Cisco Unity Connection Provisioning Interface (CUPI) API for Dial Plan

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

Partition API

Administrator can use this API to create, update, delete, or fetch the partition. In Cisco Unity Connection, you create partitions as a way to group together objects to which callers and users can address messages or place calls while interacting with Connection. One or more partitions can be grouped together as members of a search space, and a partition can be a member of more than one search space. The following types of objects belong to a partition:

- User Templates
- Call Handler Templates
- Contact Templates
- Users with mailboxes (primary extension)
- User alternate extensions
- Contacts (including VPIM contacts)
- System distribution lists
- System call handlers
- Directory handlers
- Interview handlers
- VPIM locations

Extensions must be unique within a partition, although partitions can contain objects that do not have an associated extension (for example, some contacts and system distribution lists). The names of objects do not have to be unique within a partition. Administrator-defined contact phone numbers also do not need to be unique within a partition.

Listing the Partitions

The following is an example of the GET request that lists all the existing partitions:

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

The following is the response from the above *GET* request:

```
<Partitions total="2">
  <Partition>
     <URI>/vmrest/partitions/53e16c90-1cc6-4bde-b6c8-77daf1d31d02</URI>
     <ObjectId>53e16c90-1cc6-4bde-b6c8-77daf1d31d02</ObjectId>
     <Name>ucbu-aricent-vm256 Partition</Name>
     <Description>Default Partition</Description>
     <LocationObjectId>6e81cad2-9c0b-42c1-bb6f-4edb38a70cbd</LocationObjectId>
     <LocationURI>/vmrest/locations/connectionlocations/6e81cad2-9c0b-42c1-bb6f-
    4edb38a70cbd</LocationURI>
  </Partition>
  <Partition>
     <URI>/vmrest/partitions/924ee2d5-758b-4ded-8db7-ae7bc11b59b2</URI>
     <ObjectId>924ee2d5-758b-4ded-8db7-ae7bc11b59b2</ObjectId>
     <Name>Taxoma_DefaultPartition</Name>
     <Description>Default Partition for Taxoma</Description>
     <LocationObjectId>6e81cad2-9c0b-42c1-bb6f-4edb38a70cbd</LocationObjectId>
     <LocationURI>/vmrest/locations/connectionlocations/6e81cad2-9c0b-42c1-bb6f-
    4edb38a70cbd</LocationURI>
  </Partition>
</Partitions>
```

```
Response Code: 200
```

JSON Example

The following is an example of the GET command that will list all the partitions:

```
Request URI:
GET https://<connection-server>/vmrest/partitions
Accept: applciation/json
Connection: keep_alive
```

The following is the example of the response from the above *GET* request:

```
{
  "@total": "2",
  "Partition": [
  {
     "URI": "/vmrest/partitions/d50e9d0b-656e-416d-b5b7-43c4d2e2fd0b",
     "ObjectId": "d50e9d0b-656e-416d-b5b7-43c4d2e2fd0b",
     "Name": "ucbu-aricent-vm259 Partition",
     "Description": "Default Partition",
     "LocationObjectId": "bbf3e6ed-0278-479c-9a6e-2da8756eeb6f",
     "LocationURI": "/vmrest/locations/connectionlocations/bbf3e6ed-0278-479c-
   9a6e-2da8756eeb6f"
  },
  {
     "URI": "/vmrest/partitions/5c28f078-8142-4844-b3c0-8fddf33639c4",
     "ObjectId": "5c28f078-8142-4844-b3c0-8fddf33639c4",
     "Name": "Taxoma21_Partition_1",
     "LocationObjectId": "bbf3e6ed-0278-479c-9a6e-2da8756eeb6f",
     "LocationURI": "/vmrest/locations/connectionlocations/bbf3e6ed-0278-479c-
    9a6e-2da8756eeb6f"
  }
  ]
}
```

```
Response Code: 200
```

Viewing a Specific Partition

The following is an example of the GET request that list the detail of a particular partition:

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

The following is the response from the above *GET* request:

```
<Partition>
     <LocationObjectId>bbf3e6ed-0278-479c-9a6e-2da8756eeb6f</LocationObjectId>
     <URI>/vmrest/partitions/d50e9d0b-656e-416d-b5b7-43c4d2e2fd0b</URI>
     <ObjectId>d50e9d0b-656e-416d-b5b7-43c4d2e2fd0b</ObjectId>
     <Name>ucbu-aricent-vm259 Partition</Name>
     <Description>Default Partition</Description>
     <LocationObjectId>bbf3e6ed-0278-479c-9a6e-2da8756eeb6f</LocationObjectId>
     <LocationURI>/vmrest/locations/connectionlocations/bbf3e6ed-0278-479c-9a6e-
    2da8756eeb6f</LocationURI>
</Partition>
```

```
Response Code: 200
```

JSON Example

The following is an example of the GET command that will get the detail of a particular partition:

```
Request URI:
GET https://<connection-server>/vmrest/partitions/<PartitionObjectId>
Accept: applciation/json
Connection: keep_alive
```

The following is the example of the response from the above *GET* request:

```
{
     "URI": "/vmrest/partitions/d50e9d0b-656e-416d-b5b7-43c4d2e2fd0b",
     "ObjectId": "d50e9d0b-656e-416d-b5b7-43c4d2e2fd0b",
     "Name": "ucbu-aricent-vm259 Partition",
     "Description": "Default Partition",
     "LocationObjectId": "bbf3e6ed-0278-479c-9a6e-2da8756eeb6f",
     "LocationURI": "/vmrest/locations/connectionlocations/bbf3e6ed-0278-479c-9a6e-
    2da8756eeb6f"
}
```

```
Response Code: 200
```

Creating a Partition

The following is an example of the POST request that creates a partition:

```
POST https://<connection-server>/vmrest/partitions
Request Body:
<Partition>
     <Name>Taxoma_DefaultPartition</Name>
     <Description>Default Partition for Taxoma</Description>
</Partition>
```

The following is the example of the response from the above *Post* request:

```
Response Code: 201
```

JSON Example

The following is an example of the POST command that will create a new partition:

```
Request URI:
POST https://<connection-server>/vmrest/partitions
Accept: applciation/json
Content_type: application/json
Connection: keep_alive
Request Body:
{
     "Name": "Test",
     "Description": "Test_partition"
}
```

The following is the example of the response from the above *GET* request:

```
Response Code: 201
```

Updating a Partition

The following is an example of the PUT request that allows you to update a partition:

```
PUT https://<connection-server>/vmrest/partitions/<PartitionObjectId>
Request Body:
<Partition>
     <Name>Test123</Name>
     <Description>Taxoma Default Partition</Description>
</Partition>
```

```
Response Code: 204
```

The output for this request returns the successful response code.

JSON Example

The following is an example of the PUT command that will update a partition:

```
PUT https://<connection-server>/vmrest/partitions/<PartitionObjectId>
Accept: applciation/json
Content_type: application/json
Connection: keep_alive
Request Body:
{
     "Name": "Test123",
     "Description": "Hello"
}
```

The following is the example of the response from the above *PUT * request:

```
Response Code: 204
```

Deleting a Partition

The following is an example of the DELETE request that deletes a specific partition where you need to mention the partition object ID:

Note You are not allowed to delete a default search space if a partition is referred in a search space, or referred by a user.

```
Response Code: 204
```

The output for this request returns the successful response code.

JSON Example

The following is an example of the DELETE command that will delete a partition:

```
DELETE https://<connection-server>/vmrest/partitions/<PartitionObjectId>
Accept: applciation/json
Connection: keep_alive
```

```
Response Code: 204
```

Deleting a Partition Included in Search Scope

Make sure to add the partition to a search space. The following is an example of the DELETE request that deletes a specific partition which is included in a search scope:

```
Delete https://<connection-server>/vmrest/partitions/<PartitionObjectId>
```

```
Response Code: 204
```

JSON Example

The following is an example to delete a partition included in a search scope:

```
Delete https://<connection-server>/vmrest/partitions/<PartitionObjectId>
Accept: application/json
Connection: keep_alive
```

```
Response Code: 204
```

Explanation of Data Fields

Parameter	Operations	Data Type	Comments
ObjectId	Read Only	String (36)	The primary key for this table. A systemgenerated, globally unique object identifier.
Name	Read/Write	String (50)	The unique text name of this partition. This name is used when displaying entries in the administrative interface. The name can be upto 50 characters long.
Description	Read/Write	String (50)	The description of the partition. If a description is not entered, the partition name is entered.
LocationObjectId	Read Only	String (36)	The unique identifier of the LocationVms on which this partition was created. It is used by networking to determine on which LocationVms the properties of the partition are administered.

Cisco Unity Connection Provisioning Interface (CUPI) API -- Search Space API

Search Space API

Search spaces are used to define the search scope of objects (users, distribution lists, and so on) that a user or outside caller can reach while interacting with Connection. For example, the search scope that is applied to a user identifies which users, distribution lists, or VPIM contacts the user can address messages to. It also identifies which users and contacts the user can dial by name when using the voicerecognition conversation. A search space is comprised of one or more ordered partitions. When Connection searches for an object on behalf of a caller, it searches the partitions in the order in which they are arranged in the search space. While extensions must be unique within a partition, they do not need to be unique within a search space, so you can use search spaces to handle dial plans that have overlapping extensions. Operation Supported: Create, Read, Update and Delete.

Listing the Search Spaces

The following is an example of the GET request that lists all the search spaces:

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

The following is the response from the above *GET* request.

```
<SearchSpaces total="1">
  <SearchSpace>
     <URI>/vmrest/searchspaces/7479b688-9c97-4a4b-ab69-ea786f6024be</URI>
     <ObjectId>7479b688-9c97-4a4b-ab69-ea786f6024be</ObjectId>
     <Name>Search Space</Name>
     <Description>Default Search Space</Description>
     <LocationObjectId>08b0402c-ba81-4f16-bcba-ca7883de0482</LocationObjectId>
     <LocationURI>/vmrest/locations/connectionlocations/08b0402c-ba81-4f16-bcba
  ca7883de0482</LocationURI>
     <TimeOwnershipChanged>2012-12-10T19:23:29Z</TimeOwnershipChanged>
     <SearchSpaceMembersURI>/vmrest/searchspaces/7479b688-9c97-4a4b-ab69-
  ea786f6024be/searchspacemembers</SearchSpaceMembersURI>
  </SearchSpace>
</SearchSpaces>
```

```
Response Code: 200
```

JSON Example

To list all the search spaces use the following command, do the following: Request URI: GET https://<connection-server>/vmrest/searchspaces 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",
 "SearchSpace": [
 {
    "URI":"/vmrest/searchspaces/2e836e16-f715-4a18-bb7c-ee5e33281706"
    "ObjectId":"2e836e16-f715-4a18-bb7c-ee5e33281706"
    "Name":"ucbu-aricent-vm463 Search Space"
    "Description":"Default Search Space"
    "LocationObjectId":"cff1347e-87af-4409-bead-d1970625f82e"
    "LocationURI":"/vmrest/locations/connectionlocations/cff1347e-87af-4409-bead
    d1970625f82e"
    "TimeOwnershipChanged":"2013-02-25 09:37:36.99"
    "SearchSpaceMembersURI":"/vmrest/searchspaces/2e836e16-f715-4a18-bb7c
    ee5e33281706/searchspacemembers"
 },
 {
    "URI":"/vmrest/searchspaces/97efa220-31a7-4409-a0cd-2cdfc669182f"
    "ObjectId":"97efa220-31a7-4409-a0cd-2cdfc669182f"
    "Name":"hello_SearchSpace_1"
    "LocationObjectId":"cff1347e-87af-4409-bead-d1970625f82e"
   "LocationURI":"/vmrest/locations/connectionlocations/cff1347e-87af-4409-beadd1970625f82e"

    "TimeOwnershipChanged":"2013-02-26T05:05:10Z"
    "SearchSpaceMembersURI":"/vmrest/searchspaces/97efa220-31a7-4409-a0cd-
    2cdfc669182f/searchspacemembers"
 }
 ]
```

```
Response Code: 200
```

Viewing the Specific Search Space Details

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

```
GET https://<connection-server>/vmrest/searchspaces/<searchspaceobjectId>
```

The following is the response from the above *GET* request:

```
<SearchSpaceMembers total="1">
  <SearchSpaceMember>
     <URI>/vmrest/searchspaces/7479b688-9c97-4a4b-ab69-
    ea786f6024be/searchspacemembers/8433db04-60e0-414d-99bc-c48dcf8254cb</URI>
     <ObjectId>8433db04-60e0-414d-99bc-c48dcf8254cb</ObjectId>
    <PartitionObjectId>ff1c9ff2-6fcc-47e9-a4e0-a220c6e3ae4e</PartitionObjectId>
    <PartitionURI>/vmrest/partitions/ff1c9ff2-6fcc-47e9-a4e0-a220c6e3ae4e</PartitionURI>
    <SearchSpaceObjectId>7479b688-9c97-4a4b-ab69-ea786f6024be</SearchSpaceObjectId>
    <SearchSpaceURI>/vmrest/searchspaces/7479b688-9c97-4a4b-ab69-
   ea786f6024be</SearchSpaceURI>
    <SortOrder>1</SortOrder>
  </SearchSpaceMember>
</SearchSpaceMembers>
```

```
Response Code: 200
```

JSON Example

To view the specific search space, use the following command:

```
GET https://<connection-server>/vmrest/searchspaces/<searchspaceobjectid>
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/searchspaces/2e836e16-f715-4a18-bb7c-ee5e33281706"
     "ObjectId":"2e836e16-f715-4a18-bb7c-ee5e33281706"
     "Name":"ucbu-aricent-vm463 Search Space"
     "Description":"Default Search Space"
     "LocationObjectId":"cff1347e-87af-4409-bead-d1970625f82e"
     "LocationURI":"/vmrest/locations/connectionlocations/cff1347e-87af-4409-bead
     d1970625f82e"
     "TimeOwnershipChanged":"2013-02-25 09:37:36.99"
     "SearchSpaceMembersURI":"/vmrest/searchspaces/2e836e16-f715-4a18-bb7c
     ee5e33281706/searchspacemembers"
}
```

```
Response Code: 200
```

Creating a Search Space

The following is an example of the POST request that creates a search space:

```
POST https://<connection-server>/vmrest/searchspaces
Request URI:
<SearchSpace>
     <Name>Taxoma_SearchSpace</Name>
</SearchSpace>
```

The following is the example of the response from the above *Post* request:

```
Response Code: 201
/vmrest/searchspaces/4fc208e7-cb84-4e3f-bcf0-11f69c212b1d
```

JSON Example

To create a search space, use the following command:

```
Request URI:
POST https://<connection-server>/vmrest/searchspaces
Accept: application/json
Content_type: application/json
Connection: keep_alive
Request Body:
{
     "Name": "Taxoma_SearchSpace1"
}
```

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/searchspaces/4fc208e7-cb84-4e3f-bcf0-11f69c212b1d
```

Creating a Search Space

The following is an example of the POST request that creates a search space:

```
POST https://<connection-server>/vmrest/searchspaces
Request URI:
<SearchSpace>
     <Name>Taxoma_SearchSpace</Name>
</SearchSpace>
```

The following is the example of the response from the above *Post* request:

```
Response Code: 201
/vmrest/searchspaces/4fc208e7-cb84-4e3f-bcf0-11f69c212b1d
```

JSON Example

To create a search space, use the following command:

```
Request URI:
POST https://<connection-server>/vmrest/searchspaces
Accept: application/json
Content_type: application/json
Connection: keep_alive
Request Body:
{
     "Name": "Taxoma_SearchSpace1"
}
```

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/searchspaces/4fc208e7-cb84-4e3f-bcf0-11f69c212b1d
```

Adding a Member in a Search Space

The following conditions should be checked before you add a member in a search space:

- Identify the search space to which you want to add partitions.
- Obtain the object ID of the partition that has to be added using the following URI

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

The following is an example of the POST request that allows you to add a member in a search space:

```
POST https://<connection-server>/vmrest/searchspaces/<ObjectId>/searchspacemembers
Request Body:
  <SearchSpaceMember>
     <PartitionObjectId>2be94430-a2b4-4a0c-b53c-a0290810d0ef</PartitionObjectId>
     <SortOrder>1</SortOrder>
  </SearchSpaceMember>
```

The output for this request returns the following response:

```
Response Code: 201
/vmrest/searchspaces/4fc208e7-cb84-4e3f-bcf0-11f69c212b1d/searchspacemembers/f93deeb3-71b2-
4198-bd55-53dc4a11f37c
```

JSON Example

To add a member in a search space, use the following command:

```
Request URI:
POST https://<connection-server>/vmrest/searchspaces/seacrchspaceobjectid/searchspacemembers
Accept: application/json
Content_type: application/json
Connection: keep_alive
Request Body:
{
     "PartitionObjectId": "97bf6afe-346e-4275-967e-43c50be79d32",
     "SortOrder": "1"
}
```

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/searchspaces/72bf712e-ebad-4398-997c-5c21fb1c18c4/searchspacesmembers/0453712e-ebadeadc-
997c-5c21fb1c18c4
```

List all the Search Space Members Associated with a Specific Search Space

The following is an example of the GET request that lists all the search space members associated with a specific search space:

```
GET https://<connection-server>/vmrest/searchspaces/seacrchspaceobjectid/searchspacemembers
<SearchSpaceMembers total="1">
  <SearchSpaceMember>
     <URI>/vmrest/searchspaces/7479b688-9c97-4a4b-ab69-
     ea786f6024be/searchspacemembers/8433db04-60e0-414d-99bc-c48dcf8254cb</URI>
     <ObjectId>8433db04-60e0-414d-99bc-c48dcf8254cb</ObjectId>
     <PartitionObjectId>ff1c9ff2-6fcc-47e9-a4e0-a220c6e3ae4e</PartitionObjectId>
     <PartitionURI>/vmrest/partitions/ff1c9ff2-6fcc-47e9-a4e0-
     a220c6e3ae4e</PartitionURI>
     <SearchSpaceObjectId>7479b688-9c97-4a4b-ab69-
     ea786f6024be</SearchSpaceObjectId>
     <SearchSpaceURI>/vmrest/searchspaces/7479b688-9c97-4a4b-ab69-
     ea786f6024be</SearchSpaceURI>
     <SortOrder>1</SortOrder>
  </SearchSpaceMember>
</SearchSpaceMembers>
```

```
Response Code: 200
```

JSON Example

To add a member in a search space, use the following command: GET https://<connection-server>/vmrest/searchspaces/seacrchspaceobjectid/searchspacemembers Accept: application/json Content_type: 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"
 "SearchSpaceMember":
 {
    "URI":"/vmrest/searchspaces/8190d682-9775-4622-9609-
    a113e594b84a/searchspacemembers/540fc81a-d391-4760-9c5f-3d5eac6b8568"
    "ObjectId":"540fc81a-d391-4760-9c5f-3d5eac6b8568"
    "PartitionObjectId":"97bf6afe-346e-4275-967e-43c50be79d32"
    "PartitionURI":"/vmrest/partitions/97bf6afe-346e-4275-967e-43c50be79d32"
    "SearchSpaceObjectId":"8190d682-9775-4622-9609-a113e594b84a"
    "SearchSpaceURI":"/vmrest/searchspaces/8190d682-9775-4622-9609-a113e594b84a"
    "SortOrder":"1"
 }
```

```
Response Code: 200
<pre>


=== Updating a Search Space ===

The following is an example of the PUT request that allows you to update a search space:
<pre>
  PUT https://<connection-server>/vmrest/searchspaces/<searchspaceobjectid>
Request Body:
<SearchSpace>
     <Name>Taxoma_SearchSpace</Name>
     <Description>Default search space for taxoma</Description>
</SearchSpace>
```

```
Response Code: 204
```

The output for this request returns the successful response code.

JSON Example

To update a search space, use the following command:

```
Request URI:
PUT https://<connection-server>/vmrest/searchspaces/seacrchspaceobjectid/
Accept: application/json
Content_type: application/json
Connection: keep_alive
Request Body:
{
     "Name": "Taxoma_SearchSpace",
     "Description": "Default search space for taxoma"
}
```

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 the Search Space

The following is an example of the DELETE request that deletes a specific search space where you need to mention the search space object ID:

```
DELETE https://<connection-server>/vmrest/searchspaces/<seacrchspaceobjectid>
```

```
Response Code: 204
```

Note You are not allowed to delete a default search space or if a search space that has been assigned to any user, call handler or a routing rule. The output for this request returns the successful response code.

JSON Example

To delete a search space, use the following command:

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

```
Response Code: 204
```

Deleting the Search Space Members

The following is an example of the DELETE request that deletes a specific search space member where you need to mention the search space members object ID:

```
DELETE https://<connection-server>/vmrest/searchspaces/<seacrchspaceobjectid>
/searchspacemembers/<searchspacememberobjectid>
```

```
Response Code: 204
```

JSON Example

To delete a search space member, use the following command:

```
DELETE: https://<connectionserver>/
vmrest/searchspaces/seacrchspaceobjectid/searchspacemembers/<searchspacememberobjectid>
Accept: application/json
Connection: keep_alive
```

```
Response Code: 204
```

Explanation of Data Fields

Search Space

The following chart lists all of the data fields for search space:

Parameter	Operations	Data Type	Comments
URI	Read Only	String	The search space URL.
ObjectId	Read Only	String (36)	The primary key for this table. A systemgenerated, globally unique object identifier.
Name	Read/Write	String (50)	The unique text name of this search space.
Description	Read/Write	String (50)	The description of the search space.
LocationObjectId	Read Only	String (36)	The unique identifier on which this search space was created.
LocationURI	Read Only	String	The location of the URL.
TimeOwnershipChanged		Date/Time	The time when the ownership of this search space changed. If ownership has never changed, this value will show the creation time.
SearchSpaceMemberURI	Read Only	String	URI of the search space member

Search Space Members

Parameter	Operations	Data Type	Comments
URI	Read Only	String	URI of Search Space's member
ObjectId	Read Only	String (36)	Object Id of the search space's member
PartitionObjectId	Read Only	String (36)	Object Id of the partition
PartitionURI	Read Only	String	URI of the partition.

Parameter	Operations	Data Type	Comments
SearchSpaceObjectId	Read Only	String (36)	Object Id of the search space
SearchSpaceURI	Read Only	String	URI of the search space
SortOrder	Read Only	Integer	Order preference of the member Partition.

Cisco Unity Connection Provisioning Interface (CUPI) API -- Search Space Member API

Search Space API

In Cisco Unity Connection, you create partitions as a way to group together objects to which callers and users can address messages or place calls while interacting with Connection. One or more partitions can be grouped together as members of a search space, and a partition can be a member of more than one search space. Search Space's Members provide the list of partitions and their respective search spaces so that search spaces can be queried on the basis of the partitions that it contains.

Listing the Search Spaces Members Based on Partition Object ID

The following is an example of the GET request that lists all the search space members:

```
GET:
https://<connectionserver>/vmrest/searchspacesmembers?query=(PartitionObjectId%20is%20partiti
onobjectid)
The following is the response from the above *GET* request and the actual response will
depend
upon the information given by you:
<SearchSpacesMembers total="2">
  <SearchSpacesMember>
     <URI>/vmrest/searchspacesmembers/ef101893-8687-49f6-b658-b05e110690d5</URI>
     <ObjectId> ef101893-8687-49f6-b658-b05e110690d5</ObjectId>
     <PartitionObjectId>a7108db5-c354-4b71-a72f-2c945291bda2</PartitionObjectId>
     <PartitionURI>/vmrest/partitions/a7108db5-c354-4b71-a72f-
    2c945291bda2</PartitionURI>
     <SearchSpaceObjectId>d4885446-a1f9-4e4c-810f-168bcc8489af</SearchSpaceObjectId>
     <SearchSpaceURI>/vmrest/searchspaces/d4885446-a1f9-4e4c-810f-
   168bcc8489af</SearchSpaceURI>
     <SortOrder>1</SortOrder>
  </SearchSpacesMember>
  <SearchSpacesMember>
     <URI>/vmrest/searchspacesmembers/b3431606-ccd2-4c06-9a12-9c336c78968c</URI>
     <ObjectId>b3431606-ccd2-4c06-9a12-9c336c78968c</ObjectId>
     <PartitionObjectId>a7108db5-c354-4b71-a72f-2c945291bda2</PartitionObjectId>
     <PartitionURI>/vmrest/partitions/a7108db5-c354-4b71-a72f-
    2c945291bda2</PartitionURI>
     <SearchSpaceObjectId>9b3eb53b-445f-416e-b5f3-
    15c64428d190</SearchSpaceObjectId>
     <SearchSpaceURI>/vmrest/searchspaces/9b3eb53b-445f-416e-b5f3-
    15c64428d190</SearchSpaceURI>
     <SortOrder>1</SortOrder>
  </SearchSpacesMember>
</SearchSpacesMembers>
```

```
Response Code: 200
```

JSON Example

To list all the search space members use the following command, do the following:

```
GET:
https://<connectionserver>/vmrest/searchspacesmembers?query=(PartitionObjectId%20is%20partitiono
bjectid)
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"
  "SearchSpacesMember":[
  {
     "URI":"/vmrest/searchspacesmembers/063ac262-41a9-41a2-810a-5380ffec322d"
     "ObjectId":"063ac262-41a9-41a2-810a-5380ffec322d"
     "PartitionObjectId":"97bf6afe-346e-4275-967e-43c50be79d32"
     "PartitionURI":"/vmrest/partitions/97bf6afe-346e-4275-967e-43c50be79d32"
     "SearchSpaceObjectId":"2e836e16-f715-4a18-bb7c-ee5e33281706"
     "SearchSpaceURI":"/vmrest/searchspaces/2e836e16-f715-4a18-bb7c-ee5e33281706"
     "SortOrder":"1"
  },
  {
     "URI":"/vmrest/searchspacesmembers/62a37620-50b0-4992-8c39-242cea8c7da9"
     "ObjectId":"62a37620-50b0-4992-8c39-242cea8c7da9"
     "PartitionObjectId":"44bdcf73-d5c1-4866-957d-fb35686cbe76"
     "PartitionURI":"/vmrest/partitions/44bdcf73-d5c1-4866-957d-fb35686cbe76"
     "SearchSpaceObjectId":"97efa220-31a7-4409-a0cd-2cdfc669182f"
     "SearchSpaceURI":"/vmrest/searchspaces/97efa220-31a7-4409-a0cd-2cdfc669182f"
     "SortOrder":"1"
  }
  ]
}
```

```
Response Code: 200
```

Explanation of Data Fields

Parameter	Operations	Data Type	Comments
URI	Read Only	String	URI of Search Space's member
ObjectId	Read Only	String (36)	Object Id of the search space's member
PartitionObjectId	Read Only	String (36)	Object Id of the partition.
PartitionURI	Read Only	String	URI of the partition.
SearchSpaceObjectId	Read Only	String (36)	Object Id of the search space