

Links to Other API pages: [Cisco Unity Connection APIs](#)

CUPI Guide Contents
API Overview
Index of All CUPI Documentation
Schedule Example

Contents

- [1 Tenant API](#)
 - ◆ [1.1 Listing the Tenants](#)
 - ◆ [1.2 Viewing the Specific Tenant](#)
 - ◆ [1.3 Creating a New Tenant](#)
 - ◆ [1.4 Delete the Tenant](#)
 - ◆ [1.5 Updating the Tenant](#)
 - ◆ [1.6 Viewing COSes associated with the Tenant](#)
 - ◆ [1.7 Viewing Partitions Associated with the Tenant](#)
 - ◆ [1.8 Viewing Phone Systems Associated with the Tenant](#)
 - ◆ [1.9 Viewing Schedule Sets Associated with the Tenant](#)
 - ◆ [1.10 Viewing Operator Associated with the Tenant](#)
 - ◆ [1.11 Explanation of Data Fields](#)

Tenant API

The term "tenant", depicts a small voicemail setup for a sub-company within a large Unity Connection server. In other words, if Cisco Unity Connection server is being shared by N (N=2, 3, 4?) companies for voicemail service, each company can setup as a separate "tenant" in the installation that effectively isolate them from one another. The key aspects to a tenant is that, each one is given their own set of routing rules, default operator, opening greeting call handlers, operator subscriber, schedule, switch assignment, partition, and search space along with a few other things. The idea is to map the switch assignment for that "tenant" to some ports on your phone system and inbound calls are routed to users and handlers within that tenant. Once a call is in that tenant's partition they cannot access handlers or users outside of items assigned to that tenant. Administrator can use this API to create/update/delete/fetch the tenants. You can update various attributes of a tenant using this API.

Listing the Tenants

The following is an example of the GET request that fetch the tenants:

```
GET https://<connection-server>/vmrest/tenants
```

The following is the response from the above *GET* request and the actual response will depend upon the information given by you:

```
<Tenants total="1">
  <Tenant>
    <URI>/vmrest/tenants/069b8e41-8051-4eee-b653-1854f7e68cd8</URI>
```

Cisco_Unity_Connection_Provisioning_Interface_(CUPI)_API_--_Tenant_API

```
<ObjectId>069b8e41-8051-4eee-b653-1854f7e68cd8</ObjectId>
<CreationDate>2013-01-12 14:57:32.569</CreationDate>
<Alias>Taxoma</Alias>
<SmtpDomain>Taxoma.com</SmtpDomain>
<TimeZone>175</TimeZone>
<Language>1033</Language>
<CosesURI>/vmrest/tenants/069b8e41-8051-4eee-b653-1854f7e68cd8/coses</CosesURI>
<ScheduleSetsURI>/vmrest/tenants/069b8e41-8051-4eee-b653-1854f7e68cd8/schedulesets</ScheduleSetsURI>
<PhoneSystemsURI>/vmrest/tenants/069b8e41-8051-4eee-b653-1854f7e68cd8/phonesystems</PhoneSystemsURI>
<OperatorsURI>/vmrest/tenants/069b8e41-8051-4eee-b653-1854f7e68cd8/operators</OperatorsURI>
<PartitionsURI>/vmrest/tenants/069b8e41-8051-4eee-b653-1854f7e68cd8/partitions</PartitionsURI>
<PartitionObjectId>a28e144f-a818-4c8b-a80f-09ecc96872cd</PartitionObjectId><PhoneSystemObject
09ecc96872cd</PhoneSystemObjectId>
  </Tenant>
</Tenants>
```

Response Code: 200

JSON Example

To list the tenants, do the following:

Request URI:

```
GET https://<connection-server>/vmrest/tenants
Accept: applicaiton/json
Connection: keep_alive
```

The following is the response from the above *GET* request and the actual response will depend upon the information given by you:

```
{
  "@total": "1",
  "Tenant": [
    {
      "URI": "/vmrest/tenants/e7ffe9ad-2ea4-492b-b65e-fdb31aa242ec",
      "ObjectId": "e7ffe9ad-2ea4-492b-b65e-fdb31aa242ec",
      "CreationDate": "2013-02-13 10:10:41.615",
      "Alias": "Taxoma",
      "SmtpDomain": "Texoma.com",
      "TimeZone": "175",
      "Language": "1033",
      "CosesURI": "/vmrest/tenants/e7ffe9ad-2ea4-492b-b65e-fdb31aa242ec/coses",
      "ScheduleSetsURI": "/vmrest/tenants/e7ffe9ad-2ea4-492b-b65e-fdb31aa242ec/schedulesets",
      "PhoneSystemsURI": "/vmrest/tenants/e7ffe9ad-2ea4-492b-b65e-fdb31aa242ec/phonesystems",
      "OperatorsURI": "/vmrest/tenants/e7ffe9ad-2ea4-492b-b65e-fdb31aa242ec/operators",
      "PartitionsURI": "/vmrest/tenants/e7ffe9ad-2ea4-492b-b65e-fdb31aa242ec/partitions",
      "PhoneSystemObjectId": "79b262d2-2bcb-44c3-9251-b81e3f5e75e5",
      "PartitionObjectId": "30438c03-e8de-4584-bb5b-0dae05faf0af"
    }
  ]
}
```

Response Code: 200

Viewing the Specific Tenant

The following is an example of the GET request that lists the details of specific tenant represented by the provided value of tenant object ID:

Cisco_Unity_Connection_Provisioning_Interface_(CUPI)_API_--_Tenant_API

GET https://<connection-server>/vmrest/tenants/<Tenant-ObjectId>

The following is the response from the above *GET* request and the actual response will depend upon the information given by you:

```
<Tenant>
  <URI>/vmrest/tenants/069b8e41-8051-4eee-b653-1854f7e68cd8</URI>
  <ObjectId>069b8e41-8051-4eee-b653-1854f7e68cd8</ObjectId>
  <CreationDate>2013-01-12 14:57:32.569</CreationDate>
  <Alias>Taxoma</Alias>
  <SmtpDomain>Taxoma.com</SmtpDomain>
  <TimeZone>175</TimeZone>
  <Language>1033</Language>
  <CosesURI>/vmrest/tenants/069b8e41-8051-4eee-b653-1854f7e68cd8/coses</CosesURI>
  <ScheduleSetsURI>/vmrest/tenants/069b8e41-8051-4eee-b653-1854f7e68cd8/schedulesets</ScheduleSetsURI>
  <PhoneSystemsURI>/vmrest/tenants/069b8e41-8051-4eee-b653-1854f7e68cd8/phonesystems</PhoneSystemsURI>
  <OperatorsURI>/vmrest/tenants/069b8e41-8051-4eee-b653-1854f7e68cd8/operators</OperatorsURI>
  <PartitionsURI>/vmrest/tenants/069b8e41-8051-4eee-b653-1854f7e68cd8/partitions</PartitionsURI>
  <PartitionObjectId>a28e144f-a818-4c8b-a80f-09ecc96872cd</PartitionObjectId>
  <PhoneSystemObjectId>a28e144f-a818-4c8b-a80f-09ecc96872cd</PhoneSystemObjectId>
</Tenant>
```

Response Code: 200

JSON Example

To view a specific tenant, do the following:

Request URI:

```
GET https://<connection-server>/vmrest/tenants/<Tenant-ObjectId>
Accept: application/json
Connection: keep_alive
```

The following is the response from the above *GET* request and the actual response will depend upon the information given by you:

```
{
  "URI": "/vmrest/tenants/75058a36-5eed-49b5-97bd-7267dab09e83",
  "ObjectId": "75058a36-5eed-49b5-97bd-7267dab09e83",
  "CreationDate": "2013-02-15 08:05:22.178",
  "Description": "Hello",
  "Alias": "Taxoma21",
  "SmtpDomain": "Texoma21.com",
  "TimeZone": "175",
  "Language": "1033",
  "CosesURI": "/vmrest/tenants/75058a36-5eed-49b5-97bd-7267dab09e83/coses",
  "ScheduleSetsURI": "/vmrest/tenants/75058a36-5eed-49b5-97bd-7267dab09e83/schedulesets",
  "PhoneSystemsURI": "/vmrest/tenants/75058a36-5eed-49b5-97bd-7267dab09e83/phonesystems",
  "OperatorsURI": "/vmrest/tenants/e7ffe9ad-2ea4-492b-b65e-fdb31aa242ec/operators",
  "PartitionsURI": "/vmrest/tenants/75058a36-5eed-49b5-97bd-7267dab09e83/partitions",
  "PhoneSystemObjectId": "e815916c-abd5-49be-a33d-0495d1f4e080",
  "PartitionObjectId": "1469a460-e3d2-42c1-aad8-8f89e2c56ba8"
}
```

Response Code: 200

Creating a New Tenant

We have recommended naming conventions for the objects to be created for Tenant. These naming conventions are not mandatory, but recommended to make configurations clear and unambiguous.

See the table below for object names for a tenant with the alias as Texoma: The following chart lists all of the data fields:

Object Name	Convention
Partition	Texoma_Partition_1
Search Space	Texoma_SearchSpace_1
Phone System	Texoma_PhoneSystem_1
Direct Routing Rule	Texoma_DRAAttemptSignIn_1
Forward Routing Rule	Texoma_FRAAttemptFwd_1
Direct/Fwd Routing Rule	Texoma_RROpeningGreeting_1
COS	Texoma_COS_1
User Template	Texoma_UserTemplate_1
Call Handler Template	Texoma_SystemCallhandlerTemplate_1
System Call Handler	Texoma_GoodbyeCH_1
System Call Handler	Texoma_OperatorCH_1
System Call Handler	Texoma_OpeningGreetingCH_1
Directory Handler	Texoma_SystemDirectoryHandler_1
Interview Handler	Texoma_Interviewer_1
Default Operator User	Texoma_Operator_1
Schedule	Texoma_Weekdays_1
Holiday Schedule	Texoma_Holiday_1
Users Distribution List	Texoma_allvoicemailusersDL_1
Contacts Distribution List	Texoma_allvoicemailenabledcontactsDL_1

Note: It is mandatory to specify the Alias and SMTP Domain fields in the request body while creating a tenant and it is optional to specify the Mailbox Store and Description fields. Here timezone of the tenants and objects created along with tenant creation and will have the default TimeZone(System specific).

The following is an example of POST request that can be used to create a new tenant.

```
POST https://<connection-server>/vmrest/tenants
```

Request Body:

```
<Tenant>
  <Alias>Texoma</Alias>
  <SmtDomain>Texoma.com</SmtDomain>
```

</Tenant>

Response Code: 201
/vmrest/tenants/837e1068-7a27-4b97-8483-1747d5a40aec

JSON Example

To create a new tenant, do the following:

```
POST https://<connection-server>/vmrest/tenants
Accept: application/json
Content_type: application_json
Connection: keep_alive
```

Request Body:

```
{
  "Alias": "Taxoma21",
  "SmtDomain": "Texoma21.com",
}
```

The following is the response from the above *POST* request and the actual response will depend upon the information given by you:

Response Code: 201
/vmrest/tenants/837e1068-7a27-4b97-8483-1747d5a40aec

Use the request body below to specify a Mailbox Store and Description for the Tenant. *Response Body:*

```
<Tenant>
  <Alias>Taxoma</Alias>
  <SmtDomain>Texoma.com</SmtDomain>
  <MailboxStoreObjectId>837e1068-7a27-4b97-8483-1747d5a40aec</MailboxStoreObjectId>
  <Description>text</Description>
</Tenant>
```

The following is the response from the above *POST* request and the actual response will depend upon the information given by you:

Response Code: 201

JSON Example To create a new tenant with specific mailbox store and description: *Request URI:*

```
POST https://<connection-server>/vmrest/tenants
Accept: application/json
Content_type: applciation/json
Connection: keep_alive
```

Request Body:

```
{
  "Alias": "Taxoma21",
  "SmtDomain": "Texoma21.com",
  "MailboxStoreObjectId": "c472cbc0-3baa-487a-80e2-66a99aa615e2",
  "Description": "Test"
}
```

Response Code: 201

Cisco_Unity_Connection_Provisioning_Interface_(CUPI)_API_--_Tenant_API

Use the following request body to specify a timezone for the tenant:

Request URI:

`https://<connection-server>/vmrest/tenants`

We can fetch the valid timezone values using API: <https://<connection-server>/vmrest/timezones> and taking the values of TimeZoneId parameter from the GET response.

Request body to do the POST operation:

```
<Tenant>
  <Alias>Texoma</Alias>
  <Description>Tenant for Texoma Inc</Description>
  <SmtDomain>Texoma.com</SmtDomain>
  <MailboxStore>c472cbc0-3baa-487a-80e2-66a99aa615e2</ MailboxStore>
  <TimeZone>175</TimeZone>
  <Language>1035</Language>
</Tenant>
```

Response Code: 201
`/vmrest/tenants/837e1068-7a27-4b97-8483-1747d5a40aec`

JSON example to create a new tenant with TimeZone:

```
POST https://<connection-server>/vmrest/tenants
Accept: application/json
Content-type: applciation/json
Connection: keep-alive
```

Request Body:

```
{
  "Alias": "Taxoma",
  "SmtDomain": "Texoma.com",
  "MailboxStoreObjectId": "c472cbc0-3baa-487a-80e2-66a99aa615e2",
  "Description": "Test",
  "TimeZone": "175",
  "Language": "1035"
}
```

Response Status Code: 201
`/vmrest/tenants/837e1068-7a27-4b97-8483-1747d5a40aec`

Timezone and Language are optional fields. All the default objects created automatically during tenant creation like Operator, UserTemplate, Call Handlers, CallHandler Template will have the same timezone as specified in the tenant creation request. If timezone and language parameters are not given then system default will be used. All the default objects created automatically during tenant creation will have the system default TimeZone and Language.

If the request is successfully processed then the following objects will also be created:

COS	User Operator
Partition	System Call Handlers
Phone System	Directory Handler

Schedule Set	Interview Handler
Schedule	Search Space
Schedule Detail	Search Space Member
User Template	Routing Rules
Distribution List	Call Handler Template
Distribution List Membership	

In addition to this, if no MailboxStoreObjectId is specified in the request, then the system default Mailbox Store will be assigned to the Tenant and if a MailboxStoreObjectId is specified in the request then it will be used in tenant creation.

Delete the Tenant

This request can be used to delete a tenant and all of its associated COSes, Phone Systems, Partitions, Schedules and other objects.

```
DELETE https://<connection-server>/vmrest/tenants/<TenantObjectId>
```

Response Code: 204

JSON Example

To delete a tenant:

```
DELETE https://<connection-server>/vmrest/tenants/<TenantObjectId>
Accept: application/json
Connection: keep_alive
```

Response Code: 204

On successful deletion of a tenant record, all instances of the following objects associated with the tenant will also be deleted:

COS	User Operator
Partition	System Call Handlers
Phone System	Directory Handler
Schedule Set	Interview Handler
Schedule	Search Space
Schedule Detail	Search Space Member
User Template	Routing Rules
Distribution List	Call Handler Template
Distribution List Membership	

Updating the Tenant

Updating the Description of a Tenant The following is an example of the PUT request that can be used to modify the tenant where only description field can be updated:

```
PUT https://<connection-server>/vmrest/tenants/<TenantObjectId>
```

Request Body:

```
<Tenant>
  <Description>About the tenant</Description>
</Tenant>
```

Response Code: 204

JSON Example

To update the description of tenant, do the following:

Request URI:

```
PUT https://<connection-server>/vmrest/tenants/<TenantObjectId>
Accept: applicaiton/json
Content_type: application/json
Connection: keep_alive
```

Request Body:

```
{
  "Description": "Hello"
}
```

The following is the response from the above *PUT* request and the actual response will depend upon the information given by you:

Response Code: 204

Updating Smtpdomain Description of a Tenant

For a Tenant, **Smtpdomain Description** field can be updated. This request can be used to update the Smtpdomain field for a Tenant.

On changing SMTP domain for any tenant, the SMTP address for all the users, distribution lists, personal groups, and contacts are updated and the old SMTP Address is moved to SMTP proxy address.

The following is an example of the PUT request that can be used to modify the tenant where only smtpdomain Description field can be updated:

```
PUT https://<connection-server>/vmrest/tenants/<Tenant-ObjectId>
```

Request Body:

```
<Tenant>
  <Smtpdomain>Donald.com</Smtpdomain>
</Tenant>
```


Response Code: 204

JSON example

To update the description of tenant, do the following:

Request URI:

```
PUT https://<connection-server>/vmrest/tenants/<Tenant-ObjectId>
Accept: applicaiton/json
Content_type: application/json
Connection: keep_alive
```

Request Body:

```
{
  "Tenant": { "SmtDomain": "Donald.com" }
}
```

Response Code: 204

Note: Make sure that the SMTP Domain Change for the Cisco Unity Connection is not affecting the SMTP Domain for users, user templates, distribution lists and other objects belonging to the existing tenants

Update PilotNumber and MediaSwitchObjectId of Tenant

The following is an example of the PUT request that can be used to modify the PilotNumber and MediaSwitchObjectId of Tenant:

```
PUT https://<connection-server>/vmrest/tenants/<Tenant-ObjectId>
```

Request Body:

```
<Tenant>
  <PilotNumber>12345</PilotNumber>
  <MediaSwitchObjectId>b24e4b57-8ae2-430f-807a-dec8124f76d2</MediaSwitchObjectId>
</Tenant>
```

Response Code: 204

JSON Example

To update the description of tenant, do the following:

Request URI:

```
PUT https://<connection-server>/vmrest/tenants/<Tenant-ObjectId>
Accept: applicaiton/json
Content_type: application/json
Connection: keep_alive
```

Request Body:

```
{
  "Tenant":
```

Cisco_Unity_Connection_Provisioning_Interface_(CUPI)_API_--_Tenant_API

```
{
  "PilotNumber": "12345",
  "MediaSwitchObjectId" : "b24e4b57-8ae2-430f-807a-dec8124f76d2"
}
<pre>
Response Code: 204
```

Note: If you update the mediaswitchobjectid and pilotnumber properties, the updates will be done in all the existing objects, such as users, usertemplate, etc. of particular tenant.

Update Name of Tenant

The following is an example of the PUT request that can be used to modify the Name of Tenant:

```
PUT :https://<connection-server>/vmrest/tenants/<Tenant-ObjectId>
```

Request Body:

```
<Tenant>
  <Name>Tenant Name</Name>
</Tenant>
```

```
Response Code: 204
```

JSON Example

To update the name of tenant, do the following:

Request URI:

```
PUT https://<connection-server>/vmrest/tenants/<Tenant-ObjectId>
Accept: applicaiton/json
Content_type: application/json
Connection: keep_alive
```

Request Body:

```
{
  "Tenant":
  {
    "Name": "Tenant_Name",
  }
}
<pre>
Response Code: 204
```

Viewing COSes associated with the Tenant

This request can be used to fetch COSes associated with a tenant.

```
GET https://< connection-server>/vmrest/tenants/<TenantObjectId>/coses
```

The following is the response from the above *GET* request and the actual response will depend upon the information given by you:

Cisco_Unity_Connection_Provisioning_Interface_(CUPI)_API_--_Tenant_API

```
<TenantCoses total="1">
  <TenantCos>
    <CosURI>/vmrest/coses/06ef7021-2549-4ab7-b80d-a6674749c17d</CosURI>
  </TenantCos>
</TenantCoses>
```

Response Code: 200

JSON Example

To view COSEs associated with the tenant, do the following:

```
GET https://<connection-server>/vmrest/tenants/<TenantObjectId>/coses
Accept: application/json
Connection: keep_alive
```

The following is the response from the above *GET* request and the actual response will depend upon the information given by you:

```
{
  "@total": "1",
  "TenantCos": {"CosURI": "/vmrest/coses/1afa317d-5533-46ab-b35b-6498953673a7"}
}
```

Response Code: 200

Viewing Partitions Associated with the Tenant

This request can be used to fetch partitions associated with a tenant.

```
GET https://<connection-server>/vmrest/tenants/<TenantObjectId>/partitions
```

The following is the response from the above *GET* request and the actual response will depend upon the information given by you:

```
<TenantPartitions total="1">
  <TenantPartition>
    <PartitionURI>/vmrest/coses/06ef7021-2549-4ab7-b80d-a6674749c17d</CosURI>
  </TenantPartition>
</TenantPartitions>
```

Response Code: 200

JSON Example

To view partitions associated with the tenant, do the following:

```
GET https://<connection-server>/vmrest/tenants/<TenantObjectId>/partitions
Accept: applicaiton/json
Connection: keep_alive
```

The following is the response from the above *GET* request and the actual response will depend upon the information given by you:

```
{
  "@total": "1",
  "TenantPartition": {"PartitionURI": "/vmrest/partitions/1469a460-e3d2-42c1-aad8-8f89e2c56ba8"}
}
```

Response Code: 200

Viewing Phone Systems Associated with the Tenant

This request can be used to fetch phone systems associated with a tenant.

```
GET https://<connection-server>/vmrest/tenants/<TenantObjectId>/phonesystems
```

The following is the response from the above *GET* request and the actual response will depend upon the information given by you:

```
<TenantPhoneSystems total="1">
  <TenantPhoneSystem>
    <PhoneSystemURI>/vmrest/phonesystems/598c2ca5-5df7-47be-8bf3-20af9f827657</PhoneSystemURI>
  </TenantPhoneSystem>
</TenantPhoneSystems>
```

Response Code: 200

JSON Example

To view phone system associated with the tenant, do the following:

Request URI:

```
GET https://<connection-server>/vmrest/tenants/<TenantObjectId>/phonesystems
Accept: application/json
Connection: keep_alive
```

The following is the response from the above *GET* request and the actual response will depend upon the information given by you:

```
{
  "@total": "1",
  "TenantPhoneSystem": { "PhoneSystemURI": "/vmrest/phonesystems/e815916c-abd5-49bea33d-0495d1f4"
}
```

Response Code: 200

Viewing Schedule Sets Associated with the Tenant

This request can be used to fetch schedule sets associated with a tenant.

```
GET https://<connection-server>/vmrest/tenants/<TenantObjectId>/schedulesets
```

The following is the response from the above *GET* request and the actual response will depend upon the information given by you:

```
<TenantScheduleSets total="1">
  <TenantScheduleSet>
    <ScheduleSetURI>/vmrest/schedulesets/4f5c5c0e-e853-423a-82a8-691ee8c18726</ScheduleSetURI>
  </TenantScheduleSet>
</TenantScheduleSets>
```

Response Code: 200

JSON Example

To view schedule sets associated with the tenant, do the following:

Request URI:

```
GET https://<connection-server>/vmrest/tenants/<TenantObjectId>/schedulesets
Accept: application/json
Connection: keep_alive
```

The following is the response from the above *GET* request and the actual response will depend upon the information given by you:

```
{
  "@total": "1",
  "TenantScheduleSet": {"ScheduleSetURI": "/vmrest/schedulesets/8189c416-6769-42b2-a5bc-2cc864e"}
}
```

Response Code: 200

Viewing Operator Associated with the Tenant

This request can be used to fetch operator associated with a tenant.

```
GET https://<connection-server>/vmrest/tenants/<TenantObjectId>/operators
```

The following is the response from the above *GET* request and the actual response will depend upon the information given by you:

```
<TenantOperators total="1">
  <TenantOperator>
    <OperatorURI>/vmrest/users/fd866d80-019f-4e9a-b447-8f73e46dbced</OperatorURI>
  </TenantOperator>
</TenantOperators>
```

Response Code: 200

JSON Example

To view operators associated with the tenant, do the following:

Request URI:

```
GET https://<connection-server>/vmrest/tenants/<TenantObjectId>/operators
Accept: application/json
Connection: keep_alive
```

The following is the response from the above *GET* request and the actual response will depend upon the information given by you:

```
{
  "@total": "1",
  "TenantOperator": {"OperatorURI": "/vmrest/users/e815916c-abd5-49be-a33d-0495d1f4e080"}
}
```

Response Code: 200

Explanation of Data Fields

The following chart lists all of the data fields:

Parameter	Data Type	Operations	Comments
URI	Read Only	String	URI of the tenant
ObjectId	Read Only	String (36)	Object Id of the tenant
CreationDate	Read Only	Datetime	The date and time when the tenant was created.
Alias	Read/Write	String (20)	The unique identifier for a tenant where maximum number of characters allowed is 20. Value can be specified in POST operation but cannot be modified afterwards.
Description	Read/Write	String(255)	The description of the tenant. This field is optional.
SmtptDomain	Read/Write	String(255)	Specifies an SMTP Domain for the tenant where each domain must be unique. Value can be specified in POST operation but cannot be modified afterwards.
Timezone	Read/Write	Integer	TimeZone of the tenant and the default objects created along with the tenant creation. If null value is passed in this parameter (i.e empty tag) then Pacific/wake timezone will get selected. Possible Values: TimeZoneId in the GET response of the URI: <a href="https://<connection-server>/vmrest/timezones">https://<connection-server>/vmrest/timezones • Default Value: System Default TimeZone.
Language	Read/Write	Integer	Specifies the language of the tenant. Same language is used for default objects created along with tenant creation. Possible Values: Installed languages can be retrieved using the URI: <a href="https://<connectionserver>/vmrest/installedlanguages">https://<connectionserver>/vmrest/installedlanguages • Default Values: System default Language
CosesURI	Read Only	String	The URI to fetch COSEs associated with the tenant.
ScheduleSetsURI	Read Only	String	The URI to fetch schedules associated with the tenant.
PhoneSystemsURI	Read Only	String	The URI to fetch phone systems associated with the tenant.
PartitionsURI	Read Only	String	The URI to fetch partitions associated with the tenant.
PartitionObjectId	Read Only	String(36)	The unique identifier for the Partition associated with the Tenant.
PhoneSystemObjectId	Read Only	String (36)	The unique identifier for the Phone System associated with the Tenant.
OpratorsURI	Read Only	String	The URI to fetch operator associated with the tenant.