POST	/services/v2/user/tenant?action=<<action>>	Associate a user to a tenant.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Example

```
{
  "ccs_tenant": "<TenantID>",
  "user_uid": "<<uid>>",
  "role": "User"
}
```

Request Parameters

Parameter	Description
ccs_tenant	Required. The identifier of the tenant you want the user to associate to.
user_uid	Required. The identifier of the user you want the tenant to associate to.

role	Optional. A role defines privileges and access within Horizon/Openstack. A user must have at least one role and can have more than one role. Currently, two roles exist, User and Project Administrator.
	• **User** role can be assigned to three projects within a provider portal. This role has Horizon access and is able to create instances.
	• **Project Administrator** role can be assigned to two projects within a provider portal. In addition to the user permission, this role can:
	  ◦ create and remove projects using the provider's portal console (preferred over CIS console)
	  ◦ add and edit role and remove users using the CIS console
	  ◦ modify project quotas using the CIS console

Parameter	Description
	Adding users involves redirection to the federations partner's portal in order to obtain the user_uid from the federation provider.

Response Example

```
{
    "RequisitionSubmit": {
        "id": 23,
        "customer": "API User",
        "initiator": "API User",
        "dueDateRaw": 1399047699577,
        "dueDate": "05/02/2014 9:21 AM",
        "startedDateRaw": 1399047303153,
        "startedDate": "05/02/2014 9:15 AM",
        "status": "Ongoing"
    }
}
```

Response Parameters

- The date/time format and time zone are based on the preferences set in the login user's personal profile.
- The due date returned in the API response is available only when the asynchronous submission global setting is enabled.
- Formatted dates returned in the API responses are based on the API user's timezone. Raw dates returned are in UTC.

Disassociate a User from a Tenant

Method	URI	Description
PUT	/services/user/tenant	Disassociate a user from a tenant.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Example

```
{
  "ccs_tenant": "<TenantID>",
  "user_uid": "<<uid>>",
  "role": "User"
}
```

Request Parameters

Parameter	Description
ccs_tenant	Required. The identifier of the tenant from which you want the user to disassociate.
user_uid	Required. The identifier of the user from which you want to disassociate.

role	Optional. A role defines privileges and access within Horizon/Openstack. A user must have at least one role and can have more than one role. Currently, two roles exist, User and Project Administrator.
	• **User** role can be assigned to three projects within a provider portal. This role has Horizon access and is able to create instances.
	• **Project Administrator** role can be assigned to two projects within a provider portal. In addition to the user permission, this role can:
	  ◦ create and remove projects using the provider's portal console (preferred over CIS console)
	  ◦ add and edit role and remove users using the CIS console
	  ◦ modify project quotas using the CIS console

Parameter	Description
	Adding users involves redirection to the federations partner's portal in order to obtain the user_uid from the federation provider.

Response Example

```
{
    "RequisitionSubmit": {
        "id": 23,
        "customer": "API User",
        "initiator": "API User",
        "dueDateRaw": 1399047699577,
        "dueDate": "05/02/2014 9:21 AM",
        "startedDateRaw": 1399047303153,
        "startedDate": "05/02/2014 9:15 AM",
        "status": "Ongoing"
    }
}
```

Response Parameters

- The date/time format and time zone are based on the preferences set in the login user's personal profile.
- The due date returned in the API response is available only when the asynchronous submission global setting is enabled.
- Formatted dates returned in the API responses are based on the API user's timezone. Raw dates returned are in UTC.

Associate a User to a Tenant–v2

Method	URL	Description
POST	/services/v2/user/tenant?action=<<action>>	This call either adds the user to a specific tenant or moves the user from an existing tenant to a new tenant based on the action you specified. The default action is **add**.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Example

```
{  "ccs_tenant" : "bp_tenant",
   "user_uid" : "testUser",
   "role" : "User"
}
```

Request Parameters

Parameter	Description
ccs_tenant	Required. The identifier of the tenant you want the user to associate to.
user_uid	Required. The identifier of the user you want the tenant to associate to.
role	Optional. A role defines privileges and access within Horizon/Openstack. A user must have at least one role and can have more than one role. Currently, two roles exist, User and Project Administrator. • **User** role can be assigned to three projects within a provider portal. This role has Horizon access and is able to create instances. • **Project Administrator** role can be assigned to two projects within a provider portal. In addition to the user permission, this role can: ◦ create and remove projects using the provider's portal console (preferred over CIS console)

Parameter	Description
	• add and edit role and remove users using the CIS console • modify project quotas using the CIS console Adding users involves redirection to the federations partner's portal in order to obtain the user_uid from the federation provider.
action	Optional. The action you want to take. Options are: • **add** is the default action that adds the user to a specific tenant. • **move** moves the user from an existing tenant to a new tenant.

Response Example

```
{
    "RequisitionSubmit": {
        "id": 23,
        "customer": "API User",
        "initiator": "API User",
        "dueDateRaw": 1399047699577,
        "dueDate": "05/02/2014 9:21 AM",
        "startedDateRaw": 1399047303153,
        "startedDate": "05/02/2014 9:15 AM",
        "status": "Ongoing"
    }
}
```

Response Parameters

- The date/time format and time zone are based on the preferences set in the login user's personal profile.
- The due date returned in the API response is available only when the asynchronous submission global setting is enabled.
- Formatted dates returned in the API responses are based on the API user's timezone. Raw dates returned are in UTC.

Disassociate a User from a Tenant–v2

Method	URI	Description
PUT	/services/v2/user/tenant?action=<<action>>	This call either removes the user from a specific

Method	URI	Description
		tenant or moves the user from an existing tenant to a new tenant based on the action you specified. The default action is **remove**.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Example

```
{  "ccs_tenant" : "bp_tenant",
   "user_uid" : "testUser",
   "role" : "User"
}
```

Request Parameters

Parameter	Description
ccs_tenant	Required. The identifier of the tenant from which you want the user to disassociate.
user_uid	Required. The identifier of the user from which you want to disassociate.

role	Optional. A role defines privileges and access within Horizon/Openstack. A user must have at least one role and can have more than one role. Currently, two roles exist, User and Project Administrator.
	• **User** role can be assigned to three projects within a provider portal. This role has Horizon access and is able to create instances.
	• **Project Administrator** role can be assigned to two projects within a provider portal. In addition to the user permission, this role can:
	  ◦ create and remove projects using the provider's portal console (preferred over CIS console)
	  ◦ add and edit role and remove users using the CIS console
	  ◦ modify project quotas using the CIS console
	Adding users involves redirection to the federations partner's portal in order to obtain the user_uid from the federation provider.

Parameter	Description
action	Optional. The action you want to take. Options are: • **remove** is the default action that removes the user from a specific tenant. • **move** moves the user from an existing tenant to a new tenant.

Response Example


```
{
    "RequisitionSubmit": {
        "id": 23,
        "customer": "API User",
        "initiator": "API User",
        "dueDateRaw": 1399047699577,
        "dueDate": "05/02/2014 9:21 AM",
        "startedDateRaw": 1399047303153,
        "startedDate": "05/02/2014 9:15 AM",
        "status": "Ongoing"
    }
}
```


Response Parameters

- The date/time format and time zone are based on the preferences set in the login user's personal profile.
- The due date returned in the API response is available only when the asynchronous submission global setting is enabled.
- Formatted dates returned in the API responses are based on the API user's timezone. Raw dates returned are in UTC.

Managing Users

Users are grouped under the CIS tenant construct. This allows a user to be associated with multiple projects belonging to that CIS tenant.

Service Name	Type	URI
Create a User	POST	/services/user
Delete a User	DELETE	/services/serviceProvider/<<SP>>/user/uid/<<uid>>
Get a User	GET	/services/serviceProvider/<<SP>>/user/uid/<<uid>>
Get Project Details for a Project User	GET	/services/serviceProvider/<<SP>>/user/uid/<<uid>>/projects
Update a User	PUT	/services/serviceProvider/<<SP>>/user/uid/<<uid>>
Create a User–v2	POST	/services/v2/user?notification=<<true|false>>
Get a User–v2	GET	/services/v2/serviceProvider/<<SP>>/user/uid/<<uid>>
Get User V2 For Input UIDs	POST	/services/v2/serviceProvider/<<SP>>/user/uids

Create a User

Method	URI	Description
POST	/services/user	Create a user. Call Create a User-V2 if you want to send a welcome letter and include Terms and Conditions.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Example


```
{
"email": "test@cisco.com",
"first_name": "First Name",
"last_name": "Last Name",
"serviceProvider": "Provider",
"ccs_tenant": "f343fgh",
"user_uid": "abc-123",
"role": "User"
}
```


Request Parameters

Parameter	Description
email	Required. The email address of the user the service provider is on-boarding to the Cisco Intercloud Services.
FirstName	Required. The first name of the user the service provider is on-boarding to the Cisco Intercloud Services.
LastName	Required. The last name of the user the service provider is on-boarding to the Cisco Intercloud Services.
serviceProvider	Required. The entity that provides the service for the user.
ccs_tenant	Required. The identifier of the tenant to which this user belongs.
user_uid	Required. The identifier of the user.

Parameter	Description
	This value is supplied by the service provider, and must match the user id supplied for SAML authentication.

role	Optional. A role defines privileges and access within Horizon/Openstack. A user must have at least one role and can have more than one role. Currently, two roles exist, User and Project Administrator.
	• **User** role can be assigned to three projects within a provider portal. This role has Horizon access and is able to create instances.
	• **Project Administrator** role can be assigned to two projects within a provider portal. In addition to the user permission, this role can:
	  • create and remove projects using the provider's portal console (preferred over CIS console)
	  • add and edit role and remove users using the CIS console
	  • modify project quotas using the CIS console
	Adding users involves redirection to the federations partner's portal in order to obtain the user_uid from the federation provider.

Response Example

```
{
    "RequisitionSubmit": {
        "id": 23,
        "customer": "API User",
        "initiator": "API User",
        "dueDateRaw": 1399047699577,
        "dueDate": "05/02/2014 9:21 AM",
        "startedDateRaw": 1399047303153,
        "startedDate": "05/02/2014 9:15 AM",
        "status": "Ongoing"
    }
}
```

Response Parameters

- The date/time format and time zone are based on the preferences set in the login user's personal profile.
- The due date returned in the API response is available only when the asynchronous submission global setting is enabled.
- Formatted dates returned in the API responses are based on the API user's timezone. Raw dates returned are in UTC.

Create a User–v2

Method	URI	Description
POST	/services/v2/user?notification=<<true|false>>	Create a user. You can also set the optional parameters to send a welcome email and pass the Terms and Conditions to the user.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Parameters

Parameter	Description
notification	Optional. • true indicates that a welcome email will be sent to the newly created user. • false (default) indicates otherwise.

Request Example

```
{
"email": "test1@cisco.com",
"first_name": "First Name",
"last_name": "Last Name",
"serviceProvider":"cisco",
"ccs_tenant": "bp-tenant",
"user_uid":"abc1-123",
"role":"User",
"company_name":"abc corp",
"company_address":"123, abc st, Fairfax VA 22031",
"job_role":"engineer",
"terms": {
        "referenceId": "32432",
        "status": "Accepted",
        "document_name": "doc.pdf",
        "signed_date": "03/06/2015"
    }
}
```

Request Parameters

Parameter	Description
email	Required. The email address of the user the service provider is on-boarding to the Cisco Intercloud Services.
first_name	Required. The first name of the user the service provider is on-boarding to the Cisco Intercloud Services.
last_name	Required. The last name of the user the service provider is on-boarding to the Cisco Intercloud Services.
serviceProvider	Required. The entity that provides the service for the user.
ccs_tenant	Required. The identifier of the tenant to which this user belongs.
user_uid	Required. The identifier of the user. This value is supplied by the service provider, and must match the user id supplied for SAML authentication.
role	Required. There are two role assignments: user and administrator. • **User** is able to perform add, modify, and/or delete operations. • **Administrator** can add users to or remove users from the project in addition to the user permission.
company_name	Optional.
company_ address	Optional.
Job_role	Optional.
Terms	Optional.

Response Example


```
{
    "RequisitionSubmit": {
        "id": 7601,
        "customer": "nsapi@cisco.com nsapi@cisco.com",
        "initiator": "nsapi@cisco.com nsapi@cisco.com",
        "dueDateRaw": 1434481253973,
        "dueDate": "06/16/2015 7:00 PM",
        "startedDateRaw": 1434481217877,
        "startedDate": "06/16/2015 7:00 PM",
        "status": "Ongoing"
    },
    "SDPOnboardUser": {
        "message": "",
        "status": "201"
    }
}
```


- The date/time format and time zone are based on the preferences set in the login user's personal profile.
- The due date returned in the API response is available only when the asynchronous submission global setting is enabled.
- Formatted dates returned in the API responses are based on the API user's timezone. Raw dates returned are in UTC.

Delete a User

Method	URI	Description
DELETE	/services/serviceProvider/<<SP>>/user/uid/<<uid>>	Delete a specific user.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Parameters

Parameter	Description
SP	Required. The entity that provides the service for the user.
uid	Required. The ID of the user you want to delete.

Response Example


```
{
    "RequisitionSubmit": {
        "id": 23,
        "customer": "API User",
        "initiator": "API User",
        "dueDateRaw": 1399047699577,
        "dueDate": "05/02/2014 9:21 AM",
        "startedDateRaw": 1399047303153,
        "startedDate": "05/02/2014 9:15 AM",
        "status": "Ongoing"
    }
}
```


Response Parameters

- The date/time format and time zone are based on the preferences set in the login user's personal profile.
- The due date returned in the API response is available only when the asynchronous submission global setting is enabled.

- Formatted dates returned in the API responses are based on the API user's timezone. Raw dates returned are in UTC.

Get a User

Method	URI	Description
GET	/services/serviceProvider/<<SP>>/user/uid/<<uid>>	Get basic user information, such as the name and role of the user. To get additional information, such as user status, payment type, and service level, call Get a User-V2.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Parameters

Parameter	Description
SP	Required. The entity that provides the service for the user.
uid	Required. The ID of the user whose information you want to return.

Response Example

This request returns the following information associated with the specific user ID.


```
{
"email": "test@cisco.com",
"first_name": "First Name",
"last_name": "Last Name",
"serviceProvider": "Provider",
"ccs_tenant": "f343fgh",
"user_uid": "abc-123",
"role": "User/Administrator"
}
```

Get a User–v2

Method	URI	Description
GET	/services/v2/serviceProvider/<<SP>>/user/uid/<<uid>>	Get the status of the user and the following fields: customerType, paymentType, hasBilling, serviceLevel, subscriptionId, in addition to the basic user information. Call Get a User if you want to retrieve only the basic user information, such as the name and role of the user. **Note**: When a user is inactive or invalid, a 404 is returned.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Parameters

Parameter	Description
SP	Required. The entity that provides the service for the user.
uid	Required. The ID of the user whose information you want to return.

Response Example

This request returns the following information associated with the specific user ID.

```
{
"email": "test@cisco.com",
"first_name": "First Name",
"last_name": "Last Name",
"serviceProvider":"abc company",
"ccs_tenant": "f343fgh",
"user_uid":"abc-123",
"role":"User/Administrator",
"status":"Active/Inactive",
"customerType": "Direct",
"paymentType": "PO",
"hasBilling": "True",
"serviceLevel": "Standard",
"subscriptionID": "w1234",
"tenantType": "Production"
}
```

Get User V2 For Input UIDs

Method	URI	Description
POST	/services/v2/serviceProvider/<<SP>>/user/uids	Get user information for a set of specific input UIDs.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Example

There must be at least one user in the input list.

```
[
        "user1",
        "user2"
]
```

Request Parameters

Parameter	Description
SP	Required. The entity that provides the service for the user.

Response Example


```
[
  {
    "customerType": null,
    "serviceLevel": null,
    "user_uid": "user1",
    "description":"tenant1-description",
    "ccs_tenant": "tenant1"
 },
  {
    "customerType": "Direct",
    "serviceLevel": "Standard",
    "user_uid": "user2",
    "description":"tenant1-description",
    "ccs_tenant": "tenant1"
  }
]
```

Get Project Details for a Project User

Get Project Details for a Project User–v2

Method	URI	Description
GET	/services/serviceProvider/<<SP>> /user/uid/<<uid>>/projects	Get project information for a project user.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Parameter	Description
SP	Required. The entity that provides the service for the user.
uid	Required. The ID of the user whose project you want to return.

Response Example

This request returns the name and ID of the project.

```
{
  "projects": [
    {
      "name": "0f48c063-37c6-4620-b070-71bee6c00396",
      "status": "Active",
      "description": "aj2-project",
      "externalID": "17ec0f15635049518575fc79b09c46e8",
      "billToOrganization": "Cisco",
      "emailAddress": "userx@cisco.com",
      "role": "User",
      "displayName": "aj2-project-name-1574",
      "buyerTenantID": "cisco",
      "providerTarget": "US-RDU-2",
      "keystoneEndpoint": "https://us-rdu-2.cisco.com:5000/v2.0",
      "horizonURL": "https://us-rdu-2.cisco.com/horizon"
    },
    {
      "name": "886208b2-44d5-41cb-857b-fb90dd781239",
      "status": "Active",
      "description": "Test environment for IDM team",
      "externalID": "2ab5ab2d256744e39db54c2eb4a9a679",
      "billToOrganization": "cisco",
      "emailAddress": "userx@cisco.com",
      "role": "User",
      "displayName": "IDM_Test_Env",
      "buyerTenantID": "cisco",
      "providerTarget": "US-RDU-2",
      "keystoneEndpoint": "https://us-rdu-2.cisco.com:5000/v2.0",
      "horizonURL": "https://us-rdu-2.cisco.com/horizon"
    }
  ]
}
```

Update a User

Method	URI	Description
PUT	/services/serviceProvider/<<SP>> /user/uid/<<uid>>	Update email address, first and last name of a specific user.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Example

```
{
"email": "test@cisco.com",
"first_name": "First Name",
"last_name": "Last Name"
}
```

Request Parameters

Parameter	Description
user_uid	Required. The ID of the user whose information you want to update.

Response Example

```
{
    "RequisitionSubmit": {
        "id": 23,
        "customer": "API User",
        "initiator": "API User",
        "dueDateRaw": 1399047699577,
        "dueDate": "05/02/2014 9:21 AM",
        "startedDateRaw": 1399047303153,
        "startedDate": "05/02/2014 9:15 AM",
        "status": "Ongoing"
    }
}
```

Response Parameters

- The date/time format and time zone are based on the preferences set in the login user's personal profile.
- The due date returned in the API response is available only when the asynchronous submission global setting is enabled.
- Formatted dates returned in the API responses are based on the API user's timezone. Raw dates returned are in UTC.

Managing User Roles

You can use role-based access control (RBAC) to restrict system access to authorized users. Roles are assigned and given various privileges.

Service Name	Type	URI
Create User Roles	POST	/services/v2/serviceProvider/<<SP>>/user/uid/<<uid>>/roles
Update User Role Association	PUT	/services/v2/serviceProvider/<<SP>>/user/uid/<<uid>>/roles
Delete User Role Association	DELETE	/services/v2/serviceProvider/<<SP>>/user/uid/<<uid>>/roles
Get User Roles	GET	/services/v2/serviceProvider/<<SP>>/user/uid/<<uid>>/roles
Get User Accounts	GET	/services/v2/serviceProvider/<<SP>>/user/uid/<<uid>>/accounts
Get Roles for a Specific Account and User	GET	/services/v2/serviceProvider/<<SP>>/user/uid/<<uid>>/account/<<ID>>/roles
Get Roles for a Specific Tenant and User	GET	/services/v2/serviceProvider/<<SP>>/user/uid/<<uid>>/tenant/<<ccs-tenant>>/roles
Get Users for a Specific Account	GET	/services/v2/serviceProvider/<<SP>>/account/<<ID>>/users
Get Users for a Specific Tenant	GET	/services/v2/serviceProvider/<<SP>>/tenant/<<ccs-tenant>>/users
Get Users for a Specific Account and Role	GET	/services/v2/serviceProvider/<<SP>>/account/<<ID>>/role/<<rolename>>/users
Get Users for a Specific Tenant and Role	GET	/services/v2/serviceProvider/<<SP>>/tenant/<<ccs-tenant>>/role/<<rolename>>/users

Create User Roles

Method	URI	Description
POST	/services/v2/serviceProvider/<<SP>>/user/uid/<<uid>>/roles	A user can have different roles in different tenants. A user can also have multiple roles in the same tenant.

Request Header

```
username: <nsapi_username>, password: <password>,
Content-Type: application/json, Accept: application/json
```

Request Example

```
{
   "account_name":"Account1",
   "ccs_tenant":"Tenant1",
   "roles": [{
                "name":"User Administrator",
                "status":"Active"
            },
        {
                "name":"Buyer",
                "status":"Active"
            },
        {
                "name":"Administrator",
                "status":"Active"
            },
        {
                "name":"Billing Administrator",
                "status":"Active"
            },
        {
                "name":"User",
                "status":"Active"
            }
    ]
}
```

Request Parameters

Parameter	Description
account_name	Required.
ccs_tenant	Required. The tenant identifier
roles	Required. You can assign multiple roles to a user. • **User** is a project-related role that has Horizon access, and is able to create instances within the project, edit the project profile, and view content and support pages. • **Administrator** is a project-related role that has Horizon access, and can add or remove users from the project in addition to the permission of the User role. • **Billing Administrator** is an account-related role that has limited access to the Account Management Portal: delete account, view invoice and payment methods. In addition, this role can browse request information of the marketplace. • **User Administrator** is an account-related role that can add users and assign user roles to the Account Management Portal. This role can also browse request information of the marketplace. • **Buyer** is an account-related role that has access to the marketplace for placing and canceling a purchase, and browsing request information.
status	Optional. When a user is assigned multiple roles, you can use the following options to indicate whether or not the role is granted. • **Active** for the role is granted. • **Inactive** for otherwise.

Response Example


```
{
  "status": "Success",
  "statusCode": "200",
  "message": "User-Role(s) Association created successfully"
}
```

Delete User Role Association

Method	URI	Description
DELETE	/services/v2/serviceProvider/<<SP>>/user/uid/<<uid>>/roles	Remove all user role associations.

Request Header

```
username: <nsapi_username>, password: <password>,
Content-Type: application/json, Accept: application/json
```

Request Example

```
{
   "account_name":"Account1",
   "ccs_tenant":"Tenant1"
}
```

Request Parameters

Parameter	Description
account_name	Required.
ccs_tenant	Required. The tenant identifier
roles	Required. Roles are: • User • Administrator • Billing Administrator • User Administrator • Buyer

Response Example

```
{
  "status": "Success",
  "statusCode": "200",
  "message": "User-Role(s) Association deleted successfully"
}
```

Get User Accounts

Method	URI	Description
GET	/services/v2/serviceProvider/<<SP>>/user/uid/<<uid>>/accounts	Get user accounts.

Request Header

```
username: <nsapi_username>, password: <password>,
Content-Type: application/json, Accept: application/json
```

Request Parameters

Parameter	Description
SP	Required. The identifier of the service provider.
uid	Required. The identifier of the user.

Response Example


```
{
   "accounts":[
      {
         "name":"Account1",
         "ccs_tenant":"Tenant1"
      }
   ]
}
```

Get Users for a Specific Account

Method	URI	Description
GET	/services/v2/serviceProvider/<<SP>>/account/<<ID>>/users	Get users for a specific account.

Request Header

```
username: <nsapi_username>, password: <password>,
Content-Type: application/json, Accept: application/json
```

Request Parameters

Parameter	Description
SP	Required. The identifier of the service provider.
ID	Required. The account ID.

Response Example


```
{
   "users":[
      {
         "uid":"test1@cisco.com",
         "ccs_tenant":"Tenant1",
         "account_name":"Account1",
         "roles":[
            "User Administrator",
            "Buyer",
            "Administrator",
            "Billing Administrator",
            "User"
         ]
      },
      {
         "uid":"abc1@cisco.com",
         "ccs_tenant":"Tenant1",
         "account_name":"Account1",
         "roles":[
            "User Administrator",
            "Buyer"
         ]
      }
   ]
}
```

Get Users for a Specific Account and Role

Method	URI	Description
GET	/services/v2/serviceProvider/<<SP>>/account/<<ID>>/role/<<rolename>>/users	Get users for a specific account and role.

Request Header

```
username: <nsapi_username>, password: <password>,
Content-Type: application/json, Accept: application/json
```

Request Parameters

Parameter	Description
SP	Required. The identifier of the service provider.
ID	Required. The account ID.
rolename	Required. The role to be returned.

Response Example


```
{
   "users":[
      {
         "uid":"test1@cisco.com",
         "ccs_tenant":"Tenant1",
         "account_name":"Account1"
      }
   ]
}
```

Get Roles for a Specific Account and User

Method	URI	Description
GET	/services/v2/serviceProvider/<<SP>>/user/uid/<<uid>>/account/<<ID>>/roles	Get the roles for a specific account and user.

Request Header

```
username: <nsapi_username>, password: <password>,
Content-Type: application/json, Accept: application/json
```

Request Parameters

Parameter	Description
SP	Required. The identifier of the service provider.
uid	Required. The identifier of the user.
ID	Required. The account ID.

Response Example


```
{
   "roles":[
      {
         "ccs_tenant":"Tenant1",
         "account_name":"Account1",
         "names":[
            "User",
            "Administrator",
            "Buyer",
            "Billing Administrator",
            "User Administrator"
         ]
      }
   ]
}
```

Get Users for a Specific Tenant

Method	URI	Description
GET	/services/v2/serviceProvider/<<SP>>/tenant/<<ccs-tenant>>/users	Get users for a specific tenant. This method returns only users who have been assigned a role with that tenant using the Create User Roles API call.

Request Header

```
username: <nsapi_username>, password: <password>,
Content-Type: application/json, Accept: application/json
```

Request Parameters

Parameter	Description
SP	Required. The identifier of the service provider.
ccs-tenant	Required. The identifier of the tenant.

Response Example


```
{
    "users": [
        {
            "uid": "idm-test",
            "ccs_tenant": "Tenant1",
            "account_name": "Account1",
            "roles": [
                "User",
                "Administrator",
                "Buyer",
                "Billing Administrator",
                "User Administrator"
            ]
        }
    ]
}
```

Get Users for a Specific Tenant and Role

Method	URI	Description
GET	/services/v2/serviceProvider/<<SP>>/tenant/<<ccs-tenant>>/role/<<rolename>>/users	Get users for a specific tenant and role. This method returns only users who have been assigned a role with that tenant using the Create User Roles API call.

Request Header

```
username: <nsapi_username>, password: <password>,
Content-Type: application/json, Accept: application/json
```

Request Parameters

Parameter	Description
SP	Required. The identifier of the service provider.
ccs-tenant	Required. The identifier of the tenant.
rolename	Required. The role to be returned.

Response Example


```
{
    "users": [
        {
            "uid": "idm-test",
            "ccs_tenant": "Tenant1",
            "account_name": "Account1"
        }
    ]
}
```

Get Roles for a Specific Tenant and User

Method	URI	Description
GET	/services/v2/serviceProvider/<<SP>>/user/uid/<<uid>>/tenant/<<ccs-tenant>>/roles	Get the roles of a user of a specific tenant. This method returns only users who have been assigned a role with that tenant using the Create User Roles API call.

Request Header

```
username: <nsapi_username>, password: <password>,
Content-Type: application/json, Accept: application/json
```

Request Parameters

Parameter	Description
SP	Required. The identifier of the service provider.
uid	Required. The identifier of the user.
ccs-tenant	Required. The identifier of the tenant.

Response Example


```
{
   "roles":[
      {
         "ccs_tenant":"Tenant1",
         "account_name":"Account1",
         "names":[
            "User",
            "Administrator",
            "Buyer",
            "Billing Administrator",
            "User Administrator"
         ]
      }
   ]
}
```

Get User Roles

Method	URI	Description
GET	/services/v2/serviceProvider/<<SP>>/user/uid/<<uid>>/roles	Get user roles. This method returns only users who have been assigned a role with that tenant using the Create User Roles API call.

Request Header

```
username: <nsapi_username>, password: <password>,
Content-Type: application/json, Accept: application/json
```

Request Parameters

Parameter	Description
SP	Required. The identifier of the service provider.
uid	Required. The identifier of the user.

Response Example


```
{
   "roles":[
      {
         "account_name":"Account1",
         "ccs_tenant":"Tenant1",
         "names":[
            "User Administrator",
            "Buyer",
            "Administrator",
            "Billing Administrator",
            "User"
         ]
      }
   ]
}
```

Update User Role Association

Method	URI	Description
PUT	/services/v2/serviceProvider/<<SP>>/user/uid/<<uid>>/roles	Update user role association.

Request Header

```
username: <nsapi_username>, password: <password>,
Content-Type: application/json, Accept: application/json
```

Request Example

```
{
   "account_name":"Account1",
   "ccs_tenant":"Tenant1",
   "roles": [{
                "name":"User Administrator",
                "status":"Active"
            },
        {
                "name":"Buyer",
                "status":"Inactive"
            }
    ]
}
```

Request Parameters

Parameter	Description
account_name	Required.
ccs_tenant	Required. The tenant identifier
roles	Required. You can assign multiple roles to a user. • **User** is a project-related role that has Horizon access, and is able to create instances within the project, edit the project profile, and view content and support pages. • **Administrator** is a project-related role that has Horizon access, and can add or remove users from the project in addition to the permission of the User role. • **Billing Administrator** is an account-related role that has limited access to

Parameter	Description
	the Account Management Portal: delete account, view invoice and payment methods. In addition, this role can browse request information of the marketplace. • **User Administrator** is an account-related role that can add users and assign user roles to the Account Management Portal. This role can also browse request information of the marketplace. • **Buyer** is an account-related role that has access to the marketplace for placing and canceling a purchase, and browsing request information.
status	Optional. When a user is assigned multiple roles, you can use the following options to indicate whether or not the role is granted. • **Active** for the role is granted. • **Inactive** for otherwise.

Response Example

```
{
  "status": "Success",
  "statusCode": "200",
  "message": "User-Role(s) Association updated successfully"
}
```

Managing Projects

Projects are used as organizational boundaries within an OpenStack-based cloud. Within these boundaries, users have access to infrastructure resources such as instances (VMs), firewalls, load-balancers, and networks.

Service Name	Type	URI
Create a Project	POST	/services/project
Get Project Info by Req Id	GET	/services/project/byReqId/<<reqID>>
Remove a Project	DELETE	/services/project/<<projectID>>?force=true/false/verify
Resume a Project	PUT	/services/project/<<projectID>>/resume
Suspend a Project	PUT	/services/project/<<projectID>>/suspend
Associate a User to Project	POST	/services/user/project
Disassociate a User from Project	PUT	/services/user/project
Version 2		
Create a Project-v2	POST	/services/v2/project
Get Project Users-v2	GET	/services/v2/project/<<id>>/users
Get a Project by Name-v2	GET	/services/v2/project/displayName/<<name>>
Get a Project by a Project ID-v2	GET	/services/v2/project/<<id>>
Get Project Info by Req ID-v2	GET	/services/v2/project/byReqId/<<reqId>>
Get Project Details for a Project User-v2	GET	/services/v2/serviceProvider/<<SP>>/user/uid/<<uid>>/projects
Get Projects of a Specific Tenant of a Partner-v2	GET	/services/v2/serviceProvider/<<SP>>/tenants

Create a Project

Method	URI	Description
POST	/services/project	Create a project.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Example

```
{
"displayName":"<Project Name>",
"description":"<description>",
"ccs_tenant":"<TenantID>",
"providerTarget":"US-RDU-1",
"applicationID":"<ApplicationID>"
}
```

Request Parameters

Parameter	Description
displayName	Required. The name for the project that is displayed on Horizon.
description	Optional. A notation that describes the project.
ccs_tenant	Required. The identifier of the tenant to which this project belongs.
providerTarget	Required. The location of the cloud where the project is available.
applicationID	Optional. The user account name that is displayed on Horizon. The maximum length is 128 alphanumeric characters. Non-alphanumeric characters are replaced with hyphens "-" as in "R&D" to "R-D”.

Response Example


```
{
    "RequisitionSubmit": {
        "id": 23,
        "customer": "API User",
        "initiator": "API User",
        "dueDateRaw": 1399047699577,
        "dueDate": "05/02/2014 9:21 AM",
        "startedDateRaw": 1399047303153,
        "startedDate": "05/02/2014 9:15 AM",
        "status": "Ongoing”
    }
}
```


Response Parameters

- The date/time format and time zone are based on the preferences set in the login user's personal profile.
- The due date returned in the API response is available only when the asynchronous submission global setting is enabled.
- Formatted dates returned in the API responses are based on the API user's timezone. Raw dates returned are in UTC.

Create a Project-v2

Method	URI	Description
POST	/services/v2/project	Create a project.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Example

```
{
"displayName":"<Project Name>",
"description":"<description>",
"ccs_tenant":"<TenantID>",
"providerTarget":"US-RDU-1",
"applicationID":"<ApplicationID>"
}
```

Request Parameters

Parameter	Description
displayName	Required. The name for the project that is displayed on Horizon.
description	Optional. A notation that describes the project.
ccs_tenant	Required. The identifier of the tenant to which this project belongs.
providerTarget	Required. The location of the cloud where the project is available.
applicationID	Optional. The user account name that is displayed on Horizon. The maximum length is 128 alphanumeric characters. Non-alphanumeric characters are replaced with hyphens "-" as in "R&D" to "R-D”.

Response Example


```
{
    "RequisitionSubmit": {
        "id": 23,
        "customer": "API User",
        "initiator": "API User",
        "dueDateRaw": 1399047699577,
        "dueDate": "05/02/2014 9:21 AM",
        "startedDateRaw": 1399047303153,
        "startedDate": "05/02/2014 9:15 AM",
        "status": "Ongoing”
    }
}
```


Response Parameters

- The date/time format and time zone are based on the preferences set in the login user's personal profile.
- The due date returned in the API response is available only when the asynchronous submission global setting is enabled.
- Formatted dates returned in the API responses are based on the API user's timezone. Raw dates returned are in UTC.

Get Project Users–v2

Method	URI	Description
GET	/services/v2/project/<<id>>/users	Get all users of a project.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Response Example


```
[
  {
    "email": "user1@cisco.com",
    "user_uid": "user1",
    "role": "user"
  },
  {
    "email": "user2@cisco.com",
    "user_uid": "user2",
    "role": "admin,user"
  }
]
```

Get Project Info by Req ID

Method	URI	Description
GET	/services/project/byReqId/<<reqID>>	Return a projectid (UUID) that can be used for user assignment and correlation of mediation data.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Parameters

Parameter	Description
reqID	Required.

Response Example

This request returns the following information for the project associated with the specific request ID:


```
{
   "projectId":"435443sdfsdfadfaf232",
   "displayName":"ProjectName",
   "description":"Project Description",
   "providerTarget":"ProviderTarget",
   "ccs_tenant":"3243dsdad"
}
```

Get Project Info by Req ID-v2

Method	URI	Description
GET	/services/v2/project/byReqId/<<reqId>>	Return a projectid (UUID) that can be used for user assignment and correlation of mediation data. It also returns the project status .

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Parameters

Parameter	Description
reqID	Required.

Response Example

This request returns the following information for the project associated with the specific request ID and its status.


```
{
  "projectId": "d7ceb94260ff49628dfd231111f5392d",
  "displayName": "idm-test",
  "description": "idm-test-project",
  "providerTarget": "US-RDU-2",
  "ccs_tenant": "idm-test-tenant",
  "status": "Active"
}
```

Remove a Project

Method	URI	Description
DELETE	/services/project/<<projectID>>?force=true/false/verify	Remove a project. Before removing a project, it is best practice to first clean up all resources , such as shutting down all instances and removing images and snapshots. (Deleting these resources might take a while depending on the number of your allocated resources.) During cleaning up project associated resources, the backend process exits immediately if it fails to delete "Nova" and "Cinder" resources. The backend process removes and logs all issues for the remaining project resources.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Parameters

Parameter	Description
projectID	Required. The ID of the project you want to remove.
force	Optional: • **True** indicates that the backend process deletes resources associated with the project. If the process fails to clean up any resources, the requisition will be canceled. • **False** (default) indicates that the backend process deletes project without verifying any remaining resources. In this case, caller must clean all resources first to avoid any resources still in the system without being aware of it. There are still some resources attached to the project, and user can clean up manually.

Parameter	Description
	• **Verify** indicates that the backend process detects remaining resources and cancels the requisition; the process is stopped without removing any resources. If there are no remaining resources attached to the project, the project is deleted.

Response Example


```
{
    "RequisitionSubmit": {
        "id": 23,
        "customer": "API User",
        "initiator": "API User",
        "dueDateRaw": 1399047699577,
        "dueDate": "05/02/2014 9:21 AM",
        "startedDateRaw": 1399047303153,
        "startedDate": "05/02/2014 9:15 AM",
        "status": "Ongoing”
    }
}
```


Response Parameters

- The date/time format and time zone are based on the preferences set in the login user's personal profile.
- The due date returned in the API response is available only when the asynchronous submission global setting is enabled.
- Formatted dates returned in the API responses are based on the API user's timezone. Raw dates returned are in UTC.

Resume a Project

Method	URI	Description
PUT	/services/project/<<projectID>>/resume	Resume a project from a suspended state and reinstate access to its resources.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Parameters

Parameter	Description
projectID	Required. The ID of the project you want to resume its access to resources.

Response Example


```
{
    "RequisitionSubmit": {
        "id": 23,
        "customer": "API User",
        "initiator": "API User",
        "dueDateRaw": 1399047699577,
        "dueDate": "05/02/2014 9:21 AM",
        "startedDateRaw": 1399047303153,
        "startedDate": "05/02/2014 9:15 AM",
        "status": "Ongoing"
    }
}
```


Response Parameters

- The date/time format and time zone are based on the preferences set in the login user's personal profile.
- The due date returned in the API response is available only when the asynchronous submission global setting is enabled.

- Formatted dates returned in the API responses are based on the API user's timezone. Raw dates returned are in UTC.

Suspend a Project

Method	URI	Description
PUT	/services/project/ <<projectID>>/suspend	• When a Suspend Project API is issued, access to all projects and instances within those projects is disabled. • Remote access (such as SSH, Telnet, RDP, VNC) to all instances in all projects are blocked. • The project remains in the suspended state until an API call is received to either resume or delete the project. • Resource consumption will still be collected, provided, and charged.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Parameters

Parameter	Description
projectID	Required. The ID of the project you want to suspend its access to resources.

Response Example

```
{
    "RequisitionSubmit": {
        "id": 23,
        "customer": "API User",
        "initiator": "API User",
        "dueDateRaw": 1399047699577,
        "dueDate": "05/02/2014 9:21 AM",
        "startedDateRaw": 1399047303153,
        "startedDate": "05/02/2014 9:15 AM",
        "status": "Ongoing"
    }
}
```

Response Parameters

- The date/time format and time zone are based on the preferences set in the login user's personal profile.
- The due date returned in the API response is available only when the asynchronous submission global setting is enabled.
- Formatted dates returned in the API responses are based on the API user's timezone. Raw dates returned are in UTC.

Associate a User to Project

Method	URI	Description
POST	/services/user/project	Associate a user to a project.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Example

```
{
  "projectId": "<ProjectID>",
  "user_uid": "<<uid>>",
  "role": "User"
}
```

Request Parameters

Parameter	Description
projectID	Required. The ID of the project you want to associate the user to.
user_uid	Required. The ID of the user you want to associate the project to.
role	Optional. The user role permission.

Response Example


```
{
    "RequisitionSubmit": {
        "id": 23,
        "customer": "API User",
        "initiator": "API User",
        "dueDateRaw": 1399047699577,
        "dueDate": "05/02/2014 9:21 AM",
        "startedDateRaw": 1399047303153,
        "startedDate": "05/02/2014 9:15 AM",
        "status": "Ongoing”
    }
}
```


Response Parameters

- The date/time format and time zone are based on the preferences set in the login user's personal profile.
- The due date returned in the API response is available only when the asynchronous submission global setting is enabled.
- Formatted dates returned in the API responses are based on the API user's timezone. Raw dates returned are in UTC.

Disassociate a User from Project

Method	URI	Description
PUT	/services/user/project	Disassociate a user from a project.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Example

```
{
  "projectId": "<ProjectID>",
  "user_uid": "<<uid>>",
  "role": "User"
}
```

Request Parameters

Parameter	Description
projectID	Required. The ID of the project from which the user is you want to disassociate the user from.
user_uid	Required. The ID of the user you want to disassociate the project from.

role	Optional. A role defines privileges and access within Horizon/Openstack. A user must have at least one role and can have more than one role. Currently, two roles exist, User and Project Administrator.
	• **User** role can be assigned to three projects within a provider portal. This role has Horizon access and is able to create instances.
	• **Project Administrator** role can be assigned to two projects within a provider portal. In addition to the user permission, this role can:
	  ◦ create and remove projects using the provider's portal console (preferred over CIS console)
	  ◦ add and edit role and remove users using the CIS console
	  ◦ modify project quotas using the CIS console

Parameter	Description
	Adding users involves redirection to the federations partner's portal in order to obtain the user_uid from the federation provider.

Response Example

```
{
    "RequisitionSubmit": {
        "id": 23,
        "customer": "API User",
        "initiator": "API User",
        "dueDateRaw": 1399047699577,
        "dueDate": "05/02/2014 9:21 AM",
        "startedDateRaw": 1399047303153,
        "startedDate": "05/02/2014 9:15 AM",
        "status": "Ongoing"
    }
}
```

Response Parameters

- The date/time format and time zone are based on the preferences set in the login user's personal profile.
- The due date returned in the API response is available only when the asynchronous submission global setting is enabled.
- Formatted dates returned in the API responses are based on the API user's timezone. Raw dates returned are in UTC.

Managing Quota

You can create, view, and manage resources, such as CPU, memory, and floating IP, using the following APIs:

- Account Quota
- Tenant Quota
- Check Requisition Status

Account Quota

Service Name	Type	URI
Create an Account Quota	POST	/RequestCenter/nsapi/transaction/requisitions
Delete an Account Quota	POST	/RequestCenter/nsapi/transaction/requisitions
Get Account Quota based on Region	GET	/RequestCenter/nsapi/serviceitem/namedquery/id/4257ca02-72d6-4e18-a6ee-d1e30742bdd9?account=<account>&region=<region>
Get Account Quota based on Service	GET	/RequestCenter/nsapi/serviceitem/namedquery/id/9f1cc795-f76a-4a03-bc53-5906ab035a3a?account=<account>&service=<service>
Get Account Quota based on Service and Region	GET	/RequestCenter/nsapi/serviceitem/namedquery/id/69cf7625-a6b6-4ff7-9f02-238b05465865?account=<account>&service=<service>&region=<region>
Get All Accounts Quota	GET	/RequestCenter/nsapi/serviceitem/namedquery/id/e94483cf-4494-46c4-a72d-6bd933250331
Get Specific Account Quota	GET	/RequestCenter/nsapi/serviceitem/namedquery/id/fbff8a44-181d-48da-9ce8-f6f4bdb153e1?account=<account>
SetMaximum Account Quota Pool	POST	/RequestCenter/nsapi/transaction/requisitions

Create an Account Quota

Method	URI	Description
POST	/services/v2/quota/account	Create a quota for an account. The account can be based on a single metric or on multiple metrics.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Example

The following example shows an account with multiple metrics.

```
{
    "name": "Microsoft",
    "quotaDetails": [
        {
            "service": "IAAS",
            "region": "US-RDU-1",
            "metric": "CPU",
            "unit": "Quantity",
            "maximum": "180"
        },
        {
            "service": "IAAS",
            "region": "US-RDU-1",
            "metric": "Memory",
            "unit": "GB",
            "maximum": "8192"
        }
    ]
}
```

Request Parameters

Parameter	Description
Service	Required. The type of service.
Region	Required. The geographical location where the account is available.

Parameter	Description
Metric	Required. The following metrics are supported: • CPU • Memory • FloatingIP Be sure to use the appropriate unit of measure for the metrics.
Unit	Required. The measurement used for the metrics. • Quantity is used for CPU and FloatingIP. • GB is used for Memory. The quota value for the account must be less than or equal to the Available quota of account.
Maximum	Required. The maximum quota value.

Response Example


```
{
    "RequisitionSubmit": {
        "id": 38631,
        "customer": "nsapi@cisco.com nsapi@cisco.com",
        "initiator": "nsapi@cisco.com nsapi@cisco.com",
        "dueDateRaw": 1442258525083,
        "dueDate": "09/14/2015 7:22 PM",
        "startedDateRaw": 1442258525050,
        "startedDate": "09/14/2015 7:22 PM",
        "status": "Ongoing"
    }
}
```


Response Parameters

- The date/time format and time zone are based on the preferences set in the login user's personal profile.
- The due date returned in the API response is available only when the asynchronous submission global setting is enabled.
- Formatted dates returned in the API responses are based on the API user's timezone. Raw dates returned are in UTC.

Delete an Account Quota

Method	URI	Description
POST	/RequestCenter/nsapi/transaction/requisitions	Delete the account quota when the Consumed quota is 0, which means the account quota is deleted only when no resource is consumed by any of its tenants.

Request Header

```
username: <nsapi_username>, password: <password>,
Content-Type: application/json, Accept: application/json
```

Request Payload Example

```
{
    "name": "Microsoft",
    "quotaDetails": [
        {
            "service": "IAAS",
            "region": "US-RDU-1",
            "metric": "CPU",
            "unit": "Quantity"
        }
    ]
}
```

Request Parameters

Parameter	Description
Account	Required. The name of the account.
Service	Required. The type of service.
Region	Required. The geographical location where the account is available.
Metric	Required. The following metrics are supported: • CPU • Memory • FloatingIP Be sure to use the appropriate unit of measure for the metrics.

Parameter	Description
Unit	Required. The measurement used for the metrics. • Quantity is used for CPU and FloatingIP. • GB is used for Memory.

Response Example


```
{
    "RequisitionSubmit": {
        "id": 6382,
        "customer": "nsapi@cisco.com nsapi@cisco.com",
        "initiator": "nsapi@cisco.com nsapi@cisco.com",
        "dueDateRaw": 1433409983017,
        "dueDate": "06/04/2015 9:26 AM",
        "startedDateRaw": 1433409982967,
        "startedDate": "06/04/2015 9:26 AM",
        "status": "Ongoing"
    }
}
```


Response Parameters

- The date/time format and time zone are based on the preferences set in the login user's personal profile.
- The due date returned in the API response is available only when the asynchronous submission global setting is enabled.
- Formatted dates returned in the API responses are based on the API user's timezone. Raw dates returned are in UTC.

Get an Account Quota Based on a Region

Method	URI	Description
GET	/RequestCenter/nsapi/serviceitem/namedquery/id/4257ca02-72d6-4e18-a6ee-d1e30742bdd9?account=<account>&region=<region>	Return the maximu consumed and available quota for an account based on the specific region.

Request Header

```
username: <nsapi_username>, password: <password>,
Content-Type: application/json, Accept: application/json
```

Request Example

```
/RequestCenter/nsapi/serviceitem/namedquery/id/4257ca02-72d6-4e18-a6ee-
d1e30742bdd9?account=Microsoft&region=US-TEXAS-2
```

Request Parameters

None

Response Example


```
{
    "List": [{
        "Available": "40.00000",
        "Service": "SAPHANA",
        "Region": "US-TEXAS-2",
        "Metric": "CPU",
        "Maximum": "50.00000",
        "Account": "Microsoft",
        "Consumed": "10.00000",
        "Unit": "Quantity"
    }]
}
```

Get an Account Quota Based on Service

Method	URI	Description
GET	/RequestCenter/nsapi/serviceitem/namedquery/id/9f1cc795-f76a-4a03-bc53-5906ab035a3a?account=<account>&service=<service>	Return the maximum consumed and available quota for an account based the specific service.

Request Header

```
username: <nsapi_username>, password: <password>,
Content-Type: application/json, Accept: application/json
```

Request Example

```
/RequestCenter/nsapi/serviceitem/namedquery/id/9f1cc795-f76a-4a03-bc53-
5906ab035a3a?account=Microsoft&service=IAAS
```

Request Parameters

None

Response Example


```
{
    "List": [
        {
            "Available": "60.00000",
            "Service": "IAAS",
            "Region": "US-RDU-2",
            "Metric": "CPU",
            "Maximum": "100.00000",
            "Account": "Microsoft",
            "Consumed": "30.00000",
            "Unit": "Quantity"
        },
        {
            "Available": "8192.00000",
            "Service": "IAAS",
            "Region": "US-RDU-2",
            "Metric": "Memory",
            "Maximum": "8192.00000",
            "Account": "Microsoft",
            "Consumed": "0.00000",
            "Unit": "GB"
        },
        {
            "Available": "100.00000",
            "Service": "IAAS",
            "Region": "US-RDU-2",
            "Metric": "CPU",
            "Maximum": "100.00000",
            "Account": "Microsoft",
            "Consumed": "0.00000",
            "Unit": "Quantity"
        }
    ]
}
```

Get Account Quota Based on Service and Region

Method	URI	Description
GET	/RequestCenter/nsapi/serviceitem/namedquery/id/69cf7625-a6b6-4ff7-9f02-238b05465865?account=<account>& service=<service>&region=<region>	Return the maximum consumed and available quota for an account based on service and region.

Request Header

```
username: <nsapi_username>, password: <password>,
Content-Type: application/json, Accept: application/json
```

Request Example

```
/RequestCenter/nsapi/serviceitem/namedquery/id/69cf7625-a6b6-4ff7-9f02-
238b05465865?account=Microsoft&service=SAPHANA&region=US-TEXAS-2
```

Request Parameters

None

Response Example


```
{
    "List": [{
        "Available": "40.00000",
        "Service": "SAPHANA",
        "Region": "US-TEXAS-2",
        "Metric": "CPU",
        "Maximum": "50.00000",
        "Account": "Microsoft",
        "Consumed": "10.00000",
        "Unit": "Quantity"
    }]
}
```

Get Quota for All Accounts

Method	URI	Description
GET	/RequestCenter/nsapi/serviceitem/namedquery/id/e94483cf-4494-46c4-a72d-6bd933250331	Return the maximum consumed and available quota for all accounts. To return quotas for a specific account, see Get Quota for a Specific Account.

Request Header

```
username: <nsapi_username>, password: <password>,
Content-Type: application/json, Accept: application/json
```

Request Example

```
/RequestCenter/nsapi/serviceitem/namedquery/id/e94483cf-4494-46c4-a72d-6bd933250331
```

Request Parameters

None

Response Example


```
{
    "List": [
        {
            "Available": "40.00000",
            "Service": "SAPHANA",
            "Region": "US-TEXAS-2",
            "Metric": "CPU",
            "Maximum": "50.00000",
            "Account": "Microsoft",
            "Consumed": "10.00000",
            "Unit": "Quantity"
        },
        {
            "Available": "70.00000",
            "Service": "IAAS",
            "Region": "US-RDU-2",
            "Metric": "CPU",
            "Maximum": "100.00000",
            "Account": "Microsoft",
            "Consumed": "30.00000",
            "Unit": "Quantity"
        },
        {
            "Available": "8192.00000",
            "Service": "IAAS",
            "Region": "US-RDU-2",
            "Metric": "Memory",
            "Maximum": "8192.00000",
            "Account": "Microsoft",
            "Consumed": "0.00000",
            "Unit": "GB"
        },
        {
            "Available": "40.00000",
            "Service": "IAAS",
            "Region": "US-RDU-2",
            "Metric": "FloatingIP",
            "Maximum": "50.00000",
            "Account": "Toyota",
            "Consumed": "10.00000",
            "Unit": "Quantity"
        }
    ]
}
```

Response Parameters

Parameter	Description
Available	The quota value that is available for consumption.
Service	The type of service.
Region	The geographical location where the account is available.
Metric	Metric can be CPU, Memory, or FloatingIP.
Maximum	The maximum quota value for the account.
Account	The name of the account.
Consumed	The quota value that has been consumed.
Unit	The measurement used for the metrics.

Get Quota for a Specific Account

Method	URI	Description
GET	/services/v2/quota/account/<accountName>	Return the quota information for a specific account. To return quotas for All accounts, see Get Quota for All Accounts.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Example

```
/services/v2/quota/account/Microsoft
```

Response Example


```
{
    "List": [
        {
            "Available": "70.00000",
            "Service": "IAAS",
            "Region": "US-RDU-2",
            "Metric": "CPU",
            "Maximum": "100.00000",
            "Account": "Microsoft",
            "Consumed": "30.00000",
            "Unit": "Quantity"
        },
        {
            "Available": "8192.00000",
            "Service": "IAAS",
            "Region": "US-RDU-2",
            "Metric": "Memory",
            "Maximum": "8192.00000",
            "Account": "Microsoft",
            "Consumed": "0.00000",
            "Unit": "GB"
        },
        {
            "Available": "100.00000",
            "Service": "IAAS",
            "Region": "US-RDU-2",
            "Metric": "CPU",
            "Maximum": "100.00000",
            "Account": "Microsoft",
            "Consumed": "0.00000",
            "Unit": "Quantity"
        },
        {
            "Available": "40.00000",
            "Service": "SAPHANA",
            "Region": "US-TEXAS-2",
            "Metric": "CPU",
            "Maximum": "50.00000",
            "Account": "Microsoft",
            "Consumed": "10.00000",
            "Unit": "Quantity"
        }
    ]
}
```

Response Parameters

Parameter	Description
Available	The quota value that is available for consumption.
Service	The type of service.
Region	The geographical location where the account is available.
Metric	Metric can be CPU, Memory, or FloatingIP.
Maximum	The maximum quota value for the account.
Account	The name of the account.
Consumed	The quota value that has been consumed.
Unit	The measurement used for the metrics.

Set Maximum Account Quota Pool

Method	URI	Description
POST	/RequestCenter/nsapi/transaction/requisitions	Increase and/or decrease the account Maximum quota. When this API is called, the account quota is updated with the specified value. To see the account maximum quota and how much quota is consumed, call Get All Accounts quota.

Request Header

```
username: <nsapi_username>, password: <password>,
Content-Type: application/json, Accept: application/json
```

Request Payload Example

```
{
    "requisition": {
        "services": [{
            "name": "Update Quota Pool",
            "dictionaries": [{
                "name": "AccountQuota",
                "data": [{
                    "Account": "Microsoft",
                    "Service": "IAAS",
                    "Region": "US-RDU-1",
                    "Metric": "CPU",
                    "Unit": "Quantity"
                }]
            }, {
                "name": "QuotaType",
                "data": {
                    "Type": "Account",
                    "SetMaximum": 100
                }
            }]
        }]
    }
}
```

Request Parameters

Parameter	Description
Account	Required. The name of the account.
Service	Required. The type of service.
Region	Required. The geographical location where the account is available.
Metric	Required. The following metrics are supported: • CPU • Memory • FloatingIP **Note**: Be sure to use the appropriate unit of measure for the metrics.
Unit	Required. The measurement used for the metrics. • Quantity is used for CPU and FloatingIP. • GB is used for Memory.

Parameter	Description
Type	Required. The type of quota. Options are: • Account • Tenant
SetMaximum	Required. The value to be increased or decreased based on the value specified. The calculation is the difference between the SetMaximum and account Maximum. Suppose you want to increase the account Maximum, and the current account Maximum quota is 5 and the Available quota is 2. Using 10 for the SetMaximum value, the new account Maximum quota becomes 10 and the Available quota 7.

Managing Tenant Quota

Service Name	Type	URI
Create Tenant Quota	POST	/RequestCenter/nsapi/transaction/requisitions
Delete Tenant Quota	POST	/RequestCenter/nsapi/transaction/requisitions
Get Tenant Quota Based on Region	GET	/RequestCenter/nsapi/serviceitem/namedquery/id/37db3ba3-3ac4-4344-83c3-acef86af9d15?account=<account>&tenant=<tenant>&region=<region>
Get Tenant Quota Based on Service	GET	/RequestCenter/nsapi/serviceitem/namedquery/id/4c432711-0a4e-414a-8cc1-a8984b41e963?account=<account>&tenant=<tenant>&service=<service>
Get Tenant Quota Based on Service and Region	GET	/RequestCenter/nsapi/serviceitem/namedquery/id/bfcb9bad-9eb1-4065-bd86-2b9a2fbbd22c?account=<account>&tenant=<tenant>&service=<service>&region=<region>
Get Specific Tenant Quota	GET	/RequestCenter/nsapi/serviceitem/namedquery/id/0647bede-c1b8-4c4d-b8b9-3256f2a1eda2?account=<account>&tenant=<tenant>
Get all Tenants Quota	GET	/RequestCenter/nsapi/serviceitem/namedquery/id/b03bec42-2b49-4cc1-af92-2b65815b9216
SetMaximum Tenant Quota Pool	POST	/RequestCenter/nsapi/transaction/requisitions
Update Tenant Quota	POST	/RequestCenter/nsapi/transaction/requisitions

Create Tenant Quota

Method	URI	Description
POST	/services/v2/quota/tenant	Create a quota for a tenant. The quota can be based on a single metric or on multiple metrics.

Request Header

```
username: <nsapi_username>, password: <password>,
Content-Type: application/json, Accept: application/json
```

Request Example

```
{
    "name": "HR",
    "quotaDetails": [
        {
            "account": "Microsoft",
            "service": "IAAS",
            "region": "US-RDU-1",
            "metric": "CPU",
            "unit": "Quantity",
            "maximum": "180"
        },
        {
            "account": "Microsoft",
            "service": "IAAS",
            "region": "US-RDU-1",
            "metric": "Memory",
            "unit": "GB",
            "maximum": "8192"
        }
    ]
}
```

Request Parameters

Parameter	Description
Account	Required. The name of the account
Tenant	Required. The name of the tenant.
Service	Required. The type of service.
Region	Required. The geographical location where the account is available.

Parameter	Description
Metric	Required. The following metrics are supported: • CPU • Memory • FloatingIP Be sure to use the appropriate unit of measure for the metrics.
Unit	Required. The measurement used for the metrics. • Quantity is used for CPU and FloatingIP. • GB is used for Memory.
Maximum	Required. The maximum quota value for the account. The maximum value must be less than or equals to the available quota of the account; otherwise, no quota will be assigned to the tenant.
ParentType	Required. This is the top-level account assigned to the tenant. ParentType has two options: Account and Tenant. If Tenant is ParentType then this tenant is a sub-tenant.
ParentID	Required. This is the parent identifier.

Response Example


```
{
    "RequisitionSubmit": {
        "id": 6382,
        "customer": "nsapi@cisco.com nsapi@cisco.com",
        "initiator": "nsapi@cisco.com nsapi@cisco.com",
        "dueDateRaw": 1433409983017,
        "dueDate": "06/04/2015 9:26 AM",
        "startedDateRaw": 1433409982967,
        "startedDate": "06/04/2015 9:26 AM",
        "status": "Ongoing"
    }
}
```


Response Parameters

- The date/time format and time zone are based on the preferences set in the login user's personal profile.

- The due date returned in the API response is available only when the asynchronous submission global setting is enabled.
- Formatted dates returned in the API responses are based on the API user's timezone. Raw dates returned are in UTC.

Delete Tenant Quota

Method	URI	Description
POST	/RequestCenter/nsapi/transaction/requisitions	Delete the tenant quota when the consumed quota is 0. When Delete Tenant Quota is called, the Maximum tenant quota is added to the Available account region quota and the Maximum tenant quota is subtracted from the consumed account region quota.

Request Header

```
username: <nsapi_username>, password: <password>,
Content-Type: application/json, Accept: application/json
```

Request Payload Example

```
{
    "requisition": {
        "services": [{
            "name": "Delete Quota",
            "dictionaries": [{
                "name": "TenantQuota",
                "data": [{
                    "Account": "Microsoft",
                    "Service": "IAAS",
                    "Region": "US-RDU-2",
                    "Tenant": "HR",
                    "Metric": "RAM",
                    "Unit": "GB"
                }]
            }, {
                "name": "QuotaType",
                "data": {
                    "Type": "Tenant"
                }
            }]
        }]
    }
}
```

Request Parameters

Parameter	Description
Account	Required. The name of the account.

Parameter	Description
Service	Required. The type of service.
Region	Required. The geographical location where the account is available.
Tenant	Required. The name of the tenant.
Metric	Required. The following metrics are supported: • CPU • Memory • FloatingIP Be sure to use the appropriate unit of measure for the metrics.
Unit	Required. The measurement used for the metric. • Quantity is used for CPU and FloatingIP. • GB is used for Memory.
Maximum	Required. The maximum quota value.

Response Example


```
{
    "RequisitionSubmit": {
        "id": 6382,
        "customer": "nsapi@cisco.com nsapi@cisco.com",
        "initiator": "nsapi@cisco.com nsapi@cisco.com",
        "dueDateRaw": 1433409983017,
        "dueDate": "06/04/2015 9:26 AM",
        "startedDateRaw": 1433409982967,
        "startedDate": "06/04/2015 9:26 AM",
        "status": "Ongoing"
    }
}
```


Response Parameters

- The date/time format and time zone are based on the preferences set in the login user's personal profile.
- The due date returned in the API response is available only when the asynchronous submission global setting is enabled.

- Formatted dates returned in the API responses are based on the API user's timezone. Raw dates returned are in UTC.

Get a Tenant Quota Based on a Region

Method	URI	Description
GET	/RequestCenter/nsapi/serviceitem/namedquery/id/37db3ba3-3ac4-4344-83c3-acef86af9d15?account=<account> &tenant=<tenant>&region=<region>	Return the maximum consumed and available quota for a tenant based on the specific region. If Tenant is ParentType then this tenant is a sub-tenant.

Request Header

```
username: <nsapi_username>, password: <password>,
Content-Type: application/json, Accept: application/json
```

Request Example

```
/RequestCenter/nsapi/serviceitem/namedquery/id/37db3ba3-3ac4-4344-83c3-
acef86af9d15?account=Microsoft&tenant=Store Operations&region=US-TEXAS-2
```

Request Parameters

None

Response Example


```
{
    "List": [{
        "Available": "10.00000",
        "Service": "SAPHANA",
        "Region": "US-TEXAS-2",
        "ParentType": "Account",
        "Metric": "CPU",
        "Maximum": "10.00000",
        "Account": "Microsoft",
        "Consumed": "0.00000",
        "ParentID": "Microsoft",
        "Unit": "Quantity",
        "Tenant": "Store Operations"
    }]
}
```

Get a Tenant Quota Based on Service

Method	URI	Description
GET	/RequestCenter/nsapi/serviceitem/namedquery/id/4c432711-0a4e-414a-8cc1-a8984b41e963?account=<account>&tenant=<tenant>&service=<service>	Return the maximum consumed and available quota for a tenant based on the specific service. If Tenant is ParentType then this tenant is a sub-tenant.

Request Header

```
username: <nsapi_username>, password: <password>,
Content-Type: application/json, Accept: application/json
```

Request Example

```
/RequestCenter/nsapi/serviceitem/namedquery/id/4c432711-0a4e-414a-8cc1-
a8984b41e963?account=Microsoft&tenant=HR&service=IAAS
```

Request Parameters

None

Response Example


```
{
    "List": [{
        "Available": "30.00000",
        "Service": "IAAS",
        "Region": "US-RDU-2",
        "ParentType": "Account",
        "Metric": "CPU",
        "Maximum": "30.00000",
        "Account": "Microsoft",
        "Consumed": "0.00000",
        "ParentID": "Microsoft",
        "Unit": "Quantity",
        "Tenant": "HR"
    }]
}
```

Get a Tenant Quota Based on Service and Region

Method	URI	Description
GET	/RequestCenter/nsapi/serviceitem/namedquery/id/bfcb9bad-9eb1-4065-bd86-2b9a2fbbd22c?account=<account> &tenant=<tenant>&service=<service>&region=<region>	Return the maximum consumed and available quota for a tenant based on the specific service and region. If Tenant is ParentType then this tenant is a sub-tenant.

Request Header

```
username: <nsapi_username>, password: <password>,
Content-Type: application/json, Accept: application/json
```

Request Example

```
/RequestCenter/nsapi/serviceitem/namedquery/id/bfcb9bad-9eb1-4065-bd86-
2b9a2fbbd22c?account=Microsoft&tenant=Store Operations&service=SAPHANA&region=US-TEXAS-2
```

Request Parameters

None

Response Example


```
{
    "List": [{
        "Available": "10.00000",
        "Service": "SAPHANA",
        "Region": "US-TEXAS-2",
        "ParentType": "Account",
        "Metric": "CPU",
        "Maximum": "10.00000",
        "Account": "Microsoft",
        "Consumed": "0.00000",
        "ParentID": "Microsoft",
        "Unit": "Quantity",
        "Tenant": "Store Operations"
    }]
}
```

Get All Tenants Quota

Method	URI	Description
GET	/RequestCenter/nsapi/serviceitem/namedquery/id/b03bec42-2b49-4cc1-af92-2b65815b9216	Return the maximum consumed and available quota for all tenants. If Tenant is ParentType then this tenant is a sub-tenant. To return quota for a specific tenant, see Get Quota for a Tenant.

Request Header

```
username: <nsapi_username>, password: <password>,
Content-Type: application/json, Accept: application/json
```

Request Example

```
/RequestCenter/nsapi/serviceitem/namedquery/id/b03bec42-2b49-4cc1-af92-2b65815b9216
```

Request Parameters

None

Response Example


```
{
    "List": [{
        "Available": "30.00000",
        "Service": "IAAS",
        "Region": "US-RDU-2",
        "ParentType": "Account",
        "Metric": "CPU",
        "Maximum": "30.00000",
        "Account": "Microsoft",
        "Consumed": "0.00000",
        "ParentID": "Microsoft",
        "Unit": "Quantity",
        "Tenant": "HR"
    }, {
        "Available": "10.00000",
        "Service": "SAPHANA",
        "Region": "US-TEXAS-2",
        "ParentType": "Account",
        "Metric": "CPU",
        "Maximum": "10.00000",
        "Account": "Microsoft",
        "Consumed": "0.00000",
        "ParentID": "Microsoft",
        "Unit": "Quantity",
        "Tenant": "Store Operations"
    }, {
        "Available": "10.00000",
        "Service": "IAAS",
        "Region": "US-RDU-2",
        "ParentType": "Account",
        "Metric": "FloatingIP",
        "Maximum": "10.00000",
        "Account": "Toyota",
        "Consumed": "0.00000",
        "ParentID": "Toyota",
        "Unit": "Quantity",
        "Tenant": "Marketing"
    }]
}
```


Set Maximum Tenant Quota Pool

Method	URI	Description
POST	/RequestCenter/nsapi/transaction/requisitions	Increase the Maximum quota for a tenant. Increase and/or decrease the tenant Maximum quota. When this API is called, the tenant quota is updated

Method	URI	Description
		according to Account quota Available. To see the maximum quota and how much quota is consumed, call Get Specific Tenant Quota.

Request Header

```
username: <nsapi_username>, password: <password>,
Content-Type: application/json, Accept: application/json
```

Request Payload Example

```
{
    "requisition": {
        "services": [{
            "name": "Update Quota Pool",
            "dictionaries": [{
                "name": "TenantQuota",
                "data": [{
                    "Account": "Microsoft",
                    "Service": "IAAS",
                    "Region": "US-RDU-1",
                    "Tenant": "HR",
                    "Metric": "CPU",
                    "Unit": "Quantity"
                }]
            }, {
                "name": "QuotaType",
                "data": {
                    "Type": "Tenant",
                    "SetMaximum": 10
                }
            }]
        }]
    }
}
```

Request Parameters

Parameter	Description
Account	Required. The name of the account.
Service	Required. The type of service.
Region	Required. The geographical location where the account is available.
Metric	Required. The following metrics are supported:

Parameter	Description
	• CPU • Memory • FloatingIP **Note**: Be sure to use the appropriate unit of measure for the metrics.
Unit	Required. The measurement used for the metrics. • Quantity is used for CPU and FloatingIP. • GB is used for Memory. **Note**: The maximum value must be less than or equal to the available quota of the account; otherwise, tenant quota cannot be increased.
Type	Required. The type of quota. Options are: • Account • Tenant
SetMaximum	Required. The value to be increased according to the Account Available quota.

Update a Tenant Quota

Method	URI	Description
POST	/RequestCenter/nsapi/transaction/requisitions	Update a tenant quota using one of these attributes: Consume and Release.

Request Header

```
username: <nsapi_username>, password: <password>,
Content-Type: application/json, Accept: application/json
```

Request Payload Example

```
{
    "requisition": {
        "services": [{
            "name": "Update Quota",
            "dictionaries": [{
                "name": "TenantQuota",
                "data": [{
                    "Account": "Toyota",
                    "Service": "IAAS",
                    "Tenant": "HR",
                    "Region": "US-RDU-2",
                    "Metric": "CPU",
                    "Unit": "Quantity"
                }]
            }, {
                "name": "QuotaType",
                "data": {
                    "Type": "Tenant",
                    "Consume": 10
                }
            }]
        }]
    }
}
```

Request Parameters

Parameter	Description
Account	Required. The name of the account
Service	Required. The type of service.
Tenant	Required. The tenant identifier of the account.

Parameter	Description
Region	Required. The geographical location where the account is available.
Metric	Required. The following metrics are supported: • CPU • Memory • FloatingIP **Note**: Be sure to use the appropriate unit of measure for the metrics.
Unit	Required. The measurement used for the metrics. • Quantity is used for CPU and FloatingIP. • GB is used for Memory.
Type	Required. The type of quota. Options are: • Account • Tenant
Consume or Release	Required. The value to be consumed or released based on the attribute.

Response Example


```
{
    "RequisitionSubmit": {
        "id": 6382,
        "customer": "nsapi@cisco.com nsapi@cisco.com",
        "initiator": "nsapi@cisco.com nsapi@cisco.com",
        "dueDateRaw": 1433409983017,
        "dueDate": "06/04/2015 9:26 AM",
        "startedDateRaw": 1433409982967,
        "startedDate": "06/04/2015 9:26 AM",
        "status": "Ongoing"
    }
}
```


Response Parameters

- The date/time format and time zone are based on the preferences set in the login user's personal profile.
- The due date returned in the API response is available only when the asynchronous submission global setting is enabled.

- Formatted dates returned in the API responses are based on the API user's timezone. Raw dates returned are in UTC.

Check Requisition Status

Method	URI	Description
GET	/RequestCenter/nsapi/serviceitem/SiQuotaRequisitionStatus /RequisitionID=<requisition_id>	Check the requisition status for requests using the "id" from the response. See request example.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Example

```
{
    "RequisitionSubmit": {
        "id": 23,
        "customer": "API User",
        "initiator": "API User",
        "dueDateRaw": 1399047699577,
        "dueDate": "05/02/2014 9:21 AM",
        "startedDateRaw": 1399047303153,
        "startedDate": "05/02/2014 9:15 AM",
        "status": "Ongoing"
    }
}
```

Managing Bulk Operations

Service Name	Type	URI
Associate Many users to Many Projects	POST	/services/v2/bulk/user/project
Disassociate Many Users from Many Projects	PUT	/services/v2/bulk/user/project
Disassociate Many Users from One Tenant	PUT	/services/v2/bulk/user/tenant
Get Bulk User Details with Filters	GET	/services/v2/bulk/serviceProvider/<<SP>>/user?account=<<account>>&index=<<pagination index>> &items=<<item count in page>>&orderBy=<<role|email|user_uid>> &user_uid=<<user uid>>&role=<<User|Administrator>>&email=<<email>>
Get Bulk User-Tenant Disassociation Status	GET	/services/v2/status/bulk/user/tenant/reqId/<<redId>>?reportType=<<reportType>>

Associate Many Users to Many Projects

Method	URI	Description
POST	/services/v2/bulk/user/project	Associate many users to many projects.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Example

```
[
    {
        "users": ["user1","user2"],
        "projects":["e0ae7cb6f5b14e61bc1933f0034a1f06",
                    "4e9cf543b8fc4631829ab7c5210a2132"],
        "role":"User"
    }
]
```

Response Example


```
{
  "RequisitionSubmit": {
    "id": 11055,
    "customer": "nsapi@cisco.com nsapi@cisco.com",
    "initiator": "nsapi@cisco.com nsapi@cisco.com",
    "dueDateRaw": 1436222128257,
    "dueDate": "07/06/2015 10:35 PM",
    "startedDateRaw": 1436222128210,
    "startedDate": "07/06/2015 10:35 PM",
    "status": "Ongoing"
  }
}
```

Disassociate Many Users from Many Projects

Method	URI	Description
PUT	/services/v2/bulk/user/project	Disassociate many users from many projects.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Example

```
[
    {
        "users": ["user1","user2"],
        "projects":["e0ae7cb6f5b14e61bc1933f0034a1f06",
                    "4e9cf543b8fc4631829ab7c5210a2132"],
        "role":"User"
    }
]
```

Response Example


```
{
  "RequisitionSubmit": {
    "id": 11062,
    "customer": "nsapi@cisco.com nsapi@cisco.com",
    "initiator": "nsapi@cisco.com nsapi@cisco.com",
    "dueDateRaw": 1436222587947,
    "dueDate": "07/06/2015 10:43 PM",
    "startedDateRaw": 1436222587893,
    "startedDate": "07/06/2015 10:43 PM",
    "status": "Ongoing"
  }
}
```

Disassociate Many Users from One Tenant

Method	URI	Description
PUT	/services/v2/bulk/user/tenant	Disassociate many users from a specific tenant.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Example

```
[
   {
        "ccs_tenant":"tenant1",
        "users": ["user3","user4"]

   }
]
```

Response Example


```
{
  "RequisitionSubmit": {
    "id": 12830,
    "customer": "nsapi@cisco.com nsapi@cisco.com",
    "initiator": "nsapi@cisco.com nsapi@cisco.com",
    "dueDateRaw": 1436880590987,
    "dueDate": "07/14/2015 1:29 PM",
    "startedDateRaw": 1436880590927,
    "startedDate": "07/14/2015 1:29 PM",
    "status": "Ongoing"
  }
}
```

Get Bulk User Details with Filters

Method	URI	Description
GET	/services/v2/bulk/serviceProvider/<<SP>>/user?account=<<account>> &index=<<pagination index>> &items=<<item count in page>> &orderBy=<<role|email|user_uid>> &user_uid=<<user uid>> &role=<<User|Administrator>> &email=<<email>> &countOnly=True	Return the user details in bulk based on the filter specified. To return only the count of users, simply specify account and countOnly.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Parameters

Parameter	Description
account	Required. This is the ccs_tenant field.
index	Required. pagination index (will be 2 if you want to show the users who will appear on page 2 of your results)
items	Required. The number of items you want to show in each page.
orderedBy	Required. The field you want your results to be ordered by.
user_uid	Optional. The identifier of the user.
role	Optional. Options are: • User • Administrator
email	Optional. The user email address.
CountOnly	Required. Options are:

Parameter	Description
	• true to return the number of users matching the criteria. • false

Response Example

The example shows the user information fields that are available for all users matching the search criteria.


```
[
  {
    "email": "abc@cisco.com",
    "first_name": "AB",
    "last_name": "C",
    "ccs_tenant": "test-tenant",
    "user_uid": "abc",
    "role": "User",
    "status": "Active"
  },
  {
    "email": "abc11-123@cisco.com",
    "first_name": "First Name",
    "last_name": "Last Name",
    "ccs_tenant": "test-tenant",
    "user_uid": "abc11-123",
    "role": "User",
    "status": "Active"
  }
]
```

Get Bulk User-Tenant Disassociation Status

Method	URI	Description
GET	/services/v2/status/bulk/user/tenant/reqId/<<redId>>?reportType=<<reportType>>	Return the status of bulk user-tenant disassociation.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Parameters

Parameter	Description
reportType	Optional. Values are: • Aggregate (default) • Detail

Response Example for a Detailed Report Type


```
[
  {
    "Status": "Failed",
    "Message": "Status Code [404] - User does not exist",
    "UserName": "bulk-user3"
  },
  {
    "Status": "Failed",
    "Message": "Status Code [404] - User does not exist",
    "UserName": "bulk-user4"
  }
]
```

Response Example for an Aggregate Report Type


```
[
  {
    "Succeeded": "0",
    "Completed": "2",
    "Failed": "2",
    "Total": "2"
  }
]
```

Generate Reports

The Cisco Intercloud Services mediation system works on the concept of reports: UsageSummary, EventSummary, and LicenseReport. These reports are generated at request time for one or more accounts in the mediation system, and contain the usage values or events, relative to the requested date range, for all the resources being tracked for tenants, projects, and instances. All transactions and report values use UTC as the time zone.

Service Name	Type	URI
Generate a Report for All Tenants	POST	https://<CCSMediationFQDN>/rest/v2/reports
Generate a Report for a Specific Tenant	POST	https://<CCSMediationFQDN>/rest/v2/reports
Generate a Report for a Specific Tenant Project	POST	https://<CCSMediationFQDN>/rest/v2/reports
Generate a Report for a Specific Project Instance	POST	https://<CCSMediationFQDN>/rest/v2/reports

Generate a Report for All Tenants

Method	URI
POST	https://<CCSMediationFQDN>/rest/v2/reports

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Example

```
{
   <reportName>UsageSummary</reportName>
   <userGroupName><federationpartner>Billing</userGroupName>
   <format>CSV</format>
   <dateRange>CUSTOM</dateRange>
   <accountLevel>1</accountLevel>
   <customStartDate year="2015" month="9" day="9"/>
   <customStartDate year="2015" month="10" day="9"/>
}
```

Request Parameters

Parameter	Description
reportName	Required. The name of the report to generate. Three reports are available: • UsageSummary reports usage information. • EventSummary reports event information. • LicenseReport provides operating system information.
userGroupName	Required. The group to which the calling user belongs.
format	Optional. The format of the report to be returned. The default value (PDF) is used if no value is specified. Supported values are:

Parameter	Description
	• PDF (default) • CSV • XLS
dateRange	Optional. The period for which the report to be generated. The default value (PREVPERIOD) is used if no value is specified. Supported values are: • ALL for all dates • CURRPERIOD for the current period • PREVPERIOD (default) for the previous period • CURRMON for the current month • MONTHTODATE for the month to the current date • PREVMON for the previous month • CURRYEAR for the current year • YEARTODATE for the year to the current date • PREVYEAR for the previous year • CURRWEEK for the current week • WEEKTODATE for the week to the current date • PREVWEEK for the previous week • CUSTOM for a specific date range Month and year account for the 1st day to the last day of the month or year respectively. Periods are used if billing cycles fall between those boundaries, for example, 15th of the month. The period used for a particular tenant is a server configuration.
accountLevel	Optional. The account level at which to run the report. By default, reports run at account level 1 (highest). Supported values are: • 1 for Tenant ID • 2 for Project UUID • 3 for Instance UUID
accountRangeStart	Optional. The start of the account range or a comma-separated list of accounts to report. The default is **lowest** – the lowest account ID allowed for this particular user group. Specifying the **lowest** for accountRangeStart and **highest** for accountRangeEnd provides usage information for all accounts

Parameter	Description
	available in the user group.
accountRangeEnd	Optional. The end of the account range or a comma-separated list of accounts to report. The default is **highest** – the highest account ID allowed for this particular user group. Specifying the **lowest** for accountRangeStart and **highest** for accountRangeEnd provides usage information for all accounts available in the user group.
customStartDate	Required if **CUSTOM** is specified for dateRange. The date uses this format, for example, <customStartDate year="2015" month="1" day="1"/>
customEndDate	Required if **CUSTOM** is specified for dateRange. The date uses this format, for example, <customEndDate year="2015" month="1" day="1"/>

Response Header Example

```
reply: 'HTTP/1.1 200 OK\r\n'
header: Server: Apache-Coyote/1.1
header: Content-Disposition: attachment; filename="UsageSummary.csv"
header: Content-Type: text/csv
header: Content-Length: 181320
header: Date: Wed, 06 May 2015 19:23:43 GMT
```

Generate a Report for a Specific Tenant

Method	URI
POST	https://<CCSMediationFQDN>/rest/v2/reports

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Example

```
{
   <reportName>UsageSummary</reportName>
   <userGroupName><federationpartner>Billing</userGroupName>
   <format>CSV</format>
   <dateRange>PREVMONTH</dateRange>
   <accountLevel>1</accountLevel>
   <accountRangeStart>182afa7a23d74e4bb3bc48024e2dfb40</accountRangeStart>
   <accountRangeEnd>182afa7a23d74e4bb3bc48024e2dfb40</accountRangeEnd>
}
```

Request Parameters

Parameter	Description
reportName	Required. The name of the report to generate. Three reports are available: • UsageSummary reports usage information. • EventSummary reports event information. • LicenseReport provides operating system information.
userGroupName	Required. The group to which the calling user belongs.
format	Optional. The format of the report to be returned. The default value (PDF) is used if no value is specified. Supported values are:

Parameter	Description
	• PDF (default) • CSV • XLS
dateRange	Optional. The period for which the report to be generated. The default value (PREVPERIOD) is used if no value is specified. Supported values are: • ALL for all dates • CURRPERIOD for the current period • PREVPERIOD (default) for the previous period • CURRMON for the current month • MONTHTODATE for the month to the current date • PREVMON for the previous month • CURRYEAR for the current year • YEARTODATE for the year to the current date • PREVYEAR for the previous year • CURRWEEK for the current week • WEEKTODATE for the week to the current date • PREVWEEK for the previous week • CUSTOM for a specific date range  Month and year account for the 1st day to the last day of the month or year respectively. Periods are used if billing cycles fall between those boundaries, for example, 15th of the month. The period used for a particular tenant is a server configuration.
accountLevel	Optional. The account level at which to run the report. By default, reports run at account level 1 (highest). Supported values are: • 1 for Tenant ID • 2 for Project UUID • 3 for Instance UUID
accountRangeStart	Required. This is the tenant ID whose report you want to obtain.
accountRangeEnd	Required. This is the tenant ID whose report you want to obtain.
customStartDate	Required if **CUSTOM** is specified for dateRange. The date uses this format, for

Parameter	Description
	example, <customStartDate year="2015" month="1" day="1"/>
customEndDate	Required if **CUSTOM** is specified for dateRange. The date uses this format, for example, <customEndDate year="2015" month="1" day="1"/>

Response Header Example

```
reply: 'HTTP/1.1 200 OK\r\n'
header: Server: Apache-Coyote/1.1
header: Content-Disposition: attachment; filename="UsageSummary.csv"
header: Content-Type: text/csv
header: Content-Length: 181320
header: Date: Wed, 06 May 2015 19:23:43 GMT
```

Generate a Report for a Specific Tenant Project

Method	URI
POST	https://<CCSMediationFQDN>/rest/v2/reports

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Example

```
{
   <reportName>UsageSummary</reportName>
   <userGroupName><federationpartner>Billing</userGroupName>
   <format>CSV</format>
   <dateRange>CURRMONTH</dateRange>
   <accountLevel>2</accountLevel>
   <accountRangeStart>182afa7a23d74e4bb3bc48024e2dfb40:3115d47b16a047ea8ee104cee4f3020f</accountRangeStart>
   <accountRangeEnd>182afa7a23d74e4bb3bc48024e2dfb40:3115d47b16a047ea8ee104cee4f3020f</accountRangeEnd>
}
```

Request Parameters

Parameter	Description
reportName	Required. The name of the report to generate. Three reports are available: • UsageSummary reports usage information. • EventSummary reports event information. • LicenseReport provides operating system information.
userGroupName	Required. The group to which the calling user belongs.
format	Optional. The format of the report to be returned. The default value (PDF) is used if no value is specified. Supported values are:

Parameter	Description
	• PDF (default) • CSV • XLS
dateRange	Optional. The period for which the report to be generated. The default value (PREVPERIOD) is used if no value is specified. Supported values are: • ALL for all dates • CURRPERIOD for the current period • PREVPERIOD (default) for the previous period • CURRMON for the current month • MONTHTODATE for the month to the current date • PREVMON for the previous month • CURRYEAR for the current year • YEARTODATE for the year to the current date • PREVYEAR for the previous year • CURRWEEK for the current week • WEEKTODATE for the week to the current date • PREVWEEK for the previous week • CUSTOM for a specific date range  Month and year account for the 1st day to the last day of the month or year respectively. Periods are used if billing cycles fall between those boundaries, for example, 15th of the month. The period used for a particular tenant is a server configuration.
accountLevel	Optional. The account level at which to run the report. By default, reports run at account level 1 (highest). Supported values are: • 1 for Tenant ID • 2 for Project UUID • 3 for Instance UUID
accountRangeStart	Required. This is the tenant ID whose specific project report you want to obtain. Separate the tenant ID and project ID by a colon (:) delimiter, for example, 182afa7a23d74e4bb3bc48024e2dfb40:3115d47b16a047ea8ee104cee4f3020f

Parameter	Description
accountRangeEnd	Required. This is the tenant ID whose specific project report you want to obtain. Separate the tenant ID and project ID by a colon (:) delimiter, for example, 182afa7a23d74e4bb3bc48024e2dfb40:3115d47b16a047ea8ee104cee4f3020f
customStartDate	Required if **CUSTOM** is specified for dateRange. The date uses this format, for example, <customStartDate year="2015" month="1" day="1"/>
customEndDate	Required if **CUSTOM** is specified for dateRange. The date uses this format, for example, <customEndDate year="2015" month="1" day="1"/>

Response Header Example

```
reply: 'HTTP/1.1 200 OK\r\n'
header: Server: Apache-Coyote/1.1
header: Content-Disposition: attachment; filename="UsageSummary.csv"
header: Content-Type: text/csv
header: Content-Length: 181320
header: Date: Wed, 06 May 2015 19:23:43 GMT
```

Generate a Report for a Specific Project Instance

Method	URI
POST	https://<CCSMediationFQDN>/rest/v2/reports

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Example

```
{
   <reportName>UsageSummary</reportName>
   <userGroupName><federationpartner>Billing</userGroupName>
   <format>CSV</format>
   <dateRange>PREVMONTH</dateRange>
   <accountLevel>3</accountLevel>
   <accountRangeStart>182afa7a23d74e4bb3bc48024e2dfb40:3115d47b16a047ea8ee104cee4f3020f:51abd10e-2949-4852-9241-1e88f93f84db</accountRangeStart>
   <accountRangeEnd>182afa7a23d74e4bb3bc48024e2dfb40:3115d47b16a047ea8ee104cee4f3020f:51abd10e-2949-4852-9241-1e88f93f84db</accountRangeEnd>
}
```

Request Parameters

Parameter	Description
reportName	Required. The name of the report to generate. Three reports are available: • UsageSummary reports usage information. • EventSummary reports event information. • LicenseReport provides operating system information.
userGroupName	Required. The group to which the calling user belongs.
format	Optional. The format of the report to be returned. The default value (PDF) is used if no

Parameter	Description
	value is specified. Supported values are: • PDF (default) • CSV • XLS
dateRange	Optional. The period for which the report to be generated. The default value (PREVPERIOD) is used if no value is specified. Supported values are: • ALL for all dates • CURRPERIOD for the current period • PREVPERIOD (default) for the previous period • CURRMON for the current month • MONTHTODATE for the month to the current date • PREVMON for the previous month • CURRYEAR for the current year • YEARTODATE for the year to the current date • PREVYEAR for the previous year • CURRWEEK for the current week • WEEKTODATE for the week to the current date • PREVWEEK for the previous week • CUSTOM for a specific date range Month and year account for the 1st day to the last day of the month or year respectively. Periods are used if billing cycles fall between those boundaries, for example, 15th of the month. The period used for a particular tenant is a server configuration.
accountLevel	Optional. The account level at which to run the report. By default, reports run at account level 1 (highest). Supported values are: • 1 for Tenant ID • 2 for Project UUID • 3 for Instance UUID
accountRangeStart	Required. This is the tenant's project whose specific instance report you want to obtain. Separate the tenant ID, project ID, and instance ID by a colon (:) delimiter, for example, 182afa7a23d74e4bb3bc48024e2dfb40:3115d47b16a047ea8ee104cee4f3020f:51abd10e-

Parameter	Description
	2949-4852-9241-1e88f93f84db
accountRangeEnd	Required. This is the tenant's project whose specific instance report you want to obtain. Separate the tenant ID, project ID, and instance ID by a colon (:) delimiter, for example, 182afa7a23d74e4bb3bc48024e2dfb40:3115d47b16a047ea8ee104cee4f3020f:51abd10e-2949-4852-9241-1e88f93f84db
customStartDate	Required if **CUSTOM** is specified for dateRange. The date uses this format, for example, <customStartDate year="2015" month="1" day="1"/>
customEndDate	Required if **CUSTOM** is specified for dateRange. The date uses this format, for example, <customEndDate year="2015" month="1" day="1"/>

Response Header Example

```
reply: 'HTTP/1.1 200 OK\r\n'
header: Server: Apache-Coyote/1.1
header: Content-Disposition: attachment; filename="UsageSummary.csv"
header: Content-Type: text/csv
header: Content-Length: 181320
header: Date: Wed, 06 May 2015 19:23:43 GMT
```

Get Service Requests Status

Method	URI	Description
GET	/services/reqId/<<id>>	Return CISservice status.

Request Header

```
Authorization: Basic <<Base64 encoded (username:password)>>
Accept: application/json
Content-Type: application/json
```

Request Parameters

Parameter	Description
id	A required parameter.

Response Example


```
{
    "requisition":
    {
       "tenantId": 1,
       "userId": 5,
       "ownerId": 5,
       "serviceId": 2,
       "customerId": 5,
       "expectedDuration": 0,
       "actualDuration": 0,
       "startedDate": "03/05/2014 7:30 AM",
       "dueDate": "05/05/2014 4:06 PM",
       "expectedCost": 0,
       "status": "Ongoing",
       "requisitionId": 4,
       "flagImage": "/RequestCenter/images/flaglate.gif",
       "lateFlag": true,
       "customerName": "API User",
       "organizationalUnitName": "OU",
       "submitDate": "03/05/2014 7:30 AM",
       "statusId": 1,
       "serviceName": "Create IaaS Project",
       "ownerName": "API User",
       "organizationalUnitId": 2,
       "startedDateRaw": 1399102230937,
       "dueDateRaw": 1399305996000,
       "submitDateRaw": 1399102231210,
       "requisitionURL": "4",
       "requisitionURLOnly": "/RequestCenter/myservices/navigate.do?reqid=4",
       "milestoneLink": " ",
       "percentageCompleted": 0
    }
}
```