Cisco Unity Connection Provisioning Interface (CUPI) API for Schedules

Cisco Unity Connection Provisioning Interface (CUPI) API -- Schedules

Overview of Cisco Unity Connection Schedule Objects

Schedules in Cisco Unity Connection are somewhat complicated, since they are composed of several different types of objects. Before presenting the API for accessing schedules, it may be useful to review what these objects are and how they are used.

ScheduleSet Objects

ScheduleSet objects are the top-level objects. They are comprised of one or more Schedules, each of which is marked as included or excluded in the ScheduleSet.

Schedule Objects

Schedule objects are composed of one or more ScheduleDetail objects.

ScheduleDetail Objects

ScheduleDetail objects are the atomic schedule building blocks which comprise Schedule objects. They can be specified with Start and End Dates, Start and End Times of the day, and active Days of the Week. ScheduleDetails have a link to the Schedule objects that they are part of, and said Schedule object in a sense owns the ScheduleDetail object.

ScheduleSetMemberMap Objects

ScheduleSetMemberMap objects provide the linkage between a ScheduleSet and a Schedule that is included or excluded from it (via a boolean called Exclude). There will be one ScheduleSetMemberMap object for each Schedule that is included in or excluded from a ScheduleSet.

A ScheduleSetMemberMap abstracts the linkage between a ScheduleSet and a Schedule since neither object has an explicit linkage or ownership relationship of the other. One reason for this is that several ScheduleSets might reference the same Schedule (a Holiday Schedule for example). This differs from the relationship between a Schedule and a ScheduleDetail, since a ScheduleDetail has an explicit link to a Schedule, the Schedule essentially owns the ScheduleDetail, and no other Schedule may use another Schedule's ScheduleDetail object.

Schedule Example

For example, let's model a weekday schedule with the lunch hour blocked out as unavailable every work day, and with various holidays blocked out as well. Using the objects discussed previously, this schedule might be composed like so:

- First, we create a top-level ScheduleSet called WeekdaySet. WeekdaySet includes the WeekdaySchedule and excludes the HolidaySchedule (2 Schedule objects).
- After we create these 2 Schedule objects, we create 2 ScheduleSetMemberMap objects for WeekdaySet - one to include WeekdaySchedule and one to exclude HolidaySchedule.
- Then, we create 2 ScheduleDetail objects for WeekdaySchedule - one active from 8AM to 12PM Mon-Fri, and the other active from 1PM to 5PM Mon-Fri.
- Finally, we create various ScheduleDetails objects for HolidaySchedule, one per holiday. For example, we might create a July4ScheduleDetail with start and end dates both set to July 4 2010, and a WinterBreakScheduleDetail with a start date of Dec 23 2010 and an end date of Jan 3 2011.

To see how to create this schedule via CUPI, see the Schedule Example page.

CUPI for Schedules

The previously described database objects are accessible to the administrator via CUPI. The following sections list the URIs to access these resources along with the data contained within them.

ScheduleSets in CUPI

ScheduleSets are top-level resources in CUPI, with a base URI of +/vmrest/schedulesets+.

A ScheduleSet object has the following fields:

Field Name	Field Type	Default	Notes
ObjectId	GUID	None	NA
TenantObjectId	Read Only	String (36)	The unique identifier of the tenant to which the schedulesets belong. This field is reflected in the response only if the schedulesets belong to a particular tenant.
DisplayName	String (64)	None	NA
OwnerLocationObjectId	GUID	Null	One of the Owners must be non-NULL

Field Name	Field Type	Default	Notes
OwnerPersonalRulesSetObjectId	GUID	Null	NA
OwnerSubscriberObjectId	GUID	Null	NA
Undeletable	Boolean	False	Is only True for factory default objects

Retrieving the list of ScheduleSets

To retrieve the list of ScheduleSets, an administrator makes a GET to the schedulesets resource:

```
GET /vmrest/schedulesets
```

This would return the following on success:

```
200
OK
<?xml version="1.0" encoding="UTF-8"?>
<ScheduleSets total="3">
  <ScheduleSet>
    <URI>/vmrest/schedulesets/30d9c0df-534b-437a-a6b7-439adfd850da</URI>
    <ObjectId>30d9c0df-534b-437a-a6b7-439adfd850da</ObjectId>
    <DisplayName>Weekdays</DisplayName>
    <OwnerLocationObjectId>6a56503e-c1c8-406c-85fd-76be40994d39</OwnerLocationObjectId>

<OwnerLocationURI>/vmrest/locations/connectionlocations/6a56503e-c1c8-406c-85fd-76be40994d39</OwnerLocationURI>

    <Undeletable>true</Undeletable>

<ScheduleSetMemberURI>/vmrest/schedulesets/30d9c0df-534b-437a-a6b7-439adfd850da/schedulesetmembers</ScheduleSetMemberURI>

  </ScheduleSet>
  <ScheduleSet>
    <URI>/vmrest/schedulesets/75af01af-d290-4e0e-9862-5adf8293536c</URI>
    <ObjectId>75af01af-d290-4e0e-9862-5adf8293536c</ObjectId>
    <DisplayName>All Hours</DisplayName>
    <OwnerLocationObjectId>6a56503e-c1c8-406c-85fd-76be40994d39</OwnerLocationObjectId>

<OwnerLocationURI>/vmrest/locations/connectionlocations/6a56503e-c1c8-406c-85fd-76be40994d39</OwnerLocationURI>

    <Undeletable>true</Undeletable>

<ScheduleSetMemberURI>/vmrest/schedulesets/75af01af-d290-4e0e-9862-5adf8293536c/schedulesetmembers</ScheduleSetMemberURI>

  </ScheduleSet>
  <ScheduleSet>
    <URI>/vmrest/schedulesets/e2e381e4-6096-4643-b0bb-b17a65b101bc</URI>
    <ObjectId>e2e381e4-6096-4643-b0bb-b17a65b101bc</ObjectId>
    <TenantObjectId>fe6541fb-b42c-44f2-8404-ded14cbf7438</TenantObjectId>
    <DisplayName>Voice Recognition Update Schedule</DisplayName>
    <OwnerLocationObjectId>6a56503e-c1c8-406c-85fd-76be40994d39</OwnerLocationObjectId>

<OwnerLocationURI>/vmrest/locations/connectionlocations/6a56503e-c1c8-406c-85fd-76be40994d39</OwnerLocationURI>

    <Undeletable>false</Undeletable>

<ScheduleSetMemberURI>/vmrest/schedulesets/e2e381e4-6096-4643-b0bb-b17a65b101bc/schedulesetmembers</ScheduleSetMemberURI>

  </ScheduleSet>
</ScheduleSets>
```

Listing Specific Tenant Related ScheduleSets by System Administrator

In Cisco Unity Connection 10.5(2) and later, the system administrator can use TenantObjectID to list the specific tenant related schedulesets using the following URI:

```
GET https://<connection-server>/vmrest/schedulesets?query=(TenantObjectId is
<Tenant-ObjectId>)
```

To get the TenantObjectID, use the following URI:

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

Retrieving a ScheduleSet

To retrieve a single ScheduleSet, an administrator makes a GET to the schedulesets resource, specifying the ObjectId of the requested ScheduleSet in the URI:

```
GET /vmrest/schedulesets/30d9c0df-534b-437a-a6b7-439adfd850da
```

This would return the following on success:

```
200
OK
<?xml version="1.0" encoding="UTF-8"?>
<ScheduleSet>
  <URI>/vmrest/schedulesets/30d9c0df-534b-437a-a6b7-439adfd850da</URI>
  <ObjectId>30d9c0df-534b-437a-a6b7-439adfd850da</ObjectId>
  <DisplayName>Weekdays</DisplayName>
  <OwnerLocationObjectId>6a56503e-c1c8-406c-85fd-76be40994d39</OwnerLocationObjectId>

<OwnerLocationURI>/vmrest/locations/connectionlocations/6a56503e-c1c8-406c-85fd-76be40994d39</OwnerLocationURI>

  <Undeletable>true</Undeletable>

<ScheduleSetMemberURI>/vmrest/schedulesets/30d9c0df-534b-437a-a6b7-439adfd850da/schedulesetmembers</ScheduleSetMemberURI>
</ScheduleSet>
```

This would return the following if the specified ScheduleSet does not exist:

```
404
Not Found
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ErrorDetails>
  <errors>
    <code>NOT_FOUND</code>
    <message>scheduleset - ObjectId=30d9c0df-534b-437a-a6b7-439adfd850da</message>
  </errors>
</ErrorDetails>
```

Adding a ScheduleSet

To add a new ScheduleSet, an administrator makes a POST to the schedulesets resource, specifying the new ScheduleSet via XML:

```
POST /vmrest/schedulesets

<ScheduleSet>
  <DisplayName>Night Shift</DisplayName>
  <OwnerLocationObjectId>6a56503e-c1c8-406c-85fd-76be40994d39</OwnerLocationObjectId>
</ScheduleSet>
```

This will return the URI to the newly created ScheduleSet (which includes its ObjectId):

```
201
Created
/vmrest/user/schedulesets/0e58ec49-5064-4c9a-b1dc-dd47fe189419
```

Changing a ScheduleSet

To change an existing ScheduleSet, an administrator makes a PUT to the schedulesets resource, specifying the ObjectId of the ScheduleSet they wish to change in the URI and any data changes via XML:

```
PUT /vmrest/schedulesets/30d9c0df-534b-437a-a6b7-439adfd850da

<ScheduleSet>
  <DisplayName>Graveyard Shift</DisplayName>
</ScheduleSet>
```

This would return the following on success:

```
204
No Content
```

This would return the following if the specified scheduleset does not exist:

```
400
Bad Request
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ErrorDetails>
  <errors>
    <code>DATA_EXCEPTION</code>
    <message>ScheduleSet not found </message>
  </errors>
</ErrorDetails>
```

Deleting a ScheduleSet

To delete an existing ScheduleSet, an administrator makes a DELETE to the schedulesets resource, specifying the ObjectId of the ScheduleSet they wish to delete in the URI:

```
DELETE /vmrest/schedulesets/30d9c0df-534b-437a-a6b7-439adfd850da
```

This would return the following on success:

```
204
No Content
```

This would return the following if the specified scheduleset does not exist:

```
404
Not Found
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ErrorDetails>
  <errors>
    <code>NOT_FOUND</code>
    <message>scheduleset - ObjectId=30d9c0df-534b-437a-a6b7-439adfd850da</message>
  </errors>
</ErrorDetails>
```

Note that if an administrator deletes a ScheduleSet object, then all of the supporting ScheduleSetMembers for that ScheduleSet will also be deleted.

Schedules in CUPI

Schedules are top-level resources in CUPI, with a base URI of +/vmrest/schedules+.

A Schedule object has the following fields:

Field Name	Field Type	Default	Notes
ObjectId	GUID	None	Na
TenantObjectId	Read Only	String (36)	The unique identifier of the tenant to which the schedule belong. This field is reflected in the response only if the schedule belong to a particular tenant.
DisplayNamre	String (64)	None	NA
OwnerLocationObjectId	GUID	NULL	One of the Owner's must be non-NULL
OwnerPersonalRuleSetObjectId	GUID	NULL	Na
OwnerSubscriberObjectId	GUID	NULL	NA
Undeletable	Boolean	False	Is only True for factory default objects
StartDate	DateTime	NULL	NULL means effective immediately
EndDate	DateTime	NULL	NULL means effective indefinitely
IsHoliday	Bolean	False	Holiday Greetings are used during this period if True

Retrieving the list of Schedules

To retrieve the list of Schedules, an administrator makes a GET to the schedules resource:

```
GET /vmrest/schedules
```

This would return the following on success:

```
200
OK
<?xml version="1.0" encoding="UTF-8"?>
<Schedules total="3">
  <Schedule>
    <URI>/vmrest/schedules/387f051e-3367-4cc8-abed-810293d39f76</URI>
    <ObjectId>387f051e-3367-4cc8-abed-810293d39f76</ObjectId>
    <DisplayName>Weekdays</DisplayName>
    <OwnerLocationObjectId>6a56503e-c1c8-406c-85fd-76be40994d39</OwnerLocationObjectId>

<OwnerLocationURI>/vmrest/locations/connectionlocations/6a56503e-c1c8-406c-85fd-76be40994d39</OwnerLocationURI>

    <Undeletable>true</Undeletable>
    <IsHoliday>false</IsHoliday>

<ScheduleDetailsURI>/vmrest/schedules/387f051e-3367-4cc8-abed-810293d39f76/scheduledetails</ScheduleDetailsURI>

  </Schedule>
  <Schedule>
    <URI>/vmrest/schedules/75af01af-d290-4e0e-9862-5adf8293536c</URI>
    <ObjectId>75af01af-d290-4e0e-9862-5adf8293536c</ObjectId>
    <DisplayName>All Hours</DisplayName>
    <OwnerLocationObjectId>6a56503e-c1c8-406c-85fd-76be40994d39</OwnerLocationObjectId>

<OwnerLocationURI>/vmrest/locations/connectionlocations/6a56503e-c1c8-406c-85fd-76be40994d39</OwnerLocationURI>

    <Undeletable>true</Undeletable>
    <IsHoliday>false</IsHoliday>

<ScheduleDetailsURI>/vmrest/schedules/75af01af-d290-4e0e-9862-5adf8293536c/scheduledetails</ScheduleDetailsURI>

  </Schedule>
  <Schedule>
    <URI>/vmrest/schedules/e2e381e4-6096-4643-b0bb-b17a65b101bc</URI>
    <ObjectId>e2e381e4-6096-4643-b0bb-b17a65b101bc</ObjectId>
    <TenantObjectId>fe6541fb-b42c-44f2-8404-ded14cbf7438</TenantObjectId>
    <DisplayName>Holidays</DisplayName>
    <OwnerLocationObjectId>6a56503e-c1c8-406c-85fd-76be40994d39</OwnerLocationObjectId>

<OwnerLocationURI>/vmrest/locations/connectionlocations/6a56503e-c1c8-406c-85fd-76be40994d39</OwnerLocationURI>

    <Undeletable>false</Undeletable>
    <IsHoliday>true</IsHoliday>

<ScheduleDetailsURI>/vmrest/schedules/e2e381e4-6096-4643-b0bb-b17a65b101bc/scheduledetails</ScheduleDetailsURI>

  </Schedule>
</Schedules>
```

Listing Specific Tenant Related Schedules by System Administrator

In Cisco Unity Connection 10.5(2) and later, the system administrator can use TenantObjectID to list the specific tenant related schedules using the following URI:

```
GET https://<connection-server>/vmrest/schedules?query=(TenantObjectId is <Tenant-ObjectId>)
```

To get the TenantObjectID, use the following URI:

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

Retrieving a Schedule

To retrieve a single Schedule, an administrator makes a GET to the schedules resource, specifying the Objectid of the requested Schedule in the URI:

```
GET /vmrest/schedules/387f051e-3367-4cc8-abed-810293d39f76
```

This would return the following on success:

```
200
OK
<?xml version="1.0" encoding="UTF-8"?>
<Schedule>
  <URI>/vmrest/schedules/387f051e-3367-4cc8-abed-810293d39f76</URI>
  <ObjectId>387f051e-3367-4cc8-abed-810293d39f76</ObjectId>
  <DisplayName>Weekdays</DisplayName>
  <OwnerLocationObjectId>6a56503e-c1c8-406c-85fd-76be40994d39</OwnerLocationObjectId>

<OwnerLocationURI>/vmrest/locations/connectionlocations/6a56503e-c1c8-406c-85fd-76be40994d39</OwnerLocationURI>

  <Undeletable>true</Undeletable>
  <IsHoliday>false</IsHoliday>

<ScheduleDetailsURI>/vmrest/schedules/387f051e-3367-4cc8-abed-810293d39f76/scheduledetails</ScheduleDetailsURI>
</Schedule>
```

This would return the following if the specified Schedule does not exist:

```
404
Not Found
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ErrorDetails>
  <errors>
    <code>NOT_FOUND</code>
    <message>schedule - ObjectId=387f051e-3367-4cc8-abed-810293d39f76</message>
  </errors>
</ErrorDetails>
```

Adding a Schedule

To add a new Schedule, an administrator makes a POST to the schedules resource, specifying the new Schedule via XML:

```
POST /vmrest/schedules

<Schedule>
  <DisplayName>EveningShift</DisplayName>
  <OwnerLocationObjectId>6a56503e-c1c8-406c-85fd-76be40994d39</OwnerLocationObjectId>
  <IsHoliday>false</IsHoliday>
</Schedule>
```

This will return the URI to the newly created Schedule (which includes its ObjectId):

```
201
Created
/vmrest/user/schedules/df7faf3b-3278-4852-b488-7e3134dc3336
```

Changing a Schedule

To change an existing Schedule, an administrator makes a PUT to the schedules resource, specifying the ObjectId of the Schedule they wish to change in the URI and any data changes via XML:

```
PUT /vmrest/schedules/df7faf3b-3278-4852-b488-7e3134dc3336

<Schedule>
  <DisplayName>No Daylight Shift</DisplayName>
</Schedule>
```

This would return the following on success:

```
204
No Content
```

This would return the following if the specified schedule does not exist:

```
400
Bad Request
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ErrorDetails>
  <errors>
    <code>DATA_EXCEPTION</code>
    <message>Schedule not found </message>
  </errors>
</ErrorDetails>
```

Deleting a Schedule

To delete an existing Schedule, an administrator makes a DELETE to the schedules resource, specifying the ObjectId of the Schedule they wish to delete in the URI:

```
DELETE /vmrest/schedules/387f051e-3367-4cc8-abed-810293d39f76
```

This would return the following on success:

```
204
No Content
```

This would return the following if the specified schedule does not exist:

```
404
Not Found
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ErrorDetails>
  <errors>
    <code>NOT_FOUND</code>
    <message>schedule - ObjectId=387f051e-3367-4cc8-abed-810293d39f76</message>
  </errors>
</ErrorDetails>
```

Note that if an administrator deletes a Schedule object, then all of the supporting ScheduleDetail objects for that Schedule will also be deleted.

ScheduleSetMembers in CUPI

ScheduleSetMembers are the resources representing ScheduleSetMemberMaps objects. They are sub-resources of ScheduleSets in CUPI, and thus are at a sub-URI of the schedulesets resource: +/vmrest/scheduleset/+{_}+<ScheduleSetObjectId>+{_}+/schedulesetmembers+.

A ScheduleSetMember object has the following fields:

Field Name	Field Type	Default	Notes
ScheduleSetObjectId	GUID	None	Owning ScheduleSet
ScheduleObjectId	GUID	None	Schedule to include/exclude from ScheduleSet
Exclude	Boolean	False	False means Include, True means Exclude

Retrieving the list of ScheduleSetMembers

To retrieve the list of ScheduleSetMembers for a particular ScheduleSet, an administrator makes a GET to the schedulesetmembers resource, specifying the ObjectId of the ScheduleSet in the URI:

```
GET /vmrest/schedulesets/30d9c0df-534b-437a-a6b7-439adfd850da/schedulesetmembers
```

This would return the following on success:

```
200
OK
<?xml version="1.0" encoding="UTF-8"?>
<ScheduleSetMembers total="2">
  <ScheduleSetMember>

<URI>/vmrest/schedulesets/30d9c0df-534b-437a-a6b7-439adfd850da/schedulesetmembers/a1f34a57-b642-4d8b-9634-b4c2e37bfd2b</URI>

    <ScheduleSetObjectId>30d9c0df-534b-437a-a6b7-439adfd850da</ScheduleSetObjectId>

<ScheduleSetURI>/vmrest/schedulesets/30d9c0df-534b-437a-a6b7-439adfd850da</ScheduleSetURI>

    <ScheduleObjectId>a1f34a57-b642-4d8b-9634-b4c2e37bfd2b</ScheduleObjectId>
    <ScheduleURI>/vmrest/schedules/a1f34a57-b642-4d8b-9634-b4c2e37bfd2b</ScheduleURI>
    <Exclude>false</Exclude>
  </ScheduleSetMember>
  <ScheduleSetMember>

<URI>/vmrest/schedulesets/30d9c0df-534b-437a-a6b7-439adfd850da/schedulesetmembers/279fd73f-36a9-469a-9ad5-a0f80f09b2d</URI>

    <ScheduleSetObjectId>30d9c0df-534b-437a-a6b7-439adfd850da</ScheduleSetObjectId>

<ScheduleSetURI>/vmrest/schedulesets/30d9c0df-534b-437a-a6b7-439adfd850da</ScheduleSetURI>

    <ScheduleObjectId>279fd73f-36a9-469a-9ad5-a0f80f09b2d</ScheduleObjectId>
    <ScheduleURI>/vmrest/schedules/279fd73f-36a9-469a-9ad5-a0f80f09b2d</ScheduleURI>
    <Exclude>true</Exclude>
  </ScheduleSetMember>
</ScheduleSetMembers>
```

Retrieving a ScheduleSetMember

To retrieve a single ScheduleSetMember for a particular ScheduleSet, an administrator makes a GET to the schedulesetmembers resource, specifying the ObjectIds of the ScheduleSet and the requested ScheduleSetMember in the URI:

```
GET
/vmrest/schedulesets/30d9c0df-534b-437a-a6b7-439adfd850da/schedulesetmembers/a1f34a57-b642-4d8b-9634-b4c2e37bfd2b
```

This would return the following on success:

```
200
OK
<?xml version="1.0" encoding="UTF-8"?>
<ScheduleSetMember>

<URI>/vmrest/schedulesets/30d9c0df-534b-437a-a6b7-439adfd850da/schedulesetmembers/a1f34a57-b642-4d8b-9634-b4c2e37bfd2b</URI>

  <ScheduleSetObjectId>30d9c0df-534b-437a-a6b7-439adfd850da</ScheduleSetObjectId>
  <ScheduleSetURI>/vmrest/schedulesets/30d9c0df-534b-437a-a6b7-439adfd850da</ScheduleSetURI>

  <ScheduleObjectId>a1f34a57-b642-4d8b-9634-b4c2e37bfd2b</ScheduleObjectId>
  <ScheduleURI>/vmrest/schedules/a1f34a57-b642-4d8b-9634-b4c2e37bfd2b</ScheduleURI>
  <Exclude>false</Exclude>
</ScheduleSetMember>
```

This would return the following if the specified ScheduleSetMember does not exist (meaning the Schedule is not listed as either included or excluded from the ScheduleSet):

```
404
Not Found
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ErrorDetails>
  <errors>
    <code>NOT_FOUND</code>
    <message>schedulesetmember - ObjectId=a1f34a57-b642-4d8b-9634-b4c2e37bfd2b</message>
  </errors>
</ErrorDetails>
```

Adding a ScheduleSetMember

To add a new ScheduleSetMember to a particular ScheduleSet, an administrator makes a POST to the schedulessetmembers resource, specifying the ObjectId of the ScheduleSet in the URI and the new ScheduleSetMember via XML:

```
POST /vmrest/schedulesets/30d9c0df-534b-437a-a6b7-439adfd850da/schedulesetmembers

<ScheduleSetMember>
  <ScheduleSetObjectId>30d9c0df-534b-437a-a6b7-439adfd850da</ScheduleSetObjectId>
  <ScheduleObjectId>db46f878-bc72-4870-9482-9f1c336641ed</ScheduleObjectId>
  <Exclude>false</Exclude>
</ScheduleSetMember>
```

This will return the URI to the newly created ScheduleSetMember (which includes its ObjectId):

```
201
Created
/vmrest/user/schedulesets/30d9c0df-534b-437a-a6b7-439adfd850da/schedulesetmembers/68c53107-5d28-4f40-ae43-f83d35eee8d6
```

Note A ScheduleSet can have at most 1 Schedule marked as Included, and that Schedule must not be a Holiday Schedule. A ScheduleSet can also have at most 1 Schedule marked as Excluded, and that Schedule must be a Holiday Schedule. If an administrator makes a POST to the schedulesetmembers resource in an attempt to Include or Exclude more than 1 Schedule of a given type to a ScheduleSet, then CUPI will return an HTTP 400 error with a descriptive error message. Similarly, if an administrator makes a POST to the schedulesetmembers resource in an attempt to Include a Holiday Schedule or Exclude a non-Holiday Schedule, then CUPI will return an HTTP 400 error with a descriptive error message.

Changing a ScheduleSetMember

The schedulesetmembers resource does not support the PUT method. In order to change a ScheduleSetMember, an administrator must delete the existing one and then add it back with the requested change.

Deleting a ScheduleSetMember

To delete an existing ScheduleSetMember for a particular ScheduleSet, an administrator makes a DELETE to the schedulesetmembers resource, specifying the ObjectIds of the ScheduleSet and the ScheduleSetMember they wish to delete in the URI:

```
DELETE
/vmrest/schedulesets/30d9c0df-534b-437a-a6b7-439adfd850da/schedulesetmembers/a1f34a57-b642-4d8b-9634-b4c2e37bfd2b
```

This would return the following on success:

```
204
No Content
```

This would return the following if the specified schedulesetmember does not exist:

```
404
Not Found
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ErrorDetails>
  <errors>
    <code>NOT_FOUND</code>
    <message>schedulesetmember - ObjectId=a1f34a57-b642-4d8b-9634-b4c2e37bfd2b</message>
  </errors>
</ErrorDetails>
```

ScheduleDetails in CUPI

ScheduleDetails are sub-resources of Schedules in CUPI, and thus are at a sub-URI of the schedules resources: +/vmrest/schedules/+{_}+<ScheduleObjectId>+{_}+/scheduledetails+.

A ScheduleDetail object has the following field:

Field Name	Field Type	Default	Notes
ObjectId	GUID	None	NA
Subject	String (2048)	None	NA
ScheduleObjectId	GUID	None	Owning Schedule

Field Name	Field Type	Default	Notes
StartDate	DateTime	NULL	NULL means effective immediately, also Time portion of DateTime is ignored
StartTime	Int	NULL	NULL means 12:00AM, otherwise minutes past 12:00AM so 480 means 8:00AM
EndDate	DateTime	NULL	NULL means effective indefinitely, also Time portion of DateTime is ignored
EndTime	Int	NULL	NULL means End-of-Day (11:59:59PM), otherwise minutes past 12:00AM so 1020 means 5:00PM
IsActiveMonday	Boolean	False	NA
IsActiveTuesday	Boolean	False	NA
IsActiveWednesday	Boolean	False	NA
IsActiveThursday	Boolean	False	NA
IsActiveFriday	Boolean	False	NA
IsActiveSaturday	Boolean	False	NA
IsActiveSunday	Boolean	False	NA

Retrieving the list of ScheduleDetails

To retrieve the list of ScheduleDetails for a particular Schedule, an administrator makes a GET to the scheduledetails resource, specifying the ObjectId of the Schedule in the URI:

```
GET /vmrest/schedules/387f051e-3367-4cc8-abed-810293d39f76/scheduledetails
```

This would return the following on success:

```
200
OK
<?xml version="1.0" encoding="UTF-8"?>
<ScheduleDetails total="2">
  <ScheduleDetail>

<URI>/vmrest/schedules/387f051e-3367-4cc8-abed-810293d39f76/scheduledetails/fb6cb280-ea91-4ee5-9225-6ca9c5e3b77e</URI>

    <ObjectId>fb6cb280-ea91-4ee5-9225-6ca9c5e3b77e</ObjectId>
    <ScheduleObjectId>387f051e-3367-4cc8-abed-810293d39f76</ScheduleObjectId>
    <ScheduleURI>/vmrest/schedules/387f051e-3367-4cc8-abed-810293d39f76</ScheduleURI>
    <Subject>Weekday Mornings</Subject>
    <StartTime>480</StartTime>
    <EndTime>720</EndTime>
    <IsActiveMonday>true</IsActiveMonday>
    <IsActiveTuesday>true</IsActiveTuesday>
    <IsActiveWednesday>true</IsActiveWednesday>
    <IsActiveThursday>true</IsActiveThursday>
    <IsActiveFriday>true</IsActiveFriday>
    <IsActiveSaturday>false</IsActiveSaturday>
    <IsActiveSunday>false</IsActiveSunday>
  </ScheduleDetail>
  <ScheduleDetail>

<URI>/vmrest/schedules/387f051e-3367-4cc8-abed-810293d39f76/scheduledetails/e049dc1e-7447-4e2a-907e-03d67e2d40a1</URI>

    <ObjectId>e049dc1e-7447-4e2a-907e-03d67e2d40a1</ObjectId>
    <ScheduleObjectId>387f051e-3367-4cc8-abed-810293d39f76</ScheduleObjectId>
    <ScheduleURI>/vmrest/schedules/387f051e-3367-4cc8-abed-810293d39f76</ScheduleURI>
    <Subject>Weekday Afternoons</Subject>
    <StartTime>780</StartTime>
    <EndTime>1020</EndTime>
    <IsActiveMonday>true</IsActiveMonday>
    <IsActiveTuesday>true</IsActiveTuesday>
    <IsActiveWednesday>true</IsActiveWednesday>
    <IsActiveThursday>true</IsActiveThursday>
    <IsActiveFriday>true</IsActiveFriday>
    <IsActiveSaturday>false</IsActiveSaturday>
    <IsActiveSunday>false</IsActiveSunday>
  </ScheduleDetail>
</ScheduleDetails>
```

Retrieving a ScheduleDetail

To retrieve a single ScheduleDetail for a particular Schedule, an administrator makes a GET to the scheduledetails resource, specifying the ObjectIds of the Schedule and the requested ScheduleDetail in the URI:

```
GET
/vmrest/schedules/387f051e-3367-4cc8-abed-810293d39f76/scheduledetails/fb6cb280-ea91-4ee5-9225-6ca9c5e3b77e
```

This would return the following on success:

```
200
OK
<?xml version="1.0" encoding="UTF-8"?>
<ScheduleDetail>

<URI>/vmrest/schedules/387f051e-3367-4cc8-abed-810293d39f76/scheduledetails/fb6cb280-ea91-4ee5-9225-6ca9c5e3b77e</URI>

  <ObjectId>fb6cb280-ea91-4ee5-9225-6ca9c5e3b77e</ObjectId>
  <ScheduleObjectId>387f051e-3367-4cc8-abed-810293d39f76</ScheduleObjectId>
  <ScheduleURI>/vmrest/schedules/387f051e-3367-4cc8-abed-810293d39f76</ScheduleURI>
  <Subject>Weekday Mornings</Subject>
  <StartTime>480</StartTime>
  <EndTime>720</EndTime>
  <IsActiveMonday>true</IsActiveMonday>
  <IsActiveTuesday>true</IsActiveTuesday>
  <IsActiveWednesday>true</IsActiveWednesday>
  <IsActiveThursday>true</IsActiveThursday>
  <IsActiveFriday>true</IsActiveFriday>
  <IsActiveSaturday>false</IsActiveSaturday>
  <IsActiveSunday>false</IsActiveSunday>
</ScheduleDetail>
```

This would return the following if the specified ScheduleDetail does not exist:

```
404
Not Found
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ErrorDetails>
  <errors>
    <code>NOT_FOUND</code>
    <message>scheduledetail - ObjectId=fb6cb280-ea91-4ee5-9225-6ca9c5e3b77e</message>
  </errors>
</ErrorDetails>
```

Adding a ScheduleDetail

To add a new ScheduleDetail to a particular Schedule, an administrator makes a POST to the scheduledetails resource, specifying the ObjectId of the Schedule in the URI and the new ScheduleDetail via XML:

```
POST /vmrest/schedules/387f051e-3367-4cc8-abed-810293d39f76/scheduledetails

<ScheduleDetail>
  <Subject>Saturday (Half-day)</Subject>
  <StartTime>540</StartTime>
  <EndTime>780</EndTime>
  <IsActiveSaturday>true</IsActiveSaturday>
</ScheduleDetail>
```

This will return the URI to the newly created ScheduleDetail (which includes its ObjectId):

```
201
Created
/vmrest/schedules/387f051e-3367-4cc8-abed-810293d39f76/scheduledetails/9d8afb61-bd4b-4e65-b274-e870a2b51865
```

Changing a ScheduleDetail

To change an existing ScheduleDetail for a particular Schedule, an administrator makes a PUT to the scheduledetails resource, specifying the ObjectId of the Schedule and the ScheduleDetail they wish to change in the URI and the data changes via XML:

```
PUT
/vmrest/schedules/387f051e-3367-4cc8-abed-810293d39f76/scheduledetails/fb6cb280-ea91-4ee5-9225-6ca9c5e3b77e

<ScheduleDetail>
  <StartTime>450</StartTime>
</ScheduleDetail>
```

This would return the following on success:

```
204
No Content
```

This would return the following if the specified ScheduleDetail does not exist:

```
400
Bad Request
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ErrorDetails>
  <errors>
    <code>DATA_EXCEPTION</code>
    <message>ScheduleDetail not found </message>
  </errors>
</ErrorDetails>
```

Deleting a ScheduleDetail

To delete an existing ScheduleDetail from a particular Schedule, an administrator makes a DELETE to the scheduledetails resource, specifying the ObjectId of the Schedule and the ScheduleDetail they wish to delete in the URI:

```
 DELETE
/vmrest/schedules/387f051e-3367-4cc8-abed-810293d39f76/scheduledetails/9d8afb61-bd4b-4e65-b274-e870a2b51865
```

This would return the following on success:

```
204
No Content
```

This would return the following if the specified ScheduleDetail does not exist:

```
404
Not Found
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ErrorDetails>
  <errors>
    <code>NOT_FOUND</code>
    <message>scheduledetail - ObjectId=9d8afb61-bd4b-4e65-b274-e870a2b51865</message>
  </errors>
</ErrorDetails>
```

Cisco Unity Connection Provisioning Interface (CUPI) API -- Holiday Schedules

Holiday Schedules API

Administrator can use this API to create/update/delete/fetch the holiday schedules. Various attributes of holiday schedules can also be updated using this API.

Listing the Holiday Schedule Details

The following is an example of the GET request that list the holiday schedules:

```
GET https://<connection-server>/vmrest/schedules/<holidayschedule-objectid>/scheduledetails
```

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

```
<ScheduleDetails total="1">
     <ScheduleDetail>

<URI>/vmrest/schedules/f62e3780-4bfc-4c8f-91a4-3ac4e35803c4/scheduledetails/894f29bc-8ff1-4183-b8c1-f1c304d9109b</URI>

          <ObjectId>894f29bc-8ff1-4183-b8c1-f1c304d9109b</ObjectId>
          <ScheduleObjectId>f62e3780-4bfc-4c8f-91a4-3ac4e35803c4</ScheduleObjectId>
        <ScheduleURI>/vmrest/schedules/f62e3780-4bfc-4c8f-91a4-3ac4e35803c4</ScheduleURI>

          <IsActiveMonday>false</IsActiveMonday>
          <IsActiveTuesday>false</IsActiveTuesday>
          <IsActiveWednesday>false</IsActiveWednesday>
          <IsActiveThursday>false</IsActiveThursday>
          <IsActiveFriday>false</IsActiveFriday>
          <IsActiveSaturday>true</IsActiveSaturday>
          <IsActiveSunday>true</IsActiveSunday>
          <StartTime>480</StartTime>
          <EndTime>1020</EndTime>
     </ScheduleDetail>
</ScheduleDetails>
```

```
Response Code: 200
```

JSON Example

To list all the holiday schedules, do the following:

```
GET https://<connection-server>/vmrest/schedules/<holidayscheduleobjectid>/scheduleobjectid
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",
     "ScheduleDetail":
     {
          "URI":
"/vmrest/schedules/f62e3780-4bfc-4c8f-91a4-3ac4e35803c4/scheduledetails/894f29bc-8ff1-4183-b8c1-f1c304d9109b",

          "ObjectId": "894f29bc-8ff1-4183-b8c1-f1c304d9109b",
          "ScheduleObjectId": "f62e3780-4bfc-4c8f-91a4-3ac4e35803c4",
          "ScheduleURI": "/vmrest/schedules/f62e3780-4bfc-4c8f-91a4-3ac4e35803c4",
          "IsActiveMonday": "false",
          "IsActiveTuesday": "false",
          "IsActiveWednesday": "false",
          "IsActiveThursday": "false",
          "IsActiveFriday": "false",
          "IsActiveSaturday": "true",
          "IsActiveSunday": "true",
          "StartTime": "480",
          "EndTime": "1020"
     }
}
```

```
Response Code: 200
```

Viewing the Specific Holiday Schedule

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

```
GET https://<connection-server>/vmrest/schedules/<holidayscheduleobjectid>
```

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

```
<Schedule>
     <URI>/vmrest/schedules/f62e3780-4bfc-4c8f-91a4-3ac4e35803c4</URI>
     <ObjectId>f62e3780-4bfc-4c8f-91a4-3ac4e35803c4</ObjectId>
     <DisplayName>Tenant1_Holiday_1</DisplayName>
     <OwnerLocationObjectId>97a1e9ab-6a69-4272-952a-b0e25c08aaaf</OwnerLocationObjectId>

<OwnerLocationURI>/vmrest/locations/connectionlocations/97a1e9ab-6a69-4272-952ab0e25c08aaaf</OwnerLocationURI>

     <IsHoliday>true</IsHoliday>
     <Undeletable>false</Undeletable>

<ScheduleDetailsURI>/vmrest/schedules/f62e3780-4bfc-4c8f-91a4-3ac4e35803c4/scheduledetails</ScheduleDetailsURI>
</Schedule>
```

```
Response Code: 200
```

JSON Example

To view a specific holiday schedule, do the following:

```
GET https://<connection-server>/vmrest/schedules/<holidayscheduleobjectid>
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/schedules/10c9ac6c-6a4c-4559-be75-2c409ef85054",
     "ObjectId": "10c9ac6c-6a4c-4559-be75-2c409ef85054",
     "DisplayName": "Tenant2_Holiday_1",
     "OwnerLocationObjectId": "97a1e9ab-6a69-4272-952a-b0e25c08aaaf",
     "OwnerLocationURI":
"/vmrest/locations/connectionlocations/97a1e9ab-6a69-4272-952ab0e25c08aaaf",
     "IsHoliday": "true",
     "Undeletable": "false",
     "ScheduleDetailsURI":
"/vmrest/schedules/10c9ac6c-6a4c-4559-be75-2c409ef85054/scheduledetails"
}
```

```
Response Code: 200
```

Creating a New Holiday Schedule

The following is an example of POST request that can be used to create a new holiday schedule:

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

Request Body:

```
<Schedule>
     <DisplayName>Texoma_Holiday</DisplayName>
     <OwnerLocationObjectId>5150cb31-a665-47d8-a311-9cc4524810d39</OwnerLocationObjectId>
     <IsHoliday>true</IsHoliday>
</Schedule>
```

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/schedules/f62e3780-4bfc-4c8f-91a4-3ac4e35803c4
```

JSON Example

To create a new holiday schedule, do the following:

```
POST https://<connection-server>/vmrest/schedules
Accept: application/json
Content-Type: application/json
Connection: keep-alive
```

```
Request Body:
{
     "DisplayName":"Texoma_Holiday ",
     "OwnerLocationObjectId":"5150cb31-a665-47d8-a311-9cc4524810d3",
     "IsHoliday":"true"
}
```

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/schedules/f62e3780-4bfc-4c8f-91a4-3ac4e35803c4
```

Updating the Holiday Schedule

The following is an example of the PUT request that can be used to modify the holiday schedule where only description field can be updated:

The following is an example of the PUT request that can be used to modify the holiday schedule where only description field can be updated:

```
PUT https://<connection-server>/vmrest/schedules/<holidayscheduleobjectid>
```

Request Body:

```
<Schedule>
     <DisplayName>Texoma1_Holiday</DisplayName>
     <OwnerLocationObjectId>5150cb31-a665-47d8-a311-9cc4524810d3</OwnerLocationObjectId>
</Schedule>
```

```
Response Code: 204
```

JSON Example

To update the holiday schedule, do the following:

```
PUT https://<connection-server>/vmrest/schedules/<holidayscheduleobjectid>
Accept: application/json
Content-Type: application/json
Connection: keep-alive
```

Request Body:

```
{
"DisplayName":"Texoma1_Holiday ",
"OwnerLocationObjectId":"5150cb31-a665-47d8-a311-9cc4524810d3"
}
```

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
```

Deleting a Holiday Schedule

This request can be used to delete an existing holiday Schedule; an administrator makes a DELETE to the schedules resource, specifying the object ID of the holiday schedule they wish to delete in the URI:

```
DELETE https://<connection-server>/vmrest/schedules/<holidayscheduleobjectid>
```

```
Response Code: 204
```

JSON Example

```
DELETE https://<connection-ip>/vmrest/schedules/<holidayschedule-objectid>
Accept: application/json
Connection: keep-alive
```

```
Response Code: 204
```

Explanation of Data Fields

The following chart lists all of the data fields:

Parameter	Data Type	Operations	Comments
ObjectId	String (36)	Read Only	Specifies a globally unique, system-generated identifier for a Schedule object.
URI	String	Read Only	Specifies the URI for schedule.
DisplayName	String (64)	Read/Write	Specifies the preferred text name of this scheduleto be used when displaying entries in the administrative console and Cisco Personal Assistant.
OwnerLocationObjectId	String (36)	Read Only	Specifies the owner of this schedule. If the owner is a LocationVMS, the unique identifier of the LocationVMS objects to which this schedule belongs. Commonly referred to as a "system" schedule. Otherwise, this attribute is set to NULL.
OwnerPersonalRuleSetObjectId	String (36)	Read Only	Specifies the owner of this schedule. If the owner is a personal rule set, the unique identifier of the personal rule set to which this schedule belongs. Otherwise, this attribute is set to NULL. OwnerSubscriberObjectId String(36) Read Only Specifies the owner of this schedule. If the owner is a subscriber, the unique identifier of the Subscriber objects to which this schedule belongs. Otherwise, this attribute is set to NULL. Undeletable Boolean Read/Write Indicates a flag that checks whether this schedule can be deleted via an administrative application such as Cisco Unity Connection Administration. It is used to prevent deletion of factory defaults.
StartDate	DateTime	Read/Write	The date and time when schedule becomes active, or NULL if active immediately.
StartTime	Integer	Read/Write	The start time (in minutes) for the active day or days. The start time is stored as the number of minutes from midnight. So a value of 480 would mean 8:00 AM and 1020 would mean 5:00 PM. In addition, a value of NULL for the start time indicates 12:00 AM. • Range 0-1435
EndDate	DateTime	Read/Write	The date and time when schedule ends, or NULL if effective indefinitely.
EndTime	Integer	Read/Write	The end time (in minutes) for the active day or days. The end time is stored as the number of minutes from midnight. So a value of 480 would mean 8:00 AM and 1020 would mean 5:00 PM. In addition, a value of NULL means "till the end of the day" (e.g. 11:59:59 PM in Linux land). • Range 0-1435

Parameter	Data Type	Operations	Comments
IsHoliday	Boolean	Read/Write	A flag indicating whether this schedule represents a holiday.
SchedulesDetailsURI	String	Read Only	URI for scheduledetails

Cisco Unity Connection Provisioning Interface (CUPI) API -- Schedules Set

Schedule Set API

Administrator can use this API to create/update/delete/fetch the schedule sets. Various attributes of schedule sets can also be updated using this API.

Listing the Schedule Sets

The following is an example of the GET request that list the schedule sets:

```
GET https://<connection-server>/vmrest/schedulesets
The following is the response from the above *GET* request and the actual response will
depend upon the information given by you:
<pre>
<ScheduleSets total="3">
     <ScheduleSet>
          <URI>/vmrest/schedulesets/e01b7fa7-521b-47f7-82d0-bb898aeec903</URI>
          <ObjectId>e01b7fa7-521b-47f7-82d0-bb898aeec903</ObjectId>
          <DisplayName>Weekdays</DisplayName>
         <OwnerLocationObjectId>51bff648-e60f-44ec-b2c8-ae854dfc1f41</OwnerLocationObjectId>


<OwnerLocationURI>/vmrest/locations/connectionlocations/51bff648-e60f-44ec-b2c8-ae854dfc1f41</OwnerLocationURI>

          <Undeletable>true</Undeletable>

<ScheduleSetMembersURI>/vmrest/schedulesets/e01b7fa7-521b-47f7-82d0-bb898aeec903/schedulesetmembers</ScheduleSetMembersURI>

     </ScheduleSet>
     <ScheduleSet>
          <URI>/vmrest/schedulesets/8f2e394c-1d09-412e-8e09-c26b152344c1</URI>
          <ObjectId>8f2e394c-1d09-412e-8e09-c26b152344c1</ObjectId>
          <DisplayName>All Hours</DisplayName>
         <OwnerLocationObjectId>51bff648-e60f-44ec-b2c8-ae854dfc1f41</OwnerLocationObjectId>


<OwnerLocationURI>/vmrest/locations/connectionlocations/51bff648-e60f-44ec-b2c8-ae854dfc1f41</OwnerLocationURI>

          <Undeletable>true</Undeletable>

<ScheduleSetMembersURI>/vmrest/schedulesets/8f2e394c-1d09-412e-8e09-c26b152344c1/schedulesetmembers</ScheduleSetMembersURI>

     </ScheduleSet>
     <ScheduleSet>
          <URI>/vmrest/schedulesets/31c9ff78-f6b3-4731-9df2-dce8de411f33</URI>
          <ObjectId>31c9ff78-f6b3-4731-9df2-dce8de411f33</ObjectId>
          <TenantObjectId>fe6541fb-b42c-44f2-8404-ded14cbf7438</TenantObjectId>
          <DisplayName>Voice Recognition Update Schedule</DisplayName>
         <OwnerLocationObjectId>51bff648-e60f-44ec-b2c8-ae854dfc1f41</OwnerLocationObjectId>


<OwnerLocationURI>/vmrest/locations/connectionlocations/51bff648-e60f-44ec-b2c8-ae854dfc1f41</OwnerLocationURI>

          <Undeletable>false</Undeletable>

<ScheduleSetMembersURI>/vmrest/schedulesets/31c9ff78-f6b3-4731-9df2-dce8de411f33/schedulesetmembers</ScheduleSetMembersURI>

          </ScheduleSet>
</ScheduleSets>
```

```
Response Code: 200
```

JSON Example

To list all the schedule sets, do the following:

```
GET https://<connection-server>/vmrest/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": "3",
     "ScheduleSet": [
     {
          "URI": "/vmrest/schedulesets/e01b7fa7-521b-47f7-82d0-bb898aeec903",
          "ObjectId": "e01b7fa7-521b-47f7-82d0-bb898aeec903",
          "DisplayName": "Weekdays",
          "OwnerLocationObjectId": "51bff648-e60f-44ec-b2c8-ae854dfc1f41",
          "OwnerLocationURI":
"/vmrest/locations/connectionlocations/51bff648-e60f-44ec-b2c8-ae854dfc1f41",
          "Undeletable": "true",
          "ScheduleSetMembersURI":
"/vmrest/schedulesets/e01b7fa7-521b-47f7-82d0-bb898aeec903/schedulesetmembers"
     },
     {
          "URI": "/vmrest/schedulesets/8f2e394c-1d09-412e-8e09-c26b152344c1",
          "ObjectId": "8f2e394c-1d09-412e-8e09-c26b152344c1",
          "DisplayName": "All Hours",
          "OwnerLocationObjectId": "51bff648-e60f-44ec-b2c8-ae854dfc1f41",
          "OwnerLocationURI":
"/vmrest/locations/connectionlocations/51bff648-e60f-44ec-b2c8-ae854dfc1f41",
          "Undeletable": "true",
          "ScheduleSetMembersURI":
"/vmrest/schedulesets/8f2e394c-1d09-412e-8e09-c26b152344c1/schedulesetmembers"
     },
     {
          "URI": "/vmrest/schedulesets/31c9ff78-f6b3-4731-9df2-dce8de411f33",
          "ObjectId": "31c9ff78-f6b3-4731-9df2-dce8de411f33",
          "TenantObjectId": "fe6541fb-b42c-44f2-8404-ded14cbf7438",
          "DisplayName": "Voice Recognition Update Schedule",
          "OwnerLocationObjectId": "51bff648-e60f-44ec-b2c8-ae854dfc1f41",
          "OwnerLocationURI":
"/vmrest/locations/connectionlocations/51bff648-e60f-44ec-b2c8-ae854dfc1f41",
          "Undeletable": "false",
          "ScheduleSetMembersURI":
"/vmrest/schedulesets/31c9ff78-f6b3-4731-9df2-dce8de411f33/schedulesetmembers"
     }
]
}
```

```
Response Code: 200
```

Listing Specific Tenant Related ScheduleSets by System Administrator

In Cisco Unity Connection 10.5(2) and later, the system administrator can use TenantObjectID to list the specific tenant related schedulesets using the following URI:

```
GET https://<connection-server>/vmrest/schedulesets?query=(TenantObjectId is
<Tenant-ObjectId>)
```

To get the TenantObjectID, use the following URI:

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

Viewing the Specific Schedule Set

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

```
GET https://<connection-server>/vmrest/schedulesets/<scheduleset-objectid>
```

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

```
<ScheduleSet>
     <URI>/vmrest/schedulesets/e01b7fa7-521b-47f7-82d0-bb898aeec903</URI>
     <ObjectId>e01b7fa7-521b-47f7-82d0-bb898aeec903</ObjectId>
     <DisplayName>Weekdays</DisplayName>
     <OwnerLocationObjectId>51bff648-e60f-44ec-b2c8-ae854dfc1f41</OwnerLocationObjectId>

<OwnerLocationURI>/vmrest/locations/connectionlocations/51bff648-e60f-44ec-b2c8-ae854dfc1f41</OwnerLocationURI>

     <Undeletable>true</Undeletable>

<ScheduleSetMembersURI>/vmrest/schedulesets/e01b7fa7-521b-47f7-82d0-bb898aeec903/schedulesetmembers</ScheduleSetMembersURI>
</ScheduleSet>
```

```
Response Code: 200
```

JSON Example

To view a specific schedule set, do the following:

```
GET https://<connection-server>/vmrest/schedulesets/<scheduleset-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/schedulesets/e01b7fa7-521b-47f7-82d0-bb898aeec903",
     "ObjectId": "e01b7fa7-521b-47f7-82d0-bb898aeec903",
     "DisplayName": "Weekdays",
     "OwnerLocationObjectId": "51bff648-e60f-44ec-b2c8-ae854dfc1f41",
     "OwnerLocationURI":
"/vmrest/locations/connectionlocations/51bff648-e60f-44ec-b2c8-ae854dfc1f41",
     "Undeletable": "true",
     "ScheduleSetMembersURI":
"/vmrest/schedulesets/e01b7fa7-521b-47f7-82d0-bb898aeec903/schedulesetmembers"
}
```

```
Response Code: 200
```

Creating a New Schedule Set

The following is an example of POST request that can be used to create a new schedule set:

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

Request Body

```
<ScheduleSet>
     <DisplayName>Texoma_DayShift </DisplayName>
     <OwnerLocationObjectId>5150cb31-a665-47d8-a311-9cc4524810d3</OwnerLocationObjectId>
</ScheduleSet>
```

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/schedulesets/e01b7fa7-521b-47f7-82d0-bb898aeec903
```

JSON Example

To create a new schedule set, do the following:

```
POST https://<connection-server>/vmrest/schedulesets
Accept: application/json
Content-Type: application/json
Connection: keep-alive
```

Request Body

```
{
     "DisplayName":"Texoma_DayShift ",
     "OwnerLocationObjectId":"5150cb31-a665-47d8-a311-9cc4524810d3"
}
```

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/schedulesets/e01b7fa7-521b-47f7-82d0-bb898aeec903
```

Updating the Schedule Set

The following is an example of the PUT request that can be used to modify the schedule set where only description field can be updated:

```
PUT https://<connection-server>/vmrest/schedulesets/<scheduleset-objectid>
```

Request Body

```
<ScheduleSet>
     <DisplayName>Texoma_Evening</DisplayName>
     <OwnerLocationObjectId>5150cb31-a665-47d8-a311-9cc4524810d3</OwnerLocationObjectId>
</ScheduleSet>
```

```
Response Code: 204
```

JSON Example

To update the schedule set, do the following:

```
PUT https://<connection-server>/vmrest/schedulesets/<scheduleset-objectid>
Accept: application/json
Content-Type: application/json
Connection: keep-alive
```

Request Body

```
{
     "DisplayName":"Texoma_Evening ",
     "OwnerLocationObjectId":"5150cb31-a665-47d8-a311-9cc4524810d3"
}
```

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
```

Delete a Schedule Set

This request can be used to delete an existing schedule set; an administrator makes a DELETE to the schedule sets resource, specifying the object ID of the schedule set they wish to delete in the URI.

```
DELETE https://<connection-server>/vmrest/schedulesets/<scheduleset-objectid>
```

```
Response Code: 204
```

JSON Example

```
DELETE: https://<connection-ip>/vmrest/schedulesets/<scheduleset-objectid>
Accept: application/json
Connection: keep-alive
```

```
Response Code: 204
```

Explanation of Data Fields

Parameter	Data Type	Operation	Comments
ObjectId	String	Read Only	Specifies a globally unique, system-generated identifier for a schedule set object.
TenantObjectId	String	Read Only	The unique identifier of the tenant to which the schedule set belongs. This field is reflected in the response only if the schedule set belongs to a particular tenant.
URI	String	Read Only	Specifies the URI for schedule set.
DisplayName	String	Read/Write	Specifies the unique text name of this schedule set to be used when displaying entries in the administrative console and Cisco Personal Assistant.
OwnerLocationObjectId	String	Read Only	The owner of this schedule set. If the owner is a LocationVMS, the unique identifiers of the LocationVMS object to which this schedule set (i.e., "system" schedule) belongs. Otherwise, this attribute is set to NULL.
OwnerPersonalRuleSetObjectId	String	Read Only	The owner of this schedule set. If the owner is a personal rule set, the unique identifier of the personal rule set to which this schedule set belongs. Otherwise, this attribute is set to NULL.

Parameter	Data Type	Operation	Comments
OwnerSubscriberObjectId	String	Read Only	The owner of this schedule set. If the owner is a subscriber, the unique identifier of the Subscriber object to which this schedule set belongs. Otherwise, this attribute is set to NULL.
Undeletable	Boolean	Read/Write	Indicates a flag which checks whether this schedule set can be deleted via an administrative application such as Cisco Unity Connection Administration. It is used to prevent deletion of factory defaults. Default value: false.
ScheduleSetMembersURI	String	Read Only	Specifies the URI for schedule set members.

Cisco Unity Connection Provisioning Interface (CUPI) API -- Schedules Set Details

Schedule Set Details API

Administrator can use this API to create/update/delete/fetch the schedule set details. Various attributes of schedule set details can also be updated using this API.

Cisco Unity Connection Provisioning Interface (CUPI) API -- Schedules Set Members

Schedule Set Members API

Administrator can use this API to create/update/delete/fetch the schedule set members. Various attributes of schedule set members can also be updated using this API.

Listing the Schedule Set Members

The following is an example of the GET request that lists the schedule set members:

```
GET https://<connection-server>/vmrest/schedulesets/<scheduleset-objectid>/schedulesetmembers
```

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

```
<ScheduleSetMembers total="2">
     <ScheduleSetMember>

<URI>/vmrest/schedulesets/c309414c-0c72-4d37-8a22-9ed7d9ee9b3e/schedulesetmembers/2008f07d-4265-4570-ab6f-362228dd8180</URI>

          <ScheduleSetObjectId>c309414c-0c72-4d37-8a22-9ed7d9ee9b3e</ScheduleSetObjectId>

<ScheduleSetURI>/vmrest/schedulesets/c309414c-0c72-4d37-8a22-9ed7d9ee9b3e</ScheduleSetURI>

          <ScheduleObjectId>2008f07d-4265-4570-ab6f-362228dd8180</ScheduleObjectId>
        <ScheduleURI>/vmrest/schedules/2008f07d-4265-4570-ab6f-362228dd8180</ScheduleURI>

          <Exclude>false</Exclude>
     </ScheduleSetMember>
     <ScheduleSetMember>

<URI>/vmrest/schedulesets/c309414c-0c72-4d37-8a22-9ed7d9ee9b3e/schedulesetmembers/c1bdc7d4-fd7c-4ae7-a836-6094f987ad74</URI>

          <ScheduleSetObjectId>c309414c-0c72-4d37-8a22-9ed7d9ee9b3e</ScheduleSetObjectId>

<ScheduleSetURI>/vmrest/schedulesets/c309414c-0c72-4d37-8a22-9ed7d9ee9b3e</ScheduleSetURI>

          <ScheduleObjectId>c1bdc7d4-fd7c-4ae7-a836-6094f987ad74</ScheduleObjectId>
        <ScheduleURI>/vmrest/schedules/c1bdc7d4-fd7c-4ae7-a836-6094f987ad74</ScheduleURI>

          <Exclude>true</Exclude>
     </ScheduleSetMember>
</ScheduleSetMembers>
<pre>
Response Code: 200
```

JSON Example

To list all the schedule set members, do the following:

```
GET https://<connection-server>/vmrest/schedulesets/<scheduleset-objectid>/schedulesetmembers
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": "2",
     "ScheduleSetMember[
     {
          "URI":
"/vmrest/schedulesets/c309414c-0c72-4d37-8a22-9ed7d9ee9b3e/schedulesetmembers/2008f07d-4265-4570-ab6f-362228dd8180",

          "ScheduleSetObjectId": "c309414c-0c72-4d37-8a22-9ed7d9ee9b3e",
          "ScheduleSetURI": "/vmrest/schedulesets/c309414c-0c72-4d37-8a22-9ed7d9ee9b3e",
          "ScheduleObjectId": "2008f07d-4265-4570-ab6f-362228dd8180",
          "ScheduleURI": "/vmrest/schedules/2008f07d-4265-4570-ab6f-362228dd8180",
          "Exclude": "false"
     },
     {
          "URI":
"/vmrest/schedulesets/c309414c-0c72-4d37-8a22-9ed7d9ee9b3e/schedulesetmembers/c1bdc7d4-fd7c-4ae7-a836-6094f987ad74",

          "ScheduleSetObjectId": "c309414c-0c72-4d37-8a22-9ed7d9ee9b3e",
          "ScheduleSetURI": "/vmrest/schedulesets/c309414c-0c72-4d37-8a22-9ed7d9ee9b3e",
          "ScheduleObjectId": "c1bdc7d4-fd7c-4ae7-a836-6094f987ad74",
          "ScheduleURI": "/vmrest/schedules/c1bdc7d4-fd7c-4ae7-a836-6094f987ad74",
          "Exclude": "true"
     }
]
}
```

```
Response Code: 200
```

Viewing the Specific Schedule Set Member

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

```
GET
https://<connection-server>/vmrest/schedulesets/<schedulesetobjectid>/schedulesetmembers/<schedulesetmember-objectid>
```

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

```
<URI>/vmrest/schedulesets/c309414c-0c72-4d37-8a22-9ed7d9ee9b3e/schedulesetmembers/2008f07d-4265-4570-ab6f-362228dd8180</URI>

    <ScheduleSetObjectId>c309414c-0c72-4d37-8a22-9ed7d9ee9b3e</ScheduleSetObjectId>

<ScheduleSetURI>/vmrest/schedulesets/c309414c-0c72-4d37-8a22-9ed7d9ee9b3e</ScheduleSetURI>

    <ScheduleObjectId>2008f07d-4265-4570-ab6f-362228dd8180</ScheduleObjectId>
    <ScheduleURI>/vmrest/schedules/2008f07d-4265-4570-ab6f-362228dd8180</ScheduleURI>
    <Exclude>false</Exclude>
```

</ScheduleSetMember> </pre>

```
Response Code: 200
```

JSON Example

To view a specific schedule set member, do the following:

```
GET
https://<connection-server>/vmrest/schedulesets/<schedulesetobjectid>/schedulesetmembers/<schedulesetmember-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/schedulesets/c309414c-0c72-4d37-8a22-9ed7d9ee9b3e/schedulesetmembers/2008f07d-4265-4570-ab6f-362228dd8180",

     "ScheduleSetObjectId": "c309414c-0c72-4d37-8a22-9ed7d9ee9b3e",
     "ScheduleSetURI": "/vmrest/schedulesets/c309414c-0c72-4d37-8a22-9ed7d9ee9b3e",
     "ScheduleObjectId": "2008f07d-4265-4570-ab6f-362228dd8180",
     "ScheduleURI": "/vmrest/schedules/2008f07d-4265-4570-ab6f-362228dd8180",
     "Exclude": "false"
}
```

```
Response Code: 200
```

Creating a New Schedule Set Member

The following is an example of POST request that can be used to create a new schedule set member:

```
POST
https://<connection-server>/vmrest/schedulesets/<scheduleset-objectid>/schedulesetmembers
```

Request Body for a Non-Holiday Schedule:

```
<ScheduleSetMember>
     <ScheduleSetObjectId>11a4f6b1-c926-4404-9bba-964ebb3075c3</ScheduleSetObjectId>
     <ScheduleObjectId>4ac392b8-e276-46a1-978a-9b648c2c785b</ScheduleObjectId>
     <Exclude>false</Exclude>
</ScheduleSetMember>

The following is the response from the above *POST* request and the actual response will
depend upon the information given by you:
<pre>
Response Code: 201
/vmrest/schedulesets/c309414c-0c72-4d37-8a22-9ed7d9ee9b3e/schedulesetmembers/2008f07d-4265-4570-ab6f-362228dd8180
```

Note A schedule set can have at most 1 schedule marked as included, and that schedule must not be a holiday schedule. A schedule set can also have at most 1 schedule marked as excluded, and that schedule must be a holiday schedule.

JSON Example for a non-holiday schedule

To create a new schedule set member, do the following:

```
POST
https://<connection-server>/vmrest/schedulesets/<scheduleset-objectid>/schedulesetmembers
Accept: application/json
Content-Type: application/json
Connection: keep-alive
```

```
{
     "ScheduleSetObjectId":"11a4f6b1-c926-4404-9bba-964ebb3075c3",
     "ScheduleObjectId":"4ac392b8-e276-46a1-978a-9b648c2c785b",
     "Exclude":"false"
}
The following is the response from the above *POST* request and the actual response will
depend upon the information given by you:
<pre>
Response Code: 201
/vmrest/schedulesets/c309414c-0c72-4d37-8a22-9ed7d9ee9b3e/schedulesetmembers/2008f07d-4265-4570-ab6f-362228dd8180
```

Request Body for a Holiday Schedule:

```
<ScheduleSetMember>
     <ScheduleSetObjectId>11a4f6b1-c926-4404-9bba-964ebb3075c3</ScheduleSetObjectId>
     <ScheduleObjectId>4ac392b8-e276-46a1-978a-9b648c2c785b</ScheduleObjectId>
     <Exclude>true</Exclude>
</ScheduleSetMember>
```

```
Response Code: 201
/vmrest/schedulesets/c309414c-0c72-4d37-8a22-9ed7d9ee9b3e/schedulesetmembers/2008f07d-4265-4570-ab6f-362228dd8180
```

JSON Example to create Schedule set member which is a Holiday Schedule:

```
POST https://<connection-ip>/vmrest/schedulesets/<scheduleset-objectid>/schedulesetmembers
Accept: application/json
Content-Type: application/json
Connection: keep-alive
```

```
{
     "ScheduleSetObjectId":"11a4f6b1-c926-4404-9bba-964ebb3075c3",
     "ScheduleObjectId":"4ac392b8-e276-46a1-978a-9b648c2c785b",
     "Exclude":"true"
}
```

```
Response Code: 201
/vmrest/schedulesets/c309414c-0c72-4d37-8a22-9ed7d9ee9b3e/schedulesetmembers/2008f07d-4265-4570-ab6f-362228dd8180
```

Updating the Schedule Set Members

The following is an example of the PUT request that can be used to modify the schedule set members where only description field can be updated:

```
PUT
https://<connection-server>/vmrest/schedulesets/<schedulesetobjectid>/schedulesetmembers/<schedulesetmember-objectid>
```

Request Body

```
<ScheduleSetMember>
     <ScheduleSetObjectId>11a4f6b1-c926-4404-9bba-964ebb3075c3</ScheduleSetObjectId>
     <ScheduleObjectId>f4e064d5-e21e-4234-a157-91554fd657e5</ScheduleObjectId>
     <Exclude>true</Exclude>
</ScheduleSetMember>
```

```
Response Code: 204
```

JSON Example

To update the schedule set member, do the following:

```
PUT
https://<connection-server>/vmrest/schedulesets/<schedulesetobjectid>/schedulesetmembers/<schedulesetmember-objectid>
Accept: application/json
Content-Type: application/json
Connection: keep-alive
```

Request Body

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
```

Deleting a Schedule Set Member

To delete an existing schedule set member, an administrator makes a DELETE to the schedule set members resource, specifying the object ID of the schedule set member they wish to delete in the URI:

```
DELETE
https://<connection-server>/vmrest/schedulesets/<schedulesetobjectid>/schedulesetmembers/<schedulesetmember-objectid>
```

```
Response Code: 204
```

JSON Example to delete an existing ScheduleSet Member:

```
DELETE
https://<connection-ip>/vmrest/schedulesets/<schedulesetobjectid>/schedulesetmembers/<schedulesetmember-objectid>
Accept: application/json
Connection: keep-alive
```

```
Response Code: 204
```

Explanation of Data Fields

Parameter	Data Type	Operation	Comments
ScheduleSetObjectId	String	Read Only	The unique identifier of the ScheduleSet object to which the schedule belongs.
ScheduleSetURI	String	Read Only	URI of the ScheduleSet
ScheduleObjectId	String	Read Only	The unique identifier of the Schedule object that is a member of the schedule set
ScheduleURI	String	Read Only	URI of the Schedule
Exclude	Boolean	Read/Write	A flag indicating whether the schedule is to be included or excluded from the schedule set. Default value is false