

Oracle Flash Storage System

RESTful API Guide



FLASH STORAGE
SYSTEMS

Part Number E56929-01
Oracle FS1-2 Flash Storage System release 6.1
2015 April

Copyright © 2005, 2015, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

This software or hardware and documentation may provide access to or information on content, products and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Copyright © 2005, 2015, Oracle et/ou ses affiliés. Tous droits réservés.

Ce logiciel et la documentation qui l'accompagne sont protégés par les lois sur la propriété intellectuelle. Ils sont concédés sous licence et soumis à des restrictions d'utilisation et de divulgation. Sauf disposition de votre contrat de licence ou de la loi, vous ne pouvez pas copier, reproduire, traduire, diffuser, modifier, breveter, transmettre, distribuer, exposer, exécuter, publier ou afficher le logiciel, même partiellement, sous quelque forme et par quelque procédé que ce soit. Par ailleurs, il est interdit de procéder à toute ingénierie inverse du logiciel, de le désassembler ou de le décompiler, excepté à des fins d'interopérabilité avec des logiciels tiers ou tel que prescrit par la loi.

Les informations fournies dans ce document sont susceptibles de modification sans préavis. Par ailleurs, Oracle Corporation ne garantit pas qu'elles soient exemptes d'erreurs et vous invite, le cas échéant, à lui en faire part par écrit.

Si ce logiciel, ou la documentation qui l'accompagne, est concédé sous licence au Gouvernement des Etats-Unis, ou à toute entité qui délivre la licence de ce logiciel ou l'utilise pour le compte du Gouvernement des Etats-Unis, la notice suivante s'applique :

U.S. GOVERNMENT RIGHTS. Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

Ce logiciel ou matériel a été développé pour un usage général dans le cadre d'applications de gestion des informations. Ce logiciel ou matériel n'est pas conçu ni n'est destiné à être utilisé dans des applications à risque, notamment dans des applications pouvant causer des dommages corporels. Si vous utilisez ce logiciel ou matériel dans le cadre d'applications dangereuses, il est de votre responsabilité de prendre toutes les mesures de secours, de sauvegarde, de redondance et autres mesures nécessaires à son utilisation dans des conditions optimales de sécurité. Oracle Corporation et ses affiliés déclinent toute responsabilité quant aux dommages causés par l'utilisation de ce logiciel ou matériel pour ce type d'applications.

Oracle et Java sont des marques déposées d'Oracle Corporation et/ou de ses affiliés. Tout autre nom mentionné peut correspondre à des marques appartenant à d'autres propriétaires qu'Oracle.

Ce logiciel ou matériel et la documentation qui l'accompagne peuvent fournir des informations ou des liens donnant accès à des contenus, des produits et des services émanant de tiers. Oracle Corporation et ses affiliés déclinent toute responsabilité ou garantie expresse quant aux contenus, produits ou services émanant de tiers. En aucun cas, Oracle Corporation et ses affiliés ne sauraient être tenus pour responsables des pertes subies, des coûts occasionnés ou des dommages causés par l'accès à des contenus, produits ou services tiers, ou à leur utilisation.

Contents

- List of Tables5
- Preface6
 - Oracle Resources6
 - Command Syntax Conventions6
- Chapter 1: Welcome to the Oracle FS RESTful API8
 - Get Started with the Oracle FS System RESTful API8
 - RESTful API Authentication9
 - RESTful API Versions9
 - Common RESTful Operations9
 - Query Parameters.....10
- Chapter 2: Use the Oracle FS RESTful API11
 - Access the Service11
 - Oracle FS System Resources.....11
- Chapter 3: Oracle FS System RESTful API Requests15
 - Oracle FS System RESTful API Request Format.....15
 - HTTP Verbs17
- Chapter 4: Oracle FS System RESTful API Responses21
 - Response Message Bodies21
 - Return Codes23
- Index.....25

List of Tables

Table 1: Oracle resources.....	6
Table 2: Typography to mark command syntax.....	6
Table 3: Common RESTful operations.....	10
Table 4: Resources with IDs.....	11
Table 5: Resources that do not have IDs.....	13
Table 6: Partially-supported resources.....	13
Table 7: Mapping HTTP verbs to FSCLI subcommands.....	14
Table 8: Return codes.....	23

Preface

Oracle Resources

Table 1: Oracle resources

For help with...	Contact...
Support	http://www.oracle.com/support (www.oracle.com/support)
Training	https://education.oracle.com (https://education.oracle.com)
Documentation	<ul style="list-style-type: none">• Oracle Help Center: (http://www.oracle.com/goto/FSStorage/docs)• From Oracle FS System Manager (GUI): Help > Documentation• From Oracle FS System HTTP access: (http://system-name-ip/documentation.php where system-name-ip is the name or the public IP address of your system)
Documentation feedback	http://www.oracle.com/goto/docfeedback (http://www.oracle.com/goto/docfeedback)
Contact Oracle	http://www.oracle.com/us/corporate/contact/index.html (http://www.oracle.com/us/corporate/contact/index.html)

Command Syntax Conventions

Table 2: Typography to mark command syntax

Typographic symbol	Meaning
[]	Square brackets. Delimits an optional command parameter or a set of optional command parameters.
{ }	Braces. Delimits a set of command parameters, one of which must be selected.
	Vertical bar. Separates mutually exclusive parameters.
...	Ellipsis. Indicates that the immediately preceding parameter or group of parameters can be repeated.

Table 2: Typography to mark command syntax (continued)

Typographic symbol	Meaning
monospace	Indicates the name of a command or the name of a command option (sometimes called a <i>flag</i> or <i>switch</i>).
<i>italic</i>	Indicates a variable for which you need to supply a value.

Command parameters that are not enclosed within square brackets ([]) are required.

Important: The above symbols (and font styling) are based on the POSIX.1-2008 specification. These symbols are used in the command syntax only to clarify how to use the command parameters. *Do not enter these symbols on the command line.*

Welcome to the Oracle FS RESTful API

Get Started with the Oracle FS System RESTful API

You can monitor and manage an Oracle FS System using the Oracle FS System Representational State Transfer (REST) Application Programming Interface (API). Based on a layered client-server model, the RESTful architecture permits services to be transparently redirected through standard hubs, routers, and other network systems.

The core functionality of Oracle FS System RESTful API is built on top of the Oracle FS CLI (FSCLI) and the implementation maps to existing FSCLI commands. See the *Oracle Flash Storage System CLI Reference* on the [Oracle Help Center](http://www.oracle.com/goto/FSStorage/docs) (www.oracle.com/goto/FSStorage/docs) if you are not familiar with the FSCLI.

The Oracle FS System RESTful API includes the following features:

HTTP Verbs	Uses HTTP to implement REST. The RESTful API supports the following HTTP verbs: GET, POST, PUT, and DELETE. The GET, PUT, and DELETE verbs observe all HTTP semantics, such as idempotency.
Port Information	Accessible on the SSL secure port 8085.
Resources	Resources are the main abstraction in Oracle FS System RESTful API. Resources are anything that can have an ID or fully qualified name (FQN) associated with it. Only those resources that correspond to FSCLI commands are accessible.
CGI Query Strings	The HTTP GET verb does not support using a message body to provide additional options. Instead, the standard is to use CGI query string parameters. The Oracle FS System RESTful API supports the use of CGI query strings in the URI supplied with the GET command.
Request Formats	Supports both XML-formatted and JSON-formatted message bodies to specify the options contained in message bodies needed for POST, PUT, and some DELETE requests.
Response Formats	Responses from all Oracle FS System RESTful API GET, POST, PUT, or DELETE verbs are either an XML format or a JSON format.

RESTful API Authentication

The Oracle FS System RESTful API uses the same authentication credentials as the Oracle FS System Manager and Oracle FS CLI (FSCLI). All requests from external clients are individually authenticated using the appliance credentials and are conducted over an HTTPS connection on port 8085.

The Oracle FS System RESTful API uses the Apache HTTP header `Authentication` property. The value for the `Authentication` property follows the standard *basic authentication* pattern. The value is a Base64-encoded string consisting of the Oracle FS System administrator user ID concatenated with a colon (:), concatenated with the password Oracle FS System administrator. The resulting string is stored in the `Authentication` property in the HTTP header for the Oracle FS System RESTful API request.

Important:

Ensure that the Oracle FS System administrator that use has sufficient permissions to perform all tasks in the request. See the *Oracle Flash Storage System CLI Reference* on the [Oracle Help Center](http://www.oracle.com/goto/FSStorage/docs) (www.oracle.com/goto/FSStorage/docs) if you are not familiar with the FSCLI permissions.

The Oracle FS System performs a Base64 decode to extract the user ID and password and then authenticates permission before performing the requested action. The supported user IDs are any default IDs supported by the Oracle FS System as well as any IDs created by an Oracle FS System administrator.

The following is a basic authentication example:

```
Authorization: Basic dXN1cm5hbWU6cGFzc3dvcmQ=
```

RESTful API Versions

The RESTful API version for a given release has a global version number that you must include in the request.

This version number (`v1`) is included in all requests:

```
GET https://<oraclefs_ip_or_dns>:8085/v1/volume_group
```

Note: The RESTful API version for a given release for the API is not related to the Oracle FS System releases.

Common RESTful Operations

The Oracle FS System RESTful API employs the following HTTP verbs to implement Oracle FS System monitoring and management functions: GET, POST, PUT, and DELETE.

The following table shows the common RESTful operations for a given resource.

Table 3: Common RESTful operations

Request	Path	Description
GET	<resources>	List all <resources>.
GET	<resources>/<name>	Get an object describing the selected resource.
POST	<resources>	Create a new resource.
PUT	<resources>/<name>	Modify the selected resource.
DELETE	<resources>/<name>	Delete the selected resource.

Query Parameters

Some requests support optional query parameters that modify or enhance the data returned. Not every resource supports every query parameter.

The Oracle FS System RESTful API does not permit the following FSCLI options in message bodies or CGI query strings.

- -outputFormat or -o
- -sessionKey
- -u
- -p
- -oracleFS

If additional options are required to qualify the information being returned about the resource, these options are provided at the end of the URI as a query string (as CGI parameters).

Use the Oracle FS RESTful API

Access the Service

You access the service using a URL that contains the Oracle FS System IP or DNS name, the port number, and the version of the RESTful API.

To access the service, use this URL:

```
https://<oraclefs_ip_or_dns>:8085/v1/
```

Oracle FS System Resources

The Oracle FS System RESTful API provides a *resource* view of the world. A resource is any entity that is identified by an ID or fully qualified name (FQN).

The following Oracle FS System entities are accessible as resources with IDs or FQNs using the HTTP GET, PUT, POST, and DELETE commands. Unless otherwise noted, the IDs or FQNs are of the entities themselves.

Note: Your Oracle FS System might not support all of the resources listed.

Table 4: Resources with IDs

Resource	URI resource fragment	Notes
Account	/account	
Enclosure	/enclosure	
Cifs	/cifs	Using File Server as its ID or FQN
Cifs Share	/cifs_share	
Clone Filesystem	/clone_filesystem	
Clone LUN	/clone_lun	
Event notification	/event_notification	
File Server	/fileserv	
Filesystem	/filesystem	
Host group	/host_group	

Table 4: Resources with IDs (continued)

Resource	URI resource fragment	Notes
Host map	/hostmap	
iSCSI	/iscsi	Using the Controller ID or FQN to retrieve Controller-specific iSCSI settings
Job	/job	
Log Bundle	/system_log	
LUN	/lun	
NDMP	/ndmp	
NFS	/nfs	Using File Server as its ID or FQN
NFS Export	/nfs_export	
NFS Host	/nfs_host	
Profile	/profile	
Report	/report	
SAN host	/san_host	
Controller	/controller	
Snapshot	/snapshot	Using source filesystem ID or FQN
Snapshot Schedule	/snapshot_schedule	
SNMP Host	/snmp_host	
Storage Domain	/storage_domain /drive_group	
System Alert	/system_alert	
Task	/task	
UPS	/ups	
VIF	/vif	
Volume Group	/volume_group	

The following Oracle FS System resources are accessible as pseudo-resources as there is no ID required due to the global nature of the entity:

Table 5: Resources that do not have IDs

Resource	URI resource fragment	Notes
Call Home	/call_home	
Errors	/errors	
Event Log	/event_log	
Haltpoint	/haltpoint	
NAS	/nas	Retrieves NAS service status
Pilot	/pilot	
Quota	/quota	
Route	/route	
SAN	/san	
Software Update	/software_update	
Statistics	/statistics	
System	/system	
Time	/time	
Version	/version	

The following Oracle FS System resources are partially supported by the RESTful API:

Table 6: Partially-supported resources

Resource	URI resource fragment	Notes
Enclosure Console	enclosure_console	The API supports commands that open, close, write, and read the console. The API does not support the option to do a polling read.
Storage Allocation	storage_allocation	

The Oracle FS System RESTful API employs the following HTTP verbs to implement Oracle FS System monitoring and management functions: GET, POST, PUT, and DELETE. The following table shows how the HTTP verbs are mapped to the Oracle FS CLI subcommand calls:

Table 7: Mapping HTTP verbs to FSCLI subcommands

Action	HTTP verb	FSCLI subcommand	Notes
Create a resource	POST	-add	<p>The message body contains the options for creating the resource of the POST request. The message body is either an XML format or a JSON format and contains the minimum options required for the resource in question.</p> <p>The response is the ID and FQN of the created object.</p>
Modify a resource	PUT	-modify	<p>The URI must contain the resource ID. A message body must be provided with the PUT request and contain the properties that you want to change. This action must be idempotent.</p>
Delete a resource	DELETE	-delete	<p>If you provide the resource ID in the URI, the specified resource is deleted. If you do not provide the resource ID in the URI, you must provide a message body with the options that identify a collection of resources to delete. This action must be idempotent.</p>
Display resource by ID	GET	-list -details	<p>Retrieving by only specifying the ID in the URI returns the details of the resource. The basic FSCLI response is returned.</p>
Display a collection of resources	GET	-list	<p>By providing only the resource in the URI, the current FSCLI response is returned, namely, the ID and FQN for each resource instance.</p> <p>Any additional options supported by a given FSCLI command must be provided in a CGI query string format on the URI.</p> <p>This action must be idempotent.</p>
Perform a command	POST	Any commands that do an action.	<p>The resource ID may or may not be required based on the command. The command itself is included in the URI following the ID.</p>

Oracle FS System RESTful API Requests

Oracle FS System RESTful API Request Format

An Oracle FS System RESTful API request consists of the HTTP verb (GET, POST, PUT, or DELETE), a URI identifying a resource or set of resources, HTTP header field values, and an optional message body.

URI Format

The following example is the URI format for the Oracle FS System RESTful API HTTP verbs.

```
https://<oraclefs_ip_or_dns>:8085/v1/<resource_name>[/ID or FQN]
[/<subcommand>]
```

The optional ID or FQN values identify a specific resource instance and are provided based on the HTTP verb used. When specifying an FQN, escape the leading slash (/) with a backslash (\).

Only use the optional `/subcommand` value with the POST command to issue subcommands.

HTTP Header Fields

You can specify the HTTP `Content-Type`, `Accept`, and `Authentication` header field values in the Oracle FS System RESTful API request. The `Content-Type` field specifies the format of the message body included in the request. The `Accept` field specifies the message body format of the response. The two values supported for the `Content-Type` and the `Accept` fields are `text/xml` for XML formatting and `text/json` for JSON formatting. The Oracle FS System RESTful API supports basic authorization over HTTPS with each command execution.

Message Body

When the Oracle FS System RESTful API request is either a POST or a PUT, include a message body. When the RESTful API request is a DELETE request that deletes more than one resource, include a message body with the DELETE request to specify the criteria used for selecting the resources. Message bodies may be either an XML format or a JSON format.

The following example is a `text/xml` `Content-Type` message body.

```
<RequestBody>
  One or more element tags that map to FSCLI option names and
```

```
values.
</RequestBody>
```

A direct one-to-one mapping of FSCLI command options to the element tag names and values that are in the message body exists. Responses might contain a message body. The `Accept` HTTP header specifies the format for the contents in the message body.

The following example is a `text/xml` `Content-Type` response message body:

```
<RequestBody>
  One or more element tags containing FSCLI XML tags and values.
</RequestBody>
```

In the cases where prompt responses are required, respond to the prompt, and then resend the request with the prompt response in the request body. If there are multiple prompts required, the RESTful interface returns a failed response with a new prompt and the ID number of the failed response. Resend the previous command, appending another `<PromptResponse>` section until you finally send a request that has all the prompt responses accounted for.

The following example demonstrates a request that failed because it needed the user to reply to a prompt:

Note: See the following lines in the example:

```
<PromptRequired>Warning: Lun "414B303032323736A104781007D6A8DE"
has clones, if you
continue, they will all be deleted. If just a specific clone is
to be deleted, use clone_lun -delete.
Continue(y/N)? </PromptRequired>
```

Example

```
DELETE https://<oraclefs_ip_or_dns>:8085/v1/lun/
414B303032323736A104781007D6A8DE
Authentication: Basic <encoded username:password>
Content-type:text/xml
```

```
HTTP/1.1 409 Conflict
Date: Thu, 19 Mar 2015 22:04:56 GMT
Server: Apache/2.2.15 (Oracle)
Connection: close
Transfer-Encoding: chunked
Content-Type: text/xml
```

```
<?xml version="1.0"?>
<ResponseBody>
<CLIResponse>
<ResponseHeader>
<ClientData>FSCLI</ClientData>
<Command>lun</Command>
</ResponseHeader>
<PromptRequired>Warning: Lun "414B303032323736A104781007D6A8DE"
has clones, if you
continue, they will all be deleted. If just a specific clone is
to be deleted, use clone_lun -delete.
Continue(y/N)? </PromptRequired>
<PromptId>1</PromptId>
</CLIResponse>
</ResponseBody>
```

The following example demonstrates a resend of the request with the prompt response in the request body:

Note: See the following lines in the example:

```
<RequestBody><PromptResponse><id>1</id><response>y</response></
PromptResponse></RequestBody>
```

Example

```
DELETE https://<oraclefs_ip_or_dns>:8085/v1/lun/
414B303032323736A104781007D6A8DE
Authentication: Basic <encoded username:password>
Content-type:text/xml
```

```
<RequestBody><PromptResponse><id>1</id><response>y</response></
PromptResponse></RequestBody>
```

```
HTTP/1.1 200 OK
Connection: close
Date: Thu, 19 Mar 2015 22:04:56 GMT
Server: Apache/2.2.15 (Oracle)
Content-Type: text/xml
```

```
<?xml version="1.0"?>
<ResponseBody>
<CLIResponse>
  <ResponseHeader>
    <ClientData>FSCLI</ClientData>
    <Command>lun</Command>
  </ResponseHeader>
<TaskInformation>
  <TaskGuid>414B303032323736A13FC165BBA5CA84</TaskGuid>
  <TaskFqn>/DeleteLun/4416770/administrator</TaskFqn>
</TaskInformation>
<Status>succeeded</Status>
</CLIResponse>
</ResponseBody>
```

HTTP Verbs

The Oracle FS System RESTful API uses HTTP to implement REST. The RESTful API supports the following four HTTP verbs: GET, POST, PUT, and DELETE. The GET, PUT, and DELETE verbs observe all HTTP semantics, such as idempotency.

GET

Use the GET request to return one or more resources. If you do not provide the ID or the FQN in the request, only the ID and the FQN of all instances of the specified `resource_name` are returned. If you provide the ID or the FQN, the details for the specified instance are returned.

For example, to get the ID and the FQN for each LUN in the Oracle FS System, specify the LUN resource without any ID or the FQN:

```
GET https://<oraclefs_ip_or_dns>:8085/v1/lun HTTP/1.1
Authentication: Basic <encoded username:password>
Accept: text/xml
Content-Type: text/xml
```

To get the details for a specific LUN, include its ID or FQN, for example,

```
GET https://<oraclefs_ip_or_dns>:8085/v1/lun/\LUN_123 HTTP/1.1
Authentication: Basic <encoded username:password>
Accept: text/xml
Content-Type: text/xml
```

.

Note: The leading slash is escaped with a backslash.

If additional options are required to qualify the information being returned about the resource, provide the options at the end of the URI as a query string (as CGI parameters). For example, use the following request to get all of the offline volumes associated with a drive group:

```
GET https://<oraclefs_ip_or_dns>:8085/v1/enclosure/%5C/
ENCLOSURE-01?drivesmartdata=0 HTTP/1.1
Authentication: Basic <encoded username:password>
Accept: text/xml
Content-Type: text/xml
```

The GET request is always idempotent.

POST

Use the POST request to create a resource or to issue a command.

When the POST request is used to create a resource, the `ID` or the `FQN` is not required in the URI. Instead, provide a message body with the minimum set of options required to create the resource.

The following example creates a LUN using the POST request:

```
POST https://<oraclefs_ip_or_dns>:8085/v1/lun HTTP/1.1
Authentication: Basic <encoded username:password>
Accept: text/xml
Content-Type: text/xml
<RequestBody>
  <name>lun1</name>
  <addressableCapacity>100</addressableCapacity>
  <priority>medium</priority>
  <storageClass>hddLff</storageClass>
</RequestBody>
```

When the POST request is used to issue a command, the `ID` or the `FQN` is optional in the URI depending on the command being performed. The name of the command is included in the URI after the `ID` or the `FQN`. If the `ID` or the `FQN` is not required, the forward slash is still required. The message body is optional depending on the command. If the message body is provided, ensure that it contains any options needed by the command.

The following example requests a scan of a specific filesystem using the POST request:

```
POST https://<oraclefs_ip_or_dns>:8085/v1/filesystem/<id>/scan
HTTP/1.1
```

The following example restarts the Oracle FS System using the POST request.

Note: The `ID` is empty and the message body contains the additional options for the restart.

```
POST https://<oraclefs_ip_or_dns>:8085/v1/system//restart HTTP/
1.1
Authentication: Basic <encoded username:password>
Accept: text/xml
Content-Type: text/xml
<RequestBody>
  <serviceType>sanBias</serviceType>
  <overridePinnedData>1</overridePinnedData>
</RequestBody>
```

PUT

Use the PUT request to modify an existing resource or to modify global settings. If a resource is being modified, provide the `ID` or the `FQN`. The message body must be provided with the properties that are to be changed. The FSCLI option that identifies the `ID` or the `FQN` of the resource itself cannot be included in the message body and causes the request to fail with an HTTP return code of 400 (Bad Request).

The following example modifies the name of a LUN and Volume Group:

```
PUT https://<oraclefs_ip_or_dns>:8085/v1/lun/<id-of-lun> HTTP/1.1
Authentication: Basic <encoded username:password>
Accept: text/xml
Content-Type: text/xml
<RequestBody>
  <name>newName</name>
  <VolumeGroup>/otherVolGrp</VolumeGroup>
</RequestBody>
```

The following example modifies the global name of the Oracle FS System:

```
PUT https://<oraclefs_ip_or_dns>:8085/v1/system HTTP/1.1
Authentication: Basic <encoded username:password>
Accept: text/xml
Content-Type: text/xml
<RequestBody>
  <name>newOraclefsName</name>
</RequestBody>
```

The PUT request is idempotent except in the following cases:

- The `FQN` is used to identify the resource
- The name of the resource is being modified
- The `FQN` of the resource is composed using its name

DELETE

Use the DELETE request to delete either a specific resource instance or multiple resource instances. To delete a specific resource instance, provide the `ID` or the `FQN` of the resource. To delete multiple instances, instead of providing the `ID` or the `FQN` in the URI, include a message body that specifies one or more options that identify the instances to be deleted. The location in the URI where the `ID` or the `FQN` would normally be put is left blank. Several FSCLI commands provide alternative options for identifying a set of instances to delete. You specify those alternative options in the message body.

The following example deletes a specific LUN:

```
DELETE https://<oraclefs_ip_or_dns>:8085/v1/lun/<id-of-lun> HTTP/1.1
Authentication: Basic <encoded username:password>
Accept: text/xml
Content-Type: text/xml
```

The following example deletes multiple NFS exports that are accessible through a File Server.

```
DELETE https://<oraclefs_ip_or_dns>:8085/v1/nfs_export HTTP/1.1
Authentication: Basic <encoded username:password>
Accept: text/xml
Content-Type: text/xml
```

```
<RequestBody>  
  <fileServer>/server1</fileServer>  
</RequestBody>
```

Oracle FS System RESTful API Responses

Response Message Bodies

All Oracle FS System RESTful API responses contain a message body. At a minimum, the message body contains the `ID` and the `FQN` of the Oracle FS System task that performed or is performing the request. Depending on the request made, additional information is returned.

The value you specify for the `Accept` HTTP header field in the request determines the format of the message body that is returned. The content of the message body contains the same XML tags or JSON objects as what the Oracle FS CLI returns for the equivalent command request minus the `ResponseHeader` set of tags.

The following example is an XML message body:

```
<ResponseBody>
  <CLIResponse>
    <TaskInformation>
      <TaskGuid>aGuid</TaskGuid>
      <TaskFqn>/someFQN</TaskFqn>
    </TaskInformation>
    Tags and values specific to the request.
  </CLIResponse>
</ResponseBody>
```

The following example is a JSON message body:

```
{ "ResponseBody" : {
  "CLIResponse" : {
    "TaskInformation" : {
      "TaskGuid" : "aGuid",
      "TaskFqn" : "/someFQN"
    },
    JSON objects specific to the request
  }
}
```

For example, if these volume groups `/finance` and `/legal` are defined on the Oracle FS System, the XML-based request and response for retrieving all volume groups would be the following.

Note: The default volume group (`/`) is included in the response.

For XML, the message body looks like the following example:

```
Request:
GET https://<oraclefs_ip_or_dns>:8085/v1/volume_group HTTP/1.1
Accept: text/xml
Content-Type: text/xml
```

```

Response:
HTTP/1.1 200 OK
Date: <some date and time>
Connection: close
Content-Type: text/xml

<ResponseBody>
  <CLIResponse>
    <TaskInformation>
      <TaskGuid>4130303030303142A13F11D5EB113971</TaskGuid>
      <TaskFqn>/GetVolumeGroup/3901/administrator</TaskFqn>
    </TaskInformation>
    <VolumeGroup>
      <Fqn>/</Fqn>
      <Id>4130303030303142A10411D05C5B4344</Id>
    </VolumeGroup>
    <VolumeGroup>
      <Fqn>/finance</Fqn>
      <Id>4130303030303142A10411D1CE7C4D78</Id>
    </VolumeGroup>
    <VolumeGroup>
      <Fqn>/legal</Fqn>
      <Id>4130303030303142A10411D1CE7C4C55</Id>
    </VolumeGroup>
  </CLIResponse>
</ResponseBody>

```

For JSON, the message body looks like the following example:

```

Request:
GET https://<oraclefs_ip_or_dns>:8085/v1/volume_group HTTP/1.1
Accept: text/json
Content-Type: text/json

```

```

Response:
HTTP/1.1 200 OK
Date: <some date and time>
Connection: close
Content-Type: text/xml

```

```

{"ResponseBody" : {
  "CLIResponse" : {
    "TaskInformation" : {
      "TaskGuid" : "aGuid",
      "TaskFqn" : "/someFQN"
    },
    "VolumeGroup" : [
      {"Fqn" : "/",
        "Id" : "4130303030303142A10411D05C5B4344"
      },
      {"Fqn" : "/finance",
        "Id" : "4130303030303142A10411D1CE7C4D78"
      },
      {"Fqn" : "/legal",
        "Id" : "4130303030303142A10411D1CE7C4C55"
      }
    ]
  }
}
}

```

Return Codes

Oracle FS System RESTful API requests return HTTP status codes to indicate either successful completion of the request, or problems incurred when fulfilling the request.

The following table lists the supported codes.

Table 8: Return codes

Code	Definition	POST	PUT	GET	DELETE
200	OK Standard response for successful HTTP requests	Not applicable for creating new resources Supported for other commands	Supported A resource has been modified	Supported A resource has been returned	Supported A resource has been deleted
201	Created The request has been fulfilled and resulted in a new resource being created	Supported A new resource has been created	Not applicable	Not applicable	Not applicable
400	Bad Request The request cannot be fulfilled due to bad syntax	Supported	Supported	Supported	Supported
401	Unauthorized Authentication has failed	Supported User account is not authorized to perform request	Supported User account is not authorized to perform request	Supported User account is not authorized to perform request	Supported User account is not authorized to perform request
404	Not Found The requested resource could not be found	Supported when performing a command against a specific resource that is not found Not applicable when creating new resource	Supported	Supported	Supported

Table 8: Return codes (continued)

Code	Definition	POST	PUT	GET	DELETE
405	<p>Method not Allowed</p> <p>A request was made of a resource using a request method not supported by that resource</p>	Supported	Not applicable	Not applicable	Supported
409	<p>Conflict</p> <p>The request could not be completed due to a conflict with the current state of the resource</p>	<p>Supported</p> <p>A request was made when a similar request is already being processed</p> <p>This is also returned if the request prompts the user for a response</p>	Not applicable	Not applicable	Not applicable
500	<p>Internal Server Error</p> <p>A generic error message, given when no more specific message is suitable</p>	Supported	Supported	Supported	Supported
503	<p>Service Unavailable</p> <p>The server is unavailable</p>	Supported	Supported	Supported	Supported

Index

A

- access the service
 - URL defined [11](#)
- API versions
 - described [9](#)
- authentication
 - Base64 described [9](#)

B

- Base64
 - authentication described [9](#)

C

- CGI, *see* query parameters
- contact information [6](#)
- contacts, Oracle [6](#)
- conventions
 - command syntax [6](#)
- customer support [6](#)

D

- DELETE verb
 - defined [17-19](#)
- documentation
 - feedback [6](#)

E

- education programs [6](#)

F

- feedback, documentation [6](#)

G

- GET verb
 - defined [17-19](#)

H

- HTTP verbs
 - defined [17-19](#)

O

- online help [6](#)
- Oracle documentation [6](#)
- Oracle FS System RESTful API
 - introduction [8](#)
- Oracle Help Center (OHC) [6](#)

P

- POSIX.1-2008 specification [6](#)
- POST verb
 - defined [17-19](#)
- product support [6](#)
- PUT verb
 - defined [17-19](#)

Q

- query parameters
 - defined [10](#)

R

- request format
 - defined [15](#)
- response message bodies
 - defined [21](#)
- RESTful operations
 - common [9](#)
- return codes
 - defined [23](#)

S

- sales information [6](#)
- Support portal [6](#)
- syntax conventions [6](#)
- system resources
 - defined [11](#)

T

- training programs [6](#)