

Oracle® WebCenter Sites

Web Experience Management Framework
REST API Resource Reference

11g Release 1 (11.1.1)

April 2012

Oracle® WebCenter Sites: Web Experience Management Framework REST API Resource Reference, 11g Release 1 (11.1.1)

Copyright © 2012 Oracle and/or its affiliates. All rights reserved.

Primary Author: Tatiana Kolubayev

Contributing Author: Eric Gandt, Saikat Chaudhuri, Ravi Khanuja

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 America, 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.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

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.

Table of Contents

About This Guide	7
Audience	7
Related Documents	7
Conventions	7
Third-Party Libraries	8
REST API	9
Overview	10
REST Resources	10
<i>Asset Type</i>	11
All Asset Types	11
/types	
Selected Asset Type	11
/types/{assettype}	
All Subtypes	11
/types/{assettype}/subtypes	
Selected Subtype	11
/types/{assettype}/subtypes/{subtype}	
Site-Enabled Asset Types	12
/sites/{sitename}/types	
Site-Specific Asset Type Search	12
/sites/{sitename}/types/{assettype}/search	
Global Asset Type Search	12
/types/{assettype}/search	
<i>Asset</i>	14
Selected Asset, Selected Site	14
/sites/{sitename}/types/{assettype}/assets/{id}	

All Asset Associations, Selected Site	14
/sites/{sitename}/types/{assettype}/assets/{id}/associations	
Selected Asset Association and Site	14
/sites/{sitename}/types/{assettype}/assets/{id}/associations/{assocName}	
Site-Specific Assets Search	15
/sites/{sitename}/search	
Global Assets Search	15
/search	
<i>Index</i>	17
Indexing Configurations	17
/indexes	
Indexing Configuration	17
/indexes/{source}	
<i>Site</i>	18
All Sites	18
/sites	
Selected Site.	18
/sites/{sitename}	
<i>Site Plan Navigation</i>	18
Publication Node	18
/sites/{sitename}/navigation	
Page Node	19
/sites/{sitename}/navigation/{pageid}	
<i>Role</i>	20
All Roles	20
/roles	
Selected Role	20
/roles/{rolename}	
<i>Application</i>	20
All Applications.	20
/applications	
Selected Application	20
/applications/{applicationid}	
<i>User</i>	21
All Users	21
/users	
Selected User	21
/users/{username}	
All Site Users.	22
/sites/{sitename}/users	

Selected Site User	22
/sites/{sitename}/users/{username}	
<i>User Locale</i>	23
User Locales	23
/userlocales	
<i>User Definition</i>	23
User Def.	23
/userdef	
<i>ACLs</i>	24
ACLs	24
/acls	
<i>Groups</i>	24
All Groups	24
/groups	
Selected Group.	24
/groups/{groupname}	
<i>Geographical Location.</i>	24
Time Zone	24
/timezone	

About This Guide

This guide describes the REST API provided with the Oracle WebCenter Sites: Web Experience Management (WEM) Framework. The REST API is used to access Oracle WebCenter Sites data.

Applications discussed in this guide are former FatWire products. Naming conventions are the following:

- *Oracle WebCenter Sites* is the current name of the application previously known as *FatWire Content Server*. In this guide, *Oracle WebCenter Sites* is also called *WebCenter Sites*.
- *Oracle WebCenter Sites: Web Experience Management Framework* is the current name of the application previously known as *FatWire Web Experience Management Framework*. In this guide, *Oracle WebCenter Sites: Web Experience Management Framework* is also called *Web Experience Management Framework*, or *WEM Framework*.

Audience

This guide is intended for general administrators and developers of the WEM Framework.

Related Documents

For more information, see the following documents:

- *Oracle WebCenter Sites Developer's Guide*
- *Oracle WebCenter Sites Developer's Guide for the Web Experience Management Framework*

Conventions

The following text conventions are used in this guide:

- **Boldface** type indicates graphical user interface elements that you select.
- *Italic* type indicates book titles, emphasis, or variables for which you supply particular values.

- `Monospace` type indicates file names, URLs, sample code, or text that appears on the screen.
- **Monospace bold** type indicates a command.

Third-Party Libraries

Oracle WebCenter Sites and its applications include third-party libraries. For additional information, see *Oracle WebCenter Sites 11gR1: Third-Party Licenses*.

REST API

- [Overview](#)
- [REST Resources](#)

Overview

Oracle WebCenter Sites: Web Experience Management (WEM) Framework provides a REST API for accessing WebCenter Sites data. REST services are supported on both WebCenter Sites and Satellite Server to leverage the Satellite Server cache.

The default location of REST resources is under the WebCenter Sites BaseURI:

```
<protocol>://<hostname:port>/<servletPath>/REST/
```

For example, on a local installation the BaseURI could be the following:

```
http://localhost:8080/cs/REST/
```

The `<BaseURI>/application.wadl` file lists the resources described in this reference.

REST Resources

This section lists supported REST resources and methods, including error codes that can be returned if a resource does not exist or unforeseen errors occur.

Each resource has two types of views: list and detail. The list view lists resources of a given type (`users` for example) and for each resource, an `href` pointing to the detailed view. For example, the resource `<BaseURI>/users` returns a list view naming all users in the system, among them `fwadmin`.

```
<users>
```

```
<href>http://localhost:8080/cs/REST/users/fwadmin</href>
```

```
<name>fwadmin</name>
```

```
</users>
```

The `fwadmin` entry contains an `<href>` that specifies the URL of the detail view where more information about `fwadmin` can be found.

Except for `/timezone` and `/userlocales`, all other REST resources are protected. For information about REST resource security, see the WEM Framework developer and administrator guides.

Note

By default, if an authorization failure occurs, then the login page for Central Authentication Service (CAS) is displayed. If you want to receive a 500 error instead, add `auth-redirect=false` to the URL when making the request.

Asset Type

All Asset Types <code>/types</code>	
Description	This resource reads the list of asset types in the system.
REST Methods	GET
Possible Error Codes	500 if unforeseen error occurred

Selected Asset Type <code>/types/{assettype}</code>	
Description	This resource displays detailed information about the specified asset type.
REST Methods	GET, PUT, DELETE
Possible Error Codes	404 if asset type does not exist 500 if unforeseen error occurred

All Subtypes <code>/types/{assettype}/subtypes</code>	
Description	This resource reads all subtypes of the specified asset type.
REST Methods	GET
Possible Error Codes	404 if asset type does not exist 500 if unforeseen error occurred

Selected Subtype <code>/types/{assettype}/subtypes/{subtype}</code>	
Description	This resource reads the specified subtype of the specified asset type.
REST Methods	GET
Possible Error Codes	404 if asset type or subtype does not exist 500 if unforeseen error occurred

Site-Enabled Asset Types`/sites/{sitename}/types`

Description	This resource lists all asset types on the specified site.
REST Methods	GET
Possible Error Codes	404 if site does not exist 500 if unforeseen error occurred

Site-Specific Asset Type Search`/sites/{sitename}/types/{assettype}/search`

Description	<p>This resource supports searches on the specified asset type on the specified site.</p> <p>Note: Access to any search is provided by granting LIST privileges on an Asset resource. In this case, search access can be configured per site or for all sites, but not per asset type. All users with access to the search service will be able to search assets of all asset types (in a particular site, if one is specified).</p>
REST Methods	GET
Possible Error Codes	404 if site does not exist

Global Asset Type Search`/types/{assettype}/search`

Description	<p>This resource supports searches on the specified asset type.</p> <p>For this resource to work, the asset type index must be configured in WebCenter Sites (otherwise, an exception is thrown and the 404 error code is displayed to the user).</p> <p>Note: Access to any search is provided by granting LIST privileges on an Asset resource. In this case, search access can be configured per site or for all sites, but not per asset type. All users with access to the search service will be able to search assets of all asset types (in a particular site, if one is specified).</p>
Query Parameters	<p>q</p> <p>Specify a query String, the search engine-specific query that was specified during the index configuration. (The default search engine is Lucene.)</p>

Global Asset Type Search *(continued)*`/types/{assettype}/search`**field: [fieldname] : [Operation]**

Specify a query condition for a field of [fieldname]. [Operation] is optional and can be one of equals, range, startswith, phrase, wildcard, similar, or contains; the default is contains. For the range operation, the value's format is {lower bound} : {upper bound}, e.g., 1:10. Only the first value specified will be used in the search.

startindex

Specify the starting index from where the results will be returned.

count

Specify the total number of items to be returned.

sortfield: [fieldname] : [asc/des]

Specify the field on which the results would be sorted.

Note: For Lucene, sorting is possible only fields that were indexed as "not tokenized".

Example:

```
http://localhost:8080/cs/REST/types/Content_C/
search?field:name:contains=FSII&startindex=
5&count=3
```

REST Methods	GET
Possible Error Codes	404 if the indexing configuration does not exist 500 if unforeseen errors occurred

Asset

Selected Asset, Selected Site

/sites/{sitename}/types/{assettype}/assets/{id}

Description	This resource provides developers with Create, Edit, Update, Delete (CRUD) functions for operating on the specified asset. {id} is in the long asset id format.
REST Methods	HEAD, GET, POST, PUT, DELETE
Possible Error Codes	404 if site does not exist 404 if asset type does not exist 404 if asset does not exist 500 if unforeseen error occurred

All Asset Associations, Selected Site

/sites/{sitename}/types/{assettype}/assets/{id}/associations

Description	This resource lists all asset associations on the specified site.
REST Methods	GET
Possible Error Codes	404 if site does not exist 404 if asset type does not exist 404 if asset does not exist 500 if unforeseen error occurred

Selected Asset Association and Site

/sites/{sitename}/types/{assettype}/assets/{id}/associations/{assocName}

Description	This resource lists the specified asset association on the specified site.
REST Methods	GET
Possible Error Codes	404 if site does not exist 404 if asset type does not exist 404 if asset does not exist 500 if unforeseen error occurred

Site-Specific Assets Search

/sites/{sitename}/search

Description	<p>This resource supports searches on all assets on the specified site.</p> <p>Note: Access to any search is provided by granting <code>LIST</code> privileges on an <code>Asset</code> resource. In this case, search access can be configured per site or for all sites, but not per asset type. All users with access to the search service will be able to search assets of all asset types (in a particular site, if one is specified).</p>
REST Methods	GET
Possible Error Codes	404 if site does not exist

Global Assets Search

/search

Description	<p>This resource supports global searches on assets.</p> <p>Note: REST does not restrict the types of assets that are returned. As a result, assets of a certain type can be returned to a user who may otherwise not have permission to access the asset type.</p> <p>Note: Access to any search is provided by granting <code>LIST</code> privileges on an <code>Asset</code> resource. In this case, search access can be configured per site or for all sites, but not per asset type. All users with access to the search service will be able to search assets of all asset types (in a particular site, if one is specified).</p>
Query Parameters	<p>q Specify a query String, the search engine-specific query that was specified during the index configuration. (The default search engine is Lucene.)</p> <p>field: [fieldname] : [Operation] Specify a query condition for field of [fieldname] with the specified operation. [Operation] is optional and can be one of equals, range, startswith, phrase, wildcard, similar, or contains; the default is contains. For the range operation, the value's format is {lower bound} : {upper bound}, e.g., 1:10.</p> <p>startindex Specify the starting index from where the results will be returned.</p> <p>count Specify the total number of items to be returned.</p>

Global Assets Search *(continued)*

/search

	<p>sortfield: [fieldname] : [asc/des] Specify the field on which the results would be sorted.</p> <p>Note: For Lucene, sorting is possible only fields that were indexed as “not tokenized”.</p>
	<p>Example: http://localhost:8080/cs/REST/types/Content_C/search?field:name:contains=FSII&startindex=5&count=3</p>
REST Methods	GET
Possible Error Codes	500 if unforeseen error occurred

Index

Indexing Configurations

/indexes

Description	This resource lists configurations for global and asset type-based indexing. Global indexing produces a single index of asset types. Asset type indexing produces one index per asset type.
REST Methods	GET, PUT, POST, DELETE
Possible Error Codes	404 if indexing configuration does not exist 500 if unforeseen error occurred

Indexing Configuration

/indexes/{source}

Description	<p>This resource supports the configuration of global indexing and provides CRUD functions for operating on global and per asset type indexing configurations. Global indexing produces a single index of asset types. Asset type indexing produces one index per asset type.</p> <p>Note: In this resource's URL, source is either <code>Global</code> (for the global index) or the name of the asset type (for each asset type-based index). For asset type-based indexing, the indexing process will be also started, stopped, paused, or resumed automatically depending on your choice of HTTP method (<code>PUT</code>, <code>POST</code>, <code>DELETE</code>) and index status (<code>ENABLED</code>, <code>PAUSED</code>). Global indexing is enabled for all asset types by default, although the configuration may not exist.</p>
REST Methods	GET, POST, PUT, DELETE
Possible Error Codes	404 if indexing configuration does not exist 500 if unforeseen error occurred

Site

All Sites <code>/sites</code>	
Description	This resource lists all sites in the system.
REST Methods	GET
Possible Error Codes	500 if unforeseen error occurred

Selected Site <code>/sites/{sitename}</code>	
Description	<p>This resource provides developers with CRUD functions for operating on the specified site.</p> <p>To perform CRUD operations on a site, the user must have either the GeneralAdmin or SiteAdmin role on that site in addition to the privileges granted via groups. To manage AdminSite, a user must have the GeneralAdmin role.</p>
REST Methods	HEAD, GET, POST, PUT, DELETE
Possible Error Codes	404 if site does not exist 500 if unforeseen error occurred

Site Plan Navigation

Publication Node <code>/sites/{sitename}/navigation</code>	
Description	<p>This resource reads the Site Plan tree for the site specified by <code>{sitename}</code>. Accessing this resource requires having READ permission on the corresponding site resource. Query parameters defined for the service are as follows:</p> <p><code>depth</code>: Controls the depth to which the Site Plan tree can be read. Default value: 1 Accepted values: Integers or the string <code>all</code> (case-insensitive) to read up to the maximum depth.</p> <p><code>code</code>: Controls the type of site plan nodes (placed or unplaced) to be read. By default, both placed and unplaced pages are read back. The value for this parameter is case-insensitive.</p> <p>For more information about the Site Plan tree, see the <i>Oracle WebCenter Sites Developer's Guide</i>.</p>

Publication Node *(continued)*

/sites/{sitename}/navigation

REST Methods	GET
Possible Error Codes	<p>400 Exception in parsing depth. Depth is less than or equal to 0.</p> <p>403 User does not have permission.</p> <p>404 Site does not exist.</p> <p>500 No site plan publication node for the site. Any other unforeseen error.</p>

Page Node

/sites/{sitename}/navigation/{pageid}

Description	<p>This resource reads the Site Plan sub-tree for the site specified by {sitename} and page specified by {pageid} as the root. Accessing this resource requires having READ permission to at least the page asset resource that is specified by pageid. Query parameters defined for the service are as follows:</p> <p>depth: Controls the depth to which the Site Plan sub-tree can be read.</p> <p>Default value: 1</p> <p>Accepted values: Integers or the string all (case-insensitive) to read up to the maximum depth.</p> <p>For more information about the Site Plan tree, see the <i>Oracle WebCenter Sites Developer's Guide</i>.</p>
REST Methods	GET
Possible Error Codes	<p>400 Page asset type is not enabled on site. Exception in parsing depth. Depth is less than or equal to 0.</p> <p>403 User does not have permission.</p> <p>404 Either the site and/or page asset does not exist.</p> <p>500 Node path is null for page. No node elements exist in node path for page. Root node is not a publication node for page. Inconsistent child node. Any other unforeseen error.</p>

Role

All Roles <code>/roles</code>	
Description	This resource reads the list of roles that are stored in the system.
REST Methods	GET
Possible Error Codes	500 if unforeseen error occurred

Selected Role <code>/roles/{rolename}</code>	
Description	This resource provides CRUD functions for operating on the specified role.
REST Methods	GET, POST, PUT, DELETE
Possible Error Codes	404 if role does not exist 500 if unforeseen error occurred

Application

All Applications <code>/applications</code>	
Description	This resource lists registered applications.
REST Methods	GET
Possible Error Codes	500 if unforeseen error occurred

Selected Application <code>/applications/{applicationid}</code>	
Description	This resource registers the specified application by performing CRUD operations on the application's <code>FW_View</code> and <code>FW_Application</code> assets. More information about application registration is available in the <i>Oracle WebCenter Sites Developer's Guide for the Web Experience Management Framework</i> .
REST Methods	GET, PUT, POST, DELETE
Possible Error Codes	404 if application does not exist 500 if unforeseen error occurred

User

All Users	
<code>/users</code>	
Description	This resource lists all users that are defined in the system.
REST Methods	GET
Possible Error Codes	500 if unforeseen error occurred

Selected User	
<code>/users/{username}</code>	
Description	<p>This resource provides <code>CRUD</code> functions for operating on the specified user. If the user profile contains an image, a 90 x 90 pixel thumbnail image is returned by the service. If the thumbnail image is required in a different size, the request can be made with the following parameters: <code>imageHeight</code> and <code>imageWidth</code> (in pixels); the thumbnail image is then returned at the specified size. The actual uploaded image is always returned in its original size when user information is requested. To restrict the user service to return only the thumbnail image and not the actual image, set the <code>thumbOnly</code> parameter to <code>true</code>.</p> <p>The “otherAttributes” field provides placeholders for custom user attributes, which you define when creating a WebCenter Sites user either in LDAP or in the WebCenter Sites database. (Note that for LDAP, each user will be a descendant of the value that is specified for the <code>peopleParent</code> property in <code>dir.ini</code>. The <code>peopleParent</code> property value is set during integration with LDAP.)</p> <p>Regardless of where custom user attributes are created, to take effect in the WebCenter Sites application they need to be specified in the WebCenter Sites <code>dir.ini</code> file, in the <code>requiredPeopleAttrs</code> property as a key-value pair in the following format:</p> <pre>requiredPeopleAttrs= attribute1=description1&attribute2=description2</pre> <p>The key is the attribute name as it appears in the WebCenter Sites database (or in LDAP), and the value is the attribute’s description as it appears in the WebCenter Sites interface. The property will be looked up by both the WebCenter Sites Admin interface and REST API in defining user attributes.</p> <p>For example, if you want to specify two attributes called “Phone number” and “Fax” for all users, do the following:</p> <ol style="list-style-type: none"> 1. Define two attributes called “Phone number” and “Fax”. If you are using WebCenter Sites, define them as user attributes (they are stored in the <code>SystemUserAttr</code> table).

Selected User *(continued)*`/users/{username}`

Description	<ol style="list-style-type: none"> In the WebCenter Sites <code>dir.ini</code> file, provide the attribute references in the <code>requiredPeopleAttrs</code> property as follows: <code>requiredPeopleAttrs=phone=Phone number&fax=Fax</code> The above configuration allows <code>Phone number</code> and <code>Fax</code> to be displayed as attributes in the user account form in the WebCenter Sites Admin interface and the WEM Admin interface. If you create a user with <code>Phone number:12345678</code> and <code>Fax:23456789</code>, invoking a REST request on the user URI <code>/users/{username}</code> provides the following information for the “otherAttributes” field: <pre><otherAttributes> <name>phone</name> <value>12345678</value> </otherAttributes> <otherAttributes> <name>fax</name> <value>23456789</value> </otherAttributes></pre>
REST Methods	HEAD, GET, POST, PUT, DELETE
Possible Error Codes	404 if user does not exist 500 if unforeseen error occurred

All Site Users`/sites/{sitename}/users`

Description	This resource lists the roles of all users on the specified site.
REST Methods	GET
Possible Error Codes	404 if site does not exist 500 if unforeseen error occurred

Selected Site User`/sites/{sitename}/users/{username}`

Description	This resource lists the roles of the specified user on the specified site and supports the REST methods listed below.
REST Methods	GET
Possible Error Codes	404 if site does not exist 404 if user does not exist 500 if unforeseen error occurred

User Locale

The `/userlocales` resource is not protected

User Locales <code>/userlocales</code>	
Description	This resource lists all locales that are defined in the system.
REST Methods	GET
Possible Error Codes	500 if unforeseen error occurred

User Definition

User Def <code>/userdef</code>	
Description	This resource reads attributes in the user profile. When the database is used for authentication, only the WebCenter Sites predefined user attributes are returned. If LDAP is used, the service returns the attributes that are defined in the <code>requiredPeopleAttrs</code> property of the <code>dir.ini</code> file, along with the predefined WebCenter Sites attributes.
REST Methods	GET
Possible Error Codes	500 if unforeseen error occurred

ACLs

ACLs <code>/acls</code>	
Description	This resource lists all ACLs that are defined in WebCenter Sites.
REST Methods	GET
Possible Error Codes	500 if unforeseen error occurred

Groups

All Groups <code>/groups</code>	
Description	This resource lists all REST security groups in the system.
REST Methods	GET
Possible Error Codes	500 if unforeseen error occurred

Selected Group <code>/groups/{groupname}</code>	
Description	This resource provides the security privilege configuration of the specified group.
REST Methods	GET
Possible Error Codes	404 if requested group is not found

Geographical Location

The `/timezone` resource is not protected.

Time Zone <code>/timezone</code>	
Description	This resource returns the server's time zone.
REST Methods	GET
Possible Error Codes	500 if unforeseen error occurred