



Cisco Unity Connection Provisioning Interface (CUPI) API -- For End Users -- Directory

- [About the Directory, on page 1](#)
- [Listing and Viewing, on page 1](#)

About the Directory

A URI is provided for users to allow them to search the directory and find addressable objects that can be used to send messages, be added to a private list, and so on. The directory can be searched several different ways. The example below searches for names that start with "user":

```
GET /vmrest/directory/addressable?query=(name%20startswith%20user)
```

The directory can be searched by name or extension, or by both name and extension (to search by both name and extension, simply do not provide search criteria; see examples below). When searching by name, an alias, display name, first name, or last name on the object in question that matches the search term is treated as a match.

The directory can be searched in a "startswith" or "is" manner. A "startswith" search will match any field that starts with the search term, while an "is" search will match only those fields that exactly contain the search term.

The actual search argument can be either a string of all numbers, or a string of anything else. In most cases you would want to search by extension with numbers, and by name with alphanumeric characters.

All the matches are returned as Address objects. The Address objects contain all of the critical information that is needed to use these objects with other resources.

Listing and Viewing

Here is an example of a user search on a server with several user objects that match the search criteria:

```

GET /vmrest/directory/addressable?query=(name%20startswith%20user)
200
OK
<?xml version="1.0" encoding="UTF-8"?>
<Addresses total="2">
  <Address>
    <ObjectId>8dd65570-a1e8-4c85-95bc-0a8160877238</ObjectId>
    <Type>SUBSCRIBER</Type>
    <DisplayName>UserG</DisplayName>
    <SmtpAddress>userg@cuc-install-43.cisco.com</SmtpAddress>
    <DtmfAccessId>1019</DtmfAccessId>
  </Address>
  <Address>
    <ObjectId>7c897859-282f-43b0-9cab-1933e9b844f3</ObjectId>
    <Type>SUBSCRIBER</Type>
    <DisplayName>UserD</DisplayName>
    <SmtpAddress>userd@cuc-install-43.cisco.com</SmtpAddress>
    <DtmfAccessId>1018</DtmfAccessId>
  </Address>
</Addresses>

```

To search the directory for names that exactly match "user," the following query would be used:

```
GET /vmrest/directory/addressable?query=(name%is%20user)
```

To search the directory for names and extensions that start with "100," the following query would be used:

```
GET /vmrest/directory/addressable?query=(startswith%20100)
```

To search the directory for names and extensions that are exactly "100," the following query would be used:

```
GET /vmrest/directory/addressable?query=(is%20100)
```

To search the directory for extensions that start with "100," the following query would be used.



Note For performance reasons, we do not recommend this method for searching by extension. Doing a "startswith" search by extension should be done only when absolutely necessary.

```
GET /vmrest/directory/addressable?query=(extension%20startswith%20100)
```

To search the directory for an extension that exactly matches "100," the following query would be used.



Note This is the preferred method for searching by extension.

```
GET /vmrest/directory/addressable?query=(extension%20is%20100)
```

User accounts without mailboxes, such as the default administrative account, cannot use this directory URI. Instead, they should use the various query methods exposed on the various objects to search the entire server's directory. If an administrative user without a mailbox wants to use the directory to search, they may only do so by providing a user object ID. The user object ID represents the user who is doing the search. For example, the URI below is a search request for extension 1017 using the supplied user object ID; thus that user's search space is used.

```
GET  
/vmrest/directory/addressable?search=1017&userobjectid=58cdb193-7a7c-4cee-b0aa-e2e0cca22fec
```

Note that the maximum number of objects that can be returned from a search is 100 objects.

