

vCloud API Migration Guide

vCloud Director 1.5

This document supports the version of each product listed and supports all subsequent versions until the document is replaced by a new edition. To check for more recent editions of this document, see <http://www.vmware.com/support/pubs>.

EN-000697-00

vmware[®]

You can find the most up-to-date technical documentation on the VMware Web site at:

<http://www.vmware.com/support/>

The VMware Web site also provides the latest product updates.

If you have comments about this documentation, submit your feedback to:

docfeedback@vmware.com

Copyright © 2011 VMware, Inc. All rights reserved. This product is protected by U.S. and international copyright and intellectual property laws. VMware products are covered by one or more patents listed at <http://www.vmware.com/go/patents>.

VMware is a registered trademark or trademark of VMware, Inc. in the United States and/or other jurisdictions. All other marks and names mentioned herein may be trademarks of their respective companies.

VMware, Inc.
3401 Hillview Ave.
Palo Alto, CA 94304
www.vmware.com

Contents

vCloud API Migration Guide	5
----------------------------	---

vCloud API Migration Guide

This technical note summarizes changes made in the vCloud API between version 1.0 and version 1.5.

Schema and Operations Changes

Schema and operations changes are documented in a downloadable archive. To use the archive:

- 1 Download the compressed archive `rest-api-diff-html.zip` from <http://communities.vmware.com/community/vmttn/developer/forums/vcloudapi>.
- 2 Uncompress the archive into any convenient folder.
- 3 In the folder, open the file `index.html` in a browser.

The archive follows the formatting conventions used for the vCloud API Schema Reference download. These conventions are described in the *vCloud API Programming Guide*. Elements, types, and operations are divided into user, administrator, and extension categories. Each page of elements, types, or operations includes a table that contains links to the first version (vCloud API version 1.0) of the element, type, or operation, and the second version (vCloud API version 1.5), as well as a summary description of the changes between the two versions.

Changes in Object Identifiers

Object IDs included in href attribute values have changed. In API version 1.0, they were integers. In API version 1.5, they are universal unique identifiers (UUIDs) as defined by RFC 4122. For example, a vCloud API 1.0 URL of the form

`http://vcloud.example.com/api/v1.0/vApp/vapp-33`

might look like this in vCloud API 1.5.

`http://vcloud.example.com/api/v1.0/vApp/vapp-f5e185a4-7c00-41f1-8b91-0e552d538101`

Integer object IDs stored in the vCloud Director database are converted into UUIDs during product upgrade. The database retains a mapping of the old identifiers to the new ones, and the server typically returns a URL containing the new object identifier in response to a request that includes a URL containing an old one. See [“Server Processing of Old Style URLs,”](#) on page 6.

Changes in URLs

vCloud API client applications access vCloud API objects through an object reference in the form of a URL, which the server returns in the value of an href attribute in the XML representation of the object. Use of version 1.0 URLs in requests is supported, with limitations. See [“Server Processing of Old Style URLs,”](#) on page 6

In addition to the change in object ID format, vCloud API URLs no longer include an API version component. The /v1.0 component has been removed, and the API namespace in which a request should be processed is now indicated in an HTTP Accept header passed in the request, as shown in these two examples:

Request (API version 1.0)

```
POST https://vcloud.example.com/api/v1.0/vdc/5/action/instantiateVAppTemplate
Content-Type: application/vnd.vmware.vcloud.instantiateVAppTemplateParams+xml
...
```

Request (API version 1.5)

```
POST
https://vcloud.example.com/api/vdc/f5e185a4-7c00-41f1-8b91-0e552d538101/action/instantiateVAppTemplate
Content-Type: application/vnd.vmware.vcloud.instantiateVAppTemplateParams+xml
Accept: application/*+xml;version=1.5
...
```

Server Processing of Old Style URLs

If a client makes a request that includes a version 1.0 object identifier, the server validates that the integer object identifier is mapped to a UUID and returns HTTP response code 301 (moved permanently), along with a URL containing the new object identifier. Requests containing URLs that reference objects for which no mapping exists return HTTP response code 404 (not found).

This handling mechanism is applied to all old style URLs with the exception of those that are embedded in operations requests. For example, the `composeVApp` operation sends a `ComposeVAppParams` parameter in its request body. This element can contain multiple composition item references expressed as URLs. If any of these URLs are in the old style, the operation returns an HTTP 400 status code (bad request) with an `Error` element in the response body. This element lists the old-style URLs and their version 1.5 counterparts. You can use this information to create a new request that uses version 1.5 URLs.

The following operations exhibit this behavior:

```
instantiateVAppTemplate
captureVApp
cloneMedia
cloneVApp
cloneVAppTemplate
composeVApp
recomposeVApp
ejectMedia
insertMedia
```