



Agile Product Lifecycle Management

Recipe & Material Workspace Export/Import Guide

v9.3.1

Part No. E16510-01

September 2010

Oracle Copyright

Copyright © 1995, 2010, 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 software or related documentation 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 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 which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

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

This software 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. The RMW product includes software developed by the Visigoth Software Society.

CONTENTS

Oracle Copyright.....	ii
Chapter 1 About Export/Import Tool.....	1
Overview.....	1
RMW Administrator Objects	2
Assumptions and Constraints.....	2
Chapter 2 Installation and Configuration	3
Directory Structure.....	3
Configuring Export/Import Tool	3
Updating the CFMConfig.xml Configuration File	4
Updating the Export or Import Command Files	4
Chapter 3 Schema, Criteria, and Criteria Properties Files	7
Schema Files.....	7
When to Change Values in the Schema File.....	7
Example Schema File.....	8
Schema File Details.....	9
Updating Template Files	11
Optional: Updating the Schema Template File for Cache Refresh.....	11
Criteria Files	12
Example Criteria File	12
Criteria File Details	14
Criteria Properties Files.....	14
Example Criteria Properties File.....	15
Criteria Properties File Details.....	16
Chapter 4 Exporting and Importing Data from the Command Line.....	19
Export/Import Parameters	19
Export Parameters.....	19
Import Parameters.....	20
How to Export	21
Log File.....	22
How to Import	22
Modes of Import.....	23
Import Failure	23
Rollback.....	23
Log Files.....	23

Chapter 5 Exporting Data with the User Interface	25
Supported Operations and Formats	25
Searching and Selecting the Export Data	25
Exportable Search Results	26
Chapter 6 Scenario Examples	29
Views	29
Using Search Attributes to Export Views	29
Inserting New Views	30
Updating Views	31
Deleting Views	32
View Attributes	33
Using Search Attributes to Export Views	33
Adding Additional Attributes to Existing Views	33
Updating Existing View Attributes	34
Deleting Existing View Attributes From Views	35
Categories	36
Inserting a New Category	36
Updating an Existing Category	37
Deleting an Existing Category	37
Adding Additional Attributes To Existing Categories	38
Updating Existing Category Attributes	39
Deleting Existing Category Attributes	40
Buttons	41
Inserting New Buttons	41
Updating Existing Buttons	42
Deleting Existing Buttons	43
Renaming Existing Buttons	44
Component Buttons	46
Using Search Attributes to Export Button-Component Links	46
Linking Existing Buttons to a Component	47
Step 1: Export existing component button link(s) from source RMW application.	47
Step 2: Import the exported links to the destination RMW application, using the following	
command:	47
Step 3: Check the newly added component button links through the RMW Web user	
interface.	47
Web Services	48
Using Search Attributes to Export Web Services	48
Inserting New Web Services	48
Updating Existing Web Services	49

Deleting Existing Web Services.....	50
Web Service Operations	51
Using Search Attributes to Export Web Service Operations	51
Inserting New Web Service Operations.....	51
Updating Existing Web Service Operations.....	52
Deleting Existing Web Service Operations.....	53
Action-to-Web-Service Operations	54
Using Search Attributes to Export Action to Web Service Links.....	54
Linking Button Callback To Web Service Operation.....	54
Step 1: Export existing 'Web service operation-action link' from source RMW application ...	54
Step 2: Import the exported Web service operation to action link to destination RMW	
application, using the following command:	55
Step 3: Check the newly inserted Web service operation action link through RMW Web user	
interface	55

Preface

Oracle's Agile PLM documentation set includes Adobe® Acrobat PDF files. The [Oracle Technology Network \(OTN\) Web site](http://www.oracle.com/technetwork/documentation/agile-085940.html) <http://www.oracle.com/technetwork/documentation/agile-085940.html> contains the latest versions of the Agile PLM PDF files. You can view or download these manuals from the Web site, or you can ask your Agile administrator if there is an Agile PLM Documentation folder available on your network from which you can access the Agile PLM documentation (PDF) files.

Note To read the PDF files, you must use the free Adobe Acrobat Reader version 9.0 or later. This program can be downloaded from the [Adobe Web site](http://www.adobe.com) <http://www.adobe.com>.

The [Oracle Technology Network \(OTN\) Web site](http://www.oracle.com/technetwork/documentation/agile-085940.html) <http://www.oracle.com/technetwork/documentation/agile-085940.html> can be accessed through **Help > Manuals** in both Agile Web Client and Agile Java Client. If you need additional assistance or information, please contact My Oracle Support (<https://support.oracle.com>) for assistance.

Note Before calling Oracle Support about a problem with an Agile PLM manual, please have the full part number, which is located on the title page.

TTY Access to Oracle Support Services

Oracle provides dedicated Text Telephone (TTY) access to Oracle Support Services within the United States of America 24 hours a day, 7 days a week. For TTY support, call 800.446.2398. Outside the United States, call +1.407.458.2479.

Readme

Any last-minute information about Agile PLM can be found in the Readme file on the [Oracle Technology Network \(OTN\) Web site](http://www.oracle.com/technetwork/documentation/agile-085940.html) <http://www.oracle.com/technetwork/documentation/agile-085940.html>

Agile Training Aids

Go to the [Oracle University Web page](http://www.oracle.com/education/chooser/selectcountry_new.html) http://www.oracle.com/education/chooser/selectcountry_new.html for more information on Agile Training offerings.

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

About Export/Import Tool

This chapter includes the following:

- Overview 1
- Assumptions and Constraints 2

This chapter provides overview information about the Agile Recipe & Material Workspace (RMW) Export/Import tool.

Overview

Export and import operations are accomplished in two ways:

- From the Recipe & Material Workspace user interface (UI), or
- Using Windows or Unix command-line interface (CLI); working in a CLI, the Export/Import utility is called "ExIm".

Using the RMW UI or ExIm, you can perform the following operations:

- Export information from an RMW development, test, or staging environment and import information to an RMW production environment from the UI.
- Export or import information from the RMW application independent of the installation platform: the export / import procedures can be executed from a client system on any supported platform, using ExIm.

For information about supported platforms for Agile PLM, see the *Agile PLM Capacity Planning Guide*.

- Import information in **insert**, **update** or **delete** mode using ExIm.

The export or import operation is a console-based operation. The export operation connects to the RMW application and makes a request for the data to be exported. The application server retrieves the requested data from the application's database.

The export operation creates an XML file (cfmXML) with exported information at a location that you can specify.

The information that is exported or imported includes metadata and related data present in the application for the exported business object.

The import operation uses the exported information and updates the RMW application.

The output of an export operation is an XML file that contains the information you exported. The import operation uses this XML file to perform the user-specified import operations like insert, update, or delete.

RMW Administrator Objects

In addition to RMW business objects, you can export RMW Administrator objects:

- Users
- Sites
- Views
- Roles
- Alerts
- Workflows
- ... and other Admin objects.

Assumptions and Constraints

Here are some assumptions and constraints about the ExIm tool.

- If primary key (PK) is capable of automatic sequencing (autosequence), the autosequence in the target database for that PK is disabled by the import program and is enabled after the import operation is completed. The disabling or enabling of PKs are done in memory and no changes are made in the database.
- When migrating data that has autosequence attributes, the sequence number for the exported attribute is extracted from the source database and updated in the target database, as long as the autosequence value in the target database is less than that value in the source database.

For example, if the autosequence value for a record is 10 and this record is exported to a database in which the highest autosequence value is 5, the autosequence value in the destination database is reset to 10.

- If a field is autosequence and is "non-PK", no validations are done against this field in the target database, and the value will be inserted "as is".
- The autosequence value generation patterns in the source and target database have to be the same for the PK autosequence attributes. If they are not the same, the import will be aborted.

For example, in the source database, if the pattern values for generating the autosequence is `EQP_<000>` where `<000>` represents a three-digit number, the target database should have the same pattern: `EQP_<000>`.

- A user can run more than one instance of the import program. If same autosequence attribute is being inserted by multiple instances of the import program, only the first import will succeed and others will be aborted.
- Importing data is not possible when the target object definition does not match the source object definition, for example, if the target category definition has a different number of columns than the source category definition.

Installation and Configuration

This chapter includes the following:

- Directory Structure 3
- Configuring Export/Import Tool 3

This chapter provides information on the directory structure of component files in the Export/Import tool, and installation and configuration processes.

Directory Structure

Recipe & Material Workspace must be installed in order to access the files that are required to perform export/import operations. The files required to run export/import are located in the `exim` folder within the RMW installation folder. The `exim` utility is run from the same server on which RMW is installed.

The components of the `exim` folder are described in the table below.

Folder Name	Description
bin	Contains the <code>.bat</code> and <code>.sh</code> files for running export and import procedures in Windows and Solaris, respectively.
config	Contains the configuration files required for ExIm client to work.
config/locale	Contains the locale files for ExIm client messages.
lib	Contains the Java library (<code>.jar</code>) files used by ExIm.
templates	Contains the template files for the export operation.
templates/criteria	Contains the criteria files for the export operation.
templates/schema	Contains the schema files for exporting records using specific criteria.
templates/criteriaProperties	Contains the properties files for the export operation.

Configuring Export/Import Tool

Before exporting or importing information from/to RMW, you must update the following:

- Application server information in the following files:
 - `CFMConfig.xml` file
 - Export or import command files
- Search criteria values in the template files.

Updating the CFMConfig.xml Configuration File

Update the CFMConfig.xml file to include application server information. You must update this file each time you connect to a different server.

To update the CFMConfig.xml file:

1. Open the CFMConfig.xml file available at /migration/config.
2. Specify the source server value if you are exporting information, or specify the target server value if you are importing information:

```
<CFMProviderURL>t3://<server>:<port number></CFMProviderURL>
```

For example,

```
<CFMAppServerInfo>
  <CFMInitialContextFactory>
    Weblogic.jndi.WLInitialContextFactory
  </CFMInitialContextFactory>
<CFMProviderURL>t3://localhost:7001</CFMProviderURL>
<CFMPrincipalUser>Weblogic</CFMPrincipalUser>
  <CFMPrincipalUserPwd>
    VvFM5XvM5jLDvuvz0JROHxcP5WmpoC
  </CFMPrincipalUserPwd>
</CFMAppServerInfo>
```

CFMAppServerInfo Details

Note Do not change the CFMInitialContextFactory tag value.

3. Update values for the following tags only if login information for the WebLogic server has changed from the default value 'Weblogic':

```
<CFMPrincipalUser>
<CFMPrincipalUserPwd>
```

Note If you are updating the Weblogic login information, then the WebLogic administrator password must be encrypted, refer Password Encryption.

4. Save and close the CFMConfig.xml file.

Updating the Export or Import Command Files

To update the WebLogic application server information in the export or import command file:

1. From the AgilePharma/exim/bin/ folder, select the export or import command file you want to update and complete the following:
2. For Windows, the files to be updated are export.bat or import.bat:
 - a. Set BEA=D:\bea\Weblogic103\server\lib
(Change the path depending upon the location of the Weblogic.jar file on your machine)
 - b. Set JAVA_HOME=D:\bea\jdk160_14_R27.6.5-32
(Change the path depending upon location of this folder on your machine)

Note Exlm tool requires the `Weblogic.jar` file to be present in the `home/bea/Weblogic103/server/lib` folder to complete its operation. Since Exlm is Java-based, it requires the path of Java home.

3. For UNIX, the files to be updated are `export.sh` or `import.sh`:
- Set `BEA=home/bea/Weblogic103/server/lib`
(Change the path depending upon location of the `Weblogic.jar` file on your machine)
 - Set `JAVA_HOME=home/bea/jdk160_14_R27.6.5-32`
(Change the path depending upon location of this folder on your machine)
-

Note Exlm tool requires the `Weblogic.jar` file to be present in the `home/bea/Weblogic103/server/lib` folder to complete its operation. Since Exlm is Java-based, it requires the path of Java home.

4. Save and close the export or import command file.
-

Note Appropriate changes must be made to run Exlm client on OAS application server.

Schema, Criteria, and Criteria Properties Files

This chapter includes the following:

▪ Schema Files	7
▪ Updating Template Files	11
▪ Criteria Files	12
▪ Criteria Properties Files	14

The main components used by the ExIm tool are template files:

- Schema template file
- Criteria template file
- Criteria Properties template file

In the Export/Import operation, every business object must be associated to a Schema template file, a Criteria template file, and a Criteria Properties template file. The Criteria Properties template file must be modified before the export operation.

Schema Files

For the Export/Import tool, every business object must have an associated schema file. A schema file defines the structure of a business object. It is an XML representation of the business object's data model. A data model consists of tables and relationships that are part of a business object.

Each schema file contains one primary category and all other categories are directly or indirectly related to this category. The export program processes the schema file and determines the tables from which data needs to be exported for a particular business object.

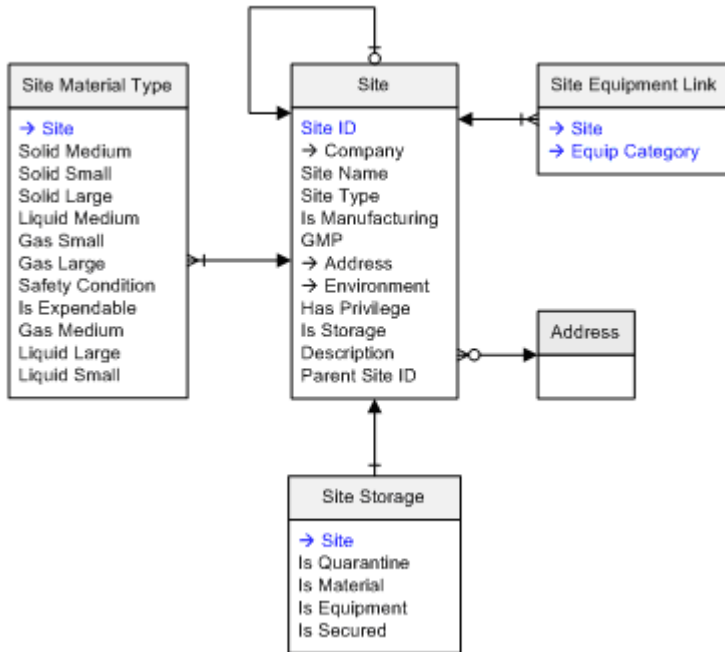
When to Change Values in the Schema File

You may need to change values in the schema file in the following instances:

- When a new category should be included as part of the business object to be exported.
- An existing relationship has changed for a business object.

Example Schema File

This is an example of a schema file for exporting the Site business object. The Entity Relationship model for the Site business object is shown.



The schema file to export data from the tables for a Site business object contains the following information:

```

<?xml version="1.0" encoding="UTF-8"?>
<!--
  Definition file for Site Business Object, including all the related
  objects
-->
<cfmXML xsi:noNamespaceSchemaLocation="../../../cfmxml.xsd"
  xmlns:xsi="http://www.w3.org/2010/XMLSchema-instance">
  <SearchInfo>
    <SearchCriteria id="1">
      <CategoryName/>
      <CategoryDBName>D_PPL_ADDRESS</CategoryDBName>
      <!-- Get 'Address' from D_ORG_SITE using searchCriteria 1 -->
      <Relationship name="Address" type="inbound" refid="2">
        <CategoryName/>
        <CategoryDBName/>
        <ObjectKey/>
      </Relationship>
    </SearchCriteria>
  </SearchInfo>

```

```

</SearchCriteria>
<SearchCriteria id="2" isPrimaryCategory="yes">
  <CategoryName/>
  <CategoryDBName>D_ORG_SITE</CategoryDBName>
</SearchCriteria>
<SearchCriteria id="3">
  <CategoryName/>
  <CategoryDBName>D_ORG_SITE_STORAGE</CategoryDBName>
  <Relationship name="Site" refid="2">
    <CategoryName/>
    <CategoryDBName/>
    <ObjectKey/>
  </Relationship>
</SearchCriteria>
<SearchCriteria id="4">
  <CategoryName/>
  <CategoryDBName>D_ORG_SITE_EQUIPMENT_LK</CategoryDBName>
  <Relationship name="Site" refid="2">
    <CategoryName/>
    <CategoryDBName/>
    <ObjectKey/>
  </Relationship>
</SearchCriteria>
<SearchCriteria id="5">
  <CategoryName/>
  <CategoryDBName>D_ORG_SITE_MATERIAL_TYPE</CategoryDBName>
  <Relationship name="Site" refid="2">
    <CategoryName/>
    <CategoryDBName/>
    <ObjectKey/>
  </Relationship>
</SearchCriteria>
</SearchInfo>
</cfmXML>

```

Schema File Details

The following XML indicates that the schema file belongs to cfmXML schema.

```

<?xml version="1.0" encoding="UTF-8"?>
<!--
  Definition file for Site Business Object, including all the related
  objects
-->
<cfmXML xsi:noNamespaceSchemaLocation="../cfmxml.xsd"
  xmlns:xsi="http://www.w3.org/2010/XMLSchema-instance">

```

Details of XML elements present in the Schema file are listed in the table.

XML Element	Synopsis	Description	Example
Search Info	<code><SearchInfo></code>	A SearchInfo tag is defined for the business object that needs to be exported. SearchInfo can contain one or more SearchCriteria tags. (Details of SearchCriteria is described separately in further sections.)	Based on the Site ER model, Site has an outbound relationship with address. So in the XML you see address category having <code>refid=1</code> , which is the ID of the site category. <code><SearchCriteria id="1"></code>
Category Name	<code><CategoryName></code>	This tag does not contain any value but required for the cfmXML schema.	<code><CategoryName /></code>
Category DB Name	<code><CategoryDBName></code>	This tag contains the database table name of the category.	<code><CategoryDBName>D_PP L_ADDRESS</CategoryDBName></code>
Relationship	<code><Relationship></code>	The <code><Relationship></code> tag specifies the relationship between categories, this information is specified in the reference ID as an attribute. The type attribute specifies if the relationship is of type inbound or outbound. An inbound relationship is an incoming relationship and an outbound relationship refers to an outgoing relationship. Each category should have only one level of relationship information which is not nested. All the tags under Relationship should be left blank as they are required fields in the cfmXML schema.	<code><Relationship name="Address" refid="2" type="inbound"></code> <code><CategoryName></code> <code></CategoryName></code> <code><CategoryDBName></code> <code></CategoryDBName></code> <code><ObjectKey></ObjectKey></code> <code></Relationship></code>
Search Criteria	<code><SearchCriteria></code>	Every table or category in a schema file has search criteria. Search criteria can have one or more relationships. Every category in the schema file should have a direct or an indirect relationship with a primary category. Every search criteria should have a unique ID. The relationship can be inbound or outbound.	The schema file could have one more categories including one primary category and the primary category should be marked as <code>isPrimaryCategory="yes"</code> . <code><SearchCriteria id="2" isPrimaryCategory="yes"></code> <code><CategoryName/></code> <code><CategoryDBName></code> <code>D_ORG_SITE</code> <code></CategoryDBName></code> <code></SearchCriteria></code>

XML Element	Synopsis	Description	Example
Additional Info	<AdditionalInfo>	The schema file can have Additional information. The Additional information tag has the Refresh Cache attribute which can be set to yes or no. By setting it to yes or no, you are specifying if the cache has to be refreshed or not after importing. You must set this value to yes or no only in the exported file.	<pre><AdditionalInfo> <Attribute name= "refreshCache"> <Value dataType= "boolean">no</Value> </Attribute> </AdditionalInfo></pre>

Updating Template Files

For exporting specific information, you must specify search criteria values in the search Criteria Properties template file. This update is required to restrict the export to selected data, and to exclude other records associated with the business object.

Template files are available at the location `/exim/templates`.

The templates folder contains three subordinate folders called Schema, Criteria, and Criteria Properties.

To provide search criteria values for the export operation:

1. Open the criteria properties file that is related to the data you intend to export; the criteria properties files are available at the following location:

`exim/templates/criteriaProperties`

2. Enter the search criteria value.

Note For the search criteria value that you enter, do not provide any space between "=" and its value.
For example:
`$$_COMPANY_NAME=MyOrg`
Assignment operator and its value should not be separated by spaces.

3. Save and close the Criteria Properties file.

Optional: Updating the Schema Template File for Cache Refresh

For some imported information to correctly display in RMW, you must do a cache refresh after you perform the update, which is an import option. To refresh cache, update the output file that is created after a successful export operation.

Some business objects that require cache refresh are roles, category, views, subviews, and so on.

To refresh the cache:

1. Edit the output file.
2. In the output file, set the Attribute value to "yes" as shown in the following xml:

```
...
- <AdditionalInfo>
  - <Attribute name="refreshCache">
    <Value dataType="boolean">yes</Value>
  </Attribute>
</AdditionalInfo>
...
```

3. Save and close the output file.

Criteria Files

A Criteria file is a collection of search criteria for retrieving a subset of information as part of the export operation. It contains the search criteria to be used when extracting business objects from the database. This file is a template for performing search.

Example Criteria File

A sample criteria file for exporting the Site business object is shown here:

```
<? xml version="1.0" encoding="UTF-8"?>
<cfmXML xsi:noNamespaceSchemaLocation="../cfmxml.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SearchInfo>
    <SearchCriteria id="2">
      <CategoryName />
      <CategoryDBName>D_ORG_SITE</CategoryDBName>
      <ViewName></ViewName>
      <Attribute name="Site ID">
        <Value dataType="String">$$_SITE_ID</Value>
        <UnitOfMeasure></UnitOfMeasure>
      </Attribute>
      <Attribute name="Company">
        <Value dataType="String">$$_COMPANY_NAME</Value>
        <UnitOfMeasure></UnitOfMeasure>
      </Attribute>
      <Attribute name="Parent Site ID">
        <Value dataType="String">$$_PARENT_SITE_ID</Value>
        <UnitOfMeasure></UnitOfMeasure>
      </Attribute>
      <Attribute name="Name">
        <Value dataType="String">$$_SITE_NAME</Value>
        <UnitOfMeasure></UnitOfMeasure>
      </Attribute>
    </SearchCriteria>
  </SearchInfo>
</cfmXML>
```

```

    <Attribute name="Site Type">
      <Value dataType="String">$$_SITE_TYPE</Value>
      <UnitOfMeasure></UnitOfMeasure>
    </Attribute>
    <Attribute name="Is Storage Area">
      <Value dataType="boolean">$$_IS_STORAGE_AREA</Value>
      <UnitOfMeasure></UnitOfMeasure>
    </Attribute>
    <Attribute name="Is Manufacturing Area">
      <Value dataType="boolean">$$_IS_MANUFACTURING</Value>
      <UnitOfMeasure></UnitOfMeasure>
    </Attribute>
    <Attribute name="Has Privilege">
      <Value dataType="boolean">$$_HAS_PRIVILEGE</Value>
      <UnitOfMeasure></UnitOfMeasure>
    </Attribute>
    <Attribute name="GMP">
      <Value dataType="boolean">$$_GMP</Value>
      <UnitOfMeasure></UnitOfMeasure>
    </Attribute>
    <Relationship name="Environmental Condition">
      <CategoryName/>
      <CategoryDBName>D_ORG_ENVIRONMENT</CategoryDBName>
      <ObjectKey>
        <Attribute name="Name">
          <Value dataType="String">$$_ENV_COND</Value>
        </Attribute>
      </ObjectKey>
    </Relationship>
  </SearchCriteria>
</SearchInfo>
<AdditionalInfo>
  <Attribute name="schema">
    <Value dataType="file">../schema/site_schema</Value>
  </Attribute>
  <Attribute name="values">
    <Value
dataType="file">../criteriaProperties/site_criteria</Value>
  </Attribute>
</AdditionalInfo>
</cfmXML>

```

Criteria File Details

Details of XML elements present in the Criteria file are listed in the table.

XML Element	Synopsis	Description	Example
Search Info	<code><SearchCriteria></code>	<p>For every search criteria in the schema file you can have a search criteria in the criteria file with all the attributes for that category. Search criteria can have a view name, which is optional.</p> <p>The criteria file corresponds to a view in R&MW but does not contain any search values. Instead of search values, it contains a place holder starting with \$\$_ substring. If there is no \$\$, then the substring is treated as a value and not as a placeholder.</p> <p>Syntax: placeholder=value (no space after the "=" symbol)</p> <p>A search criteria has an ID. It should match with the ID in the schema file.</p>	<pre> <SearchCriteria id="2"> ... <Attribute name="Site ID"> <Value dataType= "String">\$_SITE_ID </Value> <UnitOfMeasure> </UnitOfMeasure> </Attribute> ... </SearchCriteria> </pre>
Additional Info	<code><AdditionalInfo></code>	<p>Criteria file can contain an Additional Info section to provide schema and/or values file details for exporting records. The <AdditionalInfo> tag is used to specify the location and name of the schema and criteria properties file. So you need to specify only the criteria file name at the command line.</p> <p>The file names in this tag should not have an extension as the program automatically adds the extension. Also, the file names are optional.</p>	<pre> <AdditionalInfo> <Attribute name="schema"> <Value dataType= "file">../schema/Site_ schema</Value> </Attribute> <Attribute name="values"> <Value dataType= "file">../criteria Properties/Site_crit eria </Value> </Attribute> </pre>

Criteria Properties Files

The Criteria Properties file is used to specify search values to be used for extracting a subset of data as part of an export operation. Before exporting, the search values in this file must be set. The export operation is based on the values you set in this file.

This file contains values for parameters specified in the criteria file. A parameter may or may not have a value. If it does not have a value in this file then it is not used for searching. Values for the placeholders in the criteria file are given in the criteria properties file.

Example Criteria Properties File

A sample criteria properties file is shown here:

Note # is used to comment information in the criteria properties file.

```
#
# Export all Site using site id
# Values = valid site id
# eg. $$_SITE_ID=LD005
$_SITE_ID=

# Export all Site belong to a company
# Values = valid company name
# eg. $$_COMPANY_NAME=Dragerwerk
$_COMPANY_NAME=Dragerwerk

# Export all Site which belong to one parent site
# Values = valid site name
# eg. $$_PARENT_SITE_ID=Global Site
$_PARENT_SITE_ID=

# Export all Site with given name
# Values = valid site name
# eg. $$_SITE_NAME=GMP
$_SITE_NAME=

# Export site for the give site type
# Values = valid site type
# eg. $$_SITE_TYPE=Storage
$_SITE_TYPE=

# Export All non-storage/storage sites
# Values = boolean [yes/no]
# eg. $$_IS_STORAGE_AREA=yes
$_IS_STORAGE_AREA=

# Export All non-manufacturing/manufacturing sites
# Values = boolean [yes/no]
# eg. $$_IS_MANUFACTURING=yes
$_IS_MANUFACTURING=
```

```
# Export All privileged and non-privileged
# Values = boolean [yes/no]
# eg. $$_HAS_PRIVILEGE=yes
$_HAS_PRIVILEGE=

# Export All GMP/non-GMP sites
# Values = boolean [yes/no]
# eg. $$_GMP=yes
$_GMP=

# Export All sites with different environment conditions
# Values = environment types
# eg. $$_ENV_COND=Refrigerated
$_ENV_COND=
```

Criteria Properties File Details

The criteria properties file contains the following information to enable modification and edits:

```
#
# Variable values for exporting site definition
# One or more of these variables can be used to export Sites(s)
# You can also use operator along with these values
# possible operators are
# > Greater than
# < Less than
# & and
# | Or
# *,? wild cards
# User @ symbol for NULL values, eg $_CREATED_DATE=@
#
# All the above wild card can be used with any of the Search attributes
other than date
# like, created date, modified date etc. Date need to be treated
special
# All date values should be inside '(' and ')' brackets
# eg. $_CREATED_DATE=(01/12/2010)
# you can also give the date format you have used, date format is given
inside '{' '}' braces
# eg. $_CREATED_DATE=(01/12/2010 {dd/MM/yyyy})
# Valid operations in dates are.
# BETWEEN (first date {format}),(second date {format})
# get all records between two date
# eg $_MODIFIED_DATE=BETWEEN (01/01/2010 {MM/dd/yyyy}),(02/08/2010
{MM/dd/yyyy})
#
# Greater than/less than/not equal to ' >/</!= (date {format})'
```

```

# get all records between two date
# eg $$_MODIFIED_DATE= > (01/01/2010 {MM/dd/yyyy})
#
# Possible date formats
#   Date and Time Pattern                                Result
#-----
# yyyy.MM.dd G 'at' HH:mm:ss z                          2010.07.04 AD at 12:08:56 PDT
#   EEE, MMM d, ''yy                                     Wed, Jul 4, '10
#   h:mm a                                                12:08 PM
#   hh 'o'clock' a, zzzz                                  12 o'clock PM, Pacific
#   Daylight Time
#   K:mm a, z                                              0:08 PM, PDT
#   yyyy.MM.dd GGG hh:mm aaa                             02010.July.04 AD 12:08 PM
#   EEE, d MMM yyyy HH:mm:ss Z                           Wed, 4 Jul 2010 12:08:56 -0700
#   yyMMddHHmmssZ                                         100704120856-0700
#   MM/dd/yyyy                                             04/30/2010
#   MM-dd-yyyy                                             04-30-2010
#-----
#

```


Exporting and Importing Data from the Command Line

This chapter includes the following:

▪ Export/Import Parameters.....	19
▪ How to Export	21
▪ How to Import	22

This chapter provides information on commands and parameters required to run the export or import operation.

Export/Import Parameters

Export Parameters

Option	Description
-c	Path of the file containing the export criteria in XML format. If the file path contains only file name, the file is read from the current working directory. Otherwise, the file will be loaded from the specified location.
-u	Login details for user to connect to RMW in <userid>/<Password> format. Password should not contain a "slash" (/).
-d	RMW database name from which the data is being exported. This is the logical name of the database being accessed. The database name is found in CFMDBConfig.xml file or the login page of RMW. The database name is usually 'Production' or 'Development'.
-o	Output filename, which will be created by the export program containing exported data. User can also give a file pattern while exporting. The filename will be generated using the primary category data. For example: -o F:\data\migration\{Name}.xml if used while exporting subview or view, the {Name} part in the output file name will be replaced with the view/subview name. User can give one or more attributes from primary category table. If an attribute name has blank character, the blank/space character should be replaced with '#'. For example: F:\data\migration\{DB#Name}.xml
-pc	Used to override the primary category table with a different child category table when exporting. User can specify the table name with which the primary category should be replaced. It can accept comma separated table names to export from multiple categories.

Option	Description
	Distinct XML files will be generated for each of the categories and the file name would have the category table name prefixed with the output file name specified in the <code>-o</code> option
<code>-h</code>	<p>Used to export record from the entire hierarchy tree.</p> <p>User has to give the parent node table name from where he/she wants to export the sub tree hierarchy data including the parent node.</p> <p>Distinct XML files will be generated for each of the categories in the hierarchy and the file name would have the category table name prefixed with the output file name specified in the <code>-o</code> option.</p>

Import Parameters

Option	Description
<code>-u</code>	<p>Login details for user to access R&MW in <code><userid>/<Password></code> format.</p> <p>Password should not contain a "slash" (/).</p>
<code>-d</code>	<p>RMW database name from which the data is being imported.</p> <p>This is the logical name of the database in the R&MW Application being accessed. The database name is found in <code>CFMDBConfig.xml</code> file or RMW login page. The database name is usually 'Production' or 'Development'.</p>
<code>-i</code>	Used to specify the input data file name including full path, this information will be used by the import program to import data into the target database.
<code>-op</code>	<p>Used to specify the types of transactions that can be performed on the objects present in the database.</p> <p>The transactions are one of the following: <code>UPD</code>, <code>INS</code>, or <code>DEL</code>.</p>
<code>UPD</code>	<p>Read the import data file specified in the <code>-i</code> parameter.</p> <p>For every object in the file:</p> <ul style="list-style-type: none"> ▫ Search the database for object matching the same criteria (same primary keys) ▫ If object found, update the database object with values from the object in the import file. ▫ If no object matching the criteria found, log an error
<code>INS</code>	<p>Read the import data file specified in the <code>-i</code> parameter.</p> <p>For every object in the file, insert a new database object with values from the object in the import file.</p>
<code>DEL</code>	<p>Read the import data file specified in the <code>-i</code> parameter.</p> <p>For every object in the file:</p> <ul style="list-style-type: none"> ▫ Search the database for object matching the same criteria (same primary keys) ▫ If object found, delete the database object ▫ If no object matching the criteria found, log an error

How to Export

To begin exporting information from RMW:

1. Configure the Application Server information in the `CFMConfig.xml` file.
For more information, see [Updating the CFMConfig.xml Configuration File](#).
2. Configure the application server information in the export command file.
For more information, see [Updating the Export or Import Command Files](#).
3. Update the criteria properties template file to export specific information.
For more information, see [Updating Template Files](#).
4. From the respective console, start the export operation:
 - For Windows, from the command prompt, execute the following commands:


```
cd AgilePharma\exim\bin
export.bat
```

 where `AgilePharma\exim` is the location where the export/import tool is installed.
 - For UNIX, from the shell prompt, execute the following commands:


```
.cd AgilePharma/exim/bin
./export.sh
```

 where `AgilePharma\exim` is the location where the export/import tool is installed.

To complete the export process:

1. The AppServer URL information is displayed.
 - For Windows, from the command prompt, execute the export command:


```
export -u <user name/password> -d <database name> -c
<criteria file name> -o <output file name>
```

 If the record fails to export, the event is recorded in the log file that is created by the export program.
The `output.xml` and `.log` files will be created. The log files contain a record of the export operation.
 - For UNIX, from the command prompt, execute the export command:


```
./export.sh -u <user name/password> -d <database name> -c
<criteria file name> -o <output file name>
```

 If the record fails to export, the event is recorded in the log file that is created by the export program.
The `output.xml` and `.log` files will be created. The log files contain a record of the activities during export operation.
The XML file contains object tags. An object represents a record (a row in a table).

Log File

The Export tool creates a log file containing all the informational and error messages, and number of records exported from each category. The log file name is the same as the output file name with `.log` suffix.

For example: If the output file name is `my_data_xport.xml`, the log file name will be `my_data_export.xml.log`.

The export log should contain the following information:

- The login username of the user who performed the export operation
- Execution date and time of the export
- Schema, Criteria, and Properties file name used for the export operation
- A summary for each business object exported, as shown here:

Primary Category: `D_ORG_SITE`

Primary Category Internal id: `LDD424`

Processed Record count by category: `{D_ORG_SITE_MATERIAL_TYPE=0, D_PPL_ADDRESS=1, D_ORG_SITE_STORAGE=1, D_ORG_SITE_EQUIPMENT_LK=0}`

Total processed count: 3

How to Import

To import information into RMW:

1. Configure the Application Server information in `CFMConfig.xml` file.
For more information, see [Updating the CFMConfig.xml Configuration File](#).
2. Configure the Application Server information in the import Command File.
For more information, see [Updating the Export or Import Command Files](#).
3. **Optional:** Update the schema template file for cache refresh.
For more information, see [Updating the Schema template file for Refreshing Cache](#)
4. From the respective console, start the import tool:
 - For Windows, from the command prompt, execute the following commands:

```
cd AgilePharma\exim\bin
import.bat
```


where `AgilePharma\exim` is the location where the export/import tool is installed.
 - For UNIX, from the shell prompt, execute the following commands:

```
.cd AgilePharma/exim/bin
./import.sh
```


where `AgilePharma/exim` is the location where the export/import tool is installed.

5. From the command prompt, execute the import command.

- For Windows:

```
import -u <user name/password> -d <database name> -i <input file path> -op <INS|UPD|DEL>
```

- For UNIX:

```
import -u <user name/password> -d <database name> -i <input file path> -op <INS|UPD|DEL>
```

Modes of Import

- **Insert mode:** If the data being imported does not exist in the target database, it is inserted. If the data being imported already exists, an error message is displayed and no changes are made to the target database.
- **Update mode:** If the data being imported does not exist in the target database, it is inserted. If the data being imported already exists, the existing information is updated.
- **Delete mode:** If the data being imported exists in the target database, it is deleted. If the data being imported does not exist in the target database, an error message is displayed and no changes are made to the target database.

If the record fails to import, the event is recorded in the log file that is created by the export program.

Import Failure

If an import operation fails, the `<import file name>.failed` files are generated with the name of the input files. These failed files can be used for repeating the import operation.

Every failed import creates a log file using the session ID on the server. In case of a failure, you can look into this file to understand the details of activities on the server side. This log file is also created for a successful import operation.

Rollback

During the import operation, if some records related to the imported business object failed to import, the complete business object is rolled back and is not committed. This is recorded in the log file.

Log Files

Log File	Description
<code><import xml file name>.failed</code>	Logs BO records that were not imported.
<code><import xml file>.<Operation><number>.log</code>	Logs the number of insert, update, and delete operations performed.
<code><import xml file>.<Operation>.log</code>	Logs client-side processing details.
<code>CFMMigration_<sessionid>.log</code>	Logs server-side processing details.

Exporting Data with the User Interface

This chapter includes the following:

- Supported Operations and Formats 25
- Searching and Selecting the Export Data 25

Supported Operations and Formats

The application supports exporting RMW data using RMW's User Interface (UI) or the command-line interface (CLI) with Exlm. From the UI, you can export RMW data in the following file formats:

- Excel
- HTML
- PDF
- XML
 - *Object*
 - *Object + Related Objects*

Note To export RMW objects, you must use XML's *Object + Related Objects* option. The other four options list the data that appears in the object's Report in the selected format and not the object which is the data in XML format. Exlm only supports exporting *Object + Related Objects* in XML format.

The basic steps for exporting data from the RMW UI:

1. Log in to RMW.
2. Use the Search utility to locate the objects and data.
3. Select the data from your Search result.
4. Bring up the UI's **Export** utility and select the desired file format.
5. Save or open the exported file.

The procedures for exporting with use of Execute Search and Custom Search are detailed below.


Searching and Selecting the Export Data

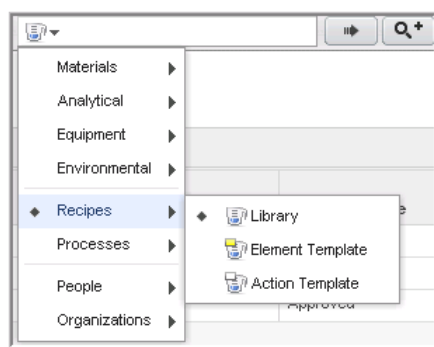
Information provided in this section assume familiarity with the RMW solution and its Execute Search utility, which provides a default report for the selected object, and the Custom Search options. These features refine specifying the criteria for the desired objects.


Exportable Search Results

In addition to exporting search results, RMW supports exporting tables of objects that do not have a related object. For example, the table content on all tables under the **History** tab and under the **Notes and Attachments** tab on all objects can be exported in EXCEL, PDF, HTML, or XML formats.

To Export data using Execute Search:

1. Log on to Agile PLM and Tools menu  select **Recipe & Material Workspace**.
2. In the **Top Navigation** bar, select the Class and subclass to locate the applicable Record. For example, to locate all records in Recipes Library, select **Recipes > Library** followed by an asterisk (*).




3. In the **Top Navigation** bar, click the Execute Search  button. For information on the Custom Search option, see [To export data using Custom Search](#) on page 27.

Records in the Records in "Recipe" is displayed.

		Recipe Name	Description	Lifecycle Phase	Effective Start Date	Effective End Date	Operating Mode	Company ID
		Create Base Chemicals 0		Approved	28-Jul-2010		Clinical Supply	Corporate
		Create Chemicals 0		Approved	27-Jul-2010		Clinical Supply	Corporate
		Water Purification 0		Approved	28-Jul-2010		Clinical Supply	Corporate

4. Select one or more records for export.

The Export button  in the menu bar is activated.

5. Click the Export button and select the export file format from Excel, PDF, HTML, XML Object Only or Object + Related Object.

Depending on the selected file format, a Windows dialog prompts you to open or save the newly created file. Sample Excel and PDF outputs appear below. The reports provide the data that appeared in Execute Search shown in the Records in "Recipe" report above. The only option that exports the object which is an XML file, is XML's *Object + Related Object*.

PDF sample:

ORACLE® Recipe Report								
Version Number	Name	Description	Lifecycle Phase	Effective Start Date	Effective End Date	Operating Mode	Company Name	Recipe Level
1.0	Create Base Chemical		Approved	28-Jul-10		Clinical Supply	Corporate	General

Excel sample:

Recipe Report										
Version Number	Name	Description	Lifecycle Phase	Effective Start Date	Effective End Date	Operating Mode	Company Name	Recipe Level	Recipe Type	Product
1	Create Base Chemical		Approved	28-Jul-10		Clinical Supply	Corporate	General	Processing	
1	Create Chemical 68		Approved	27-Jul-10		Clinical Supply	Corporate	General	Processing	
1	Water Purification		Approved	28-Jul-10		Clinical Supply	Corporate	General	Processing	


Object + Related Object sample:

```

<?xml version="1.0" encoding="UTF-8" ?>
- <!--
    Generated using SDRS Web UI
    Export Object and all its related object
    Business Object ID : Recipe Template
    Category : Recipe
    Instance Ids : [1623163]
    Date : Thu Jul 29 13:10:29 GMT-07:00 2010
-->
- <cfm>XML</cfm>
- <Payload>
  - <ObjectGroup isBO="yes" name="Recipe Template">
    - <Object name="BORoot">
      <CategoryName>Recipe</CategoryName>
      <CategoryDBName>A_RECIPE_TEMPLATE</CategoryDBName>
      <ViewName>DEFAULT</ViewName>
    - <ObjectKey>
      <ObjectID>1623163</ObjectID>
    - <Attribute name="Name">
      <Value dataType="Basic Text">Water Purification</Value>
      <UnitOfMeasure />
    - <Attribute name="Version Number">
      <Value dataType="Basic Text">1.0</Value>
      <UnitOfMeasure />
    - <Attribute>

```

To Export data using Custom Search:

1. Repeat steps 1 and 2 in [To export data using Execute Search](#) on page 26.
2. In the Top Navigation bar, click the Custom Search  button.

Depending on the selected object, fields representing the available criteria for the object are displayed in the *Basic View*. The DEFAULT View displays additional fields for a more comprehensive report.

The screenshot shows the 'Recipes : Recipe Library : Manage' interface. At the top, there is a 'Create New' button and a search icon. Below this is a 'Search Recipe' section with a 'View:' dropdown menu. The dropdown menu is open, showing 'Basic' and 'DEFAULT' options. The search criteria fields include: Name, Description, Lifecycle Phase, Effective Start Date, Effective End Date, Operating Mode, Version Number (with a 'Latest' button), Recipe Level, Purpose, Company : Name, and Product-Product ID. Each field has a dropdown arrow.

- In the selected View, click the drop-down arrows to select the desired value for each criteria and then click **Search**. You can also use * in any field to display the available records for that criteria. For example, if you typed an * in the Name field, the names of records in Recipe Library is displayed.

The screenshot shows the 'Recipes : Recipe Library : Manage' interface with the 'View' set to 'Recipe Quick Search'. The 'Name' field contains an asterisk (*). A dropdown menu is open, showing a list of records: 'Create Base Chemical', 'Create Chemical 69', and 'Water Purification'. Below the search fields, there is a 'Records in "Recipe"' section with a table. The table has columns for 'Edit', 'Delete', 'More', and 'Change'. The table shows 3 records. At the bottom, there is a 'Rows Per Page' dropdown set to 10 and a status bar indicating '0 rows selected of 6'.

- To continue, select the records you want to export.
The Export button  in the menu bar is activated.
- Click the Export button and select the export file format from Excel, PDF, HTML, XML *Object Only* or *Object + Related Object*.

Note When exporting from custom search results table, depending on the view selected for search, different set of columns are displayed. During the export operation, values displayed in these columns are also exported.

Scenario Examples

This chapter includes the following:

▪ Views	29
▪ View Attributes	33
▪ Categories	36
▪ Buttons.....	40
▪ Component Buttons	46
▪ Web Services.....	47
▪ Web Service Operations	51
▪ Action-to-Web-Service Operations	54

This chapter provides additional information about exporting and importing data from/to RMW, with examples. Complete the configuration steps outlined in [Configuring Export Import Tool](#) on page 3 before you use the scenario examples to export or import information.

Views

To insert, update or delete views from the RMW application, use View schema files.

Note When using the View schema file to import views, only view information along with attribute and other details will be imported. Related or parent categories and other references will not be imported. These reference records should be imported separately prior to view import.

Template files:

Criteria file: view_criteria.xml

Schema file: view_schema.xml

Properties file: view_criteria.properties

Using Search Attributes to Export Views

You can use the following search attributes to export a view, as described here.

- **View Name:** Provide the Web service ID for search attribute \$\$_VIEW_NAME. For example, \$\$_VIEW_NAME=Alertsearchview
- **Owner Category Name:** To export all views that belong to this category. Provide the service name for search attribute \$\$_CATEGORY_NAME. For example, \$\$_CATEGORY_NAME=AlertDefinition

Inserting New Views

To insert view(s) to the RMW application, the view(s) should be exported from the source RMW application and imported to the destination RMW application in insert mode.

Step 1: Export existing view from the source

1. Open the view_criteria.properties file present in the migration/templates/criteriaproPERTIES folder.
2. Specify the search attribute values to export the records.
3. Check the preceding section for using different search attributes.
4. Export the records using following command:

```
<migration_home>./export.sh -d Development -u admin/admin -c  
<migration_home>/templates/criteria/view_criteria.xml -o  
<migration_home>/alertsearchview.xml
```

Step 2: Import exported view to the destination RMW application

1. To import the exported xml using the following command:

```
<migration_home>./import.sh -d Development -u admin/admin -i <input  
file location>/alertsearchview.xml -op INS
```

Step 3: Verify the imported view through RMW Web user interface Meta model UI

1. Open the RMW Web URL in the Web browser.
2. Log in to the application using valid user credentials.
3. Navigate to **Administration > Setup > Meta model > View > Manage**.
4. Provide the view name as search criteria in the **View Name** column.
5. Click the **Search** button.
6. Check if the imported view is displayed in the search result.
7. Click on the view name to verify the view details.

Step 4: Check the imported view through RMW Web user interface Library module

1. Open the RMW Web URL in the Web browser.
2. Log in to the application using valid user credentials.
3. Navigate to **Administration> Library > Manage Record**.
4. Click the **Browse** tab.
5. Select category **Base > Meta Model Folder > View**.
6. Click the **Search** tab.
7. Provide the required search criteria, you can use the similar criteria values used for export.
8. Search for the record and verify the imported records shown in the search result window.
9. Click on the view link to verify the details.

Updating Views

To update existing views in the RMW application, the views should be exported from the source RMW application and imported to the destination RMW application in update mode.

Step 1: Export existing view(s) from source RMW application

1. Open the views_criteria.properties file.
2. Specify the search attribute values to export records.
3. Check the preceding section for using different search attributes.
4. Export the records using the following command:

```
<migration_home>./export.sh -d Development -u admin/admin -c
<migration_home>/templates/criteria/view_criteria.xml -o
<migration_home>/alertsearchview.xml
```

Step 2: Import exported view to the destination

1. Import the exported views using the following command:

```
<migration_home>./import.sh -d Development -u admin/admin -i
<migration_home>/alertsearchview.xml -op UPD
```

Step 3: Verify the view import through RMW Web user interface Meta model UI

1. Open the RMW Web URL in the Web browser.
2. Log in to the application using valid user credentials.
3. Navigate to **Administration > Setup > Meta model > View > Manage**.
4. Provide the **view name** as search criteria in the **View Name** column.
5. Click the **Search** button.
6. Check if the imported view is displayed in the search result.
7. Click on the view name to verify the view details.

Step 4: Check the imported view through RMW Web user interface Library module

1. Open the RMW Web URL in the Web browser.
2. Log in to the application using valid user credentials.
3. Navigate to **Administration > Library > Manage Record**.
4. Click the **Browse** tab.
5. Select category **Base > Meta Model > Folder View**.
6. Click the **Search** tab.
7. Provide the required search criteria; you can use the similar criteria values used for export.
8. Search for the record and verify the imported records shown in the search result window.
9. Click on the view link to verify the details.

Deleting Views

To delete existing view(s) in the RMW application, the view(s) should be first exported from the RMW application and then imported to the RMW application in delete mode.

Step 1: Export existing view(s) from source RMW application

1. Open the views_criteria.properties file.
2. Specify the search attribute values to export records.
3. Check the preceding section for using different search attributes.
4. Export the records using following command:

```
<migration_home>./export.sh -d Development -u admin/admin -c  
<migration_home>/templates/criteria/view_criteria.xml -o  
<migration_home>/alertsearchview.xml
```

Step 2: Import the exported views to the destination RMW application

1. Import the exported xml using the following command:

```
<migration_home>./import.sh -d Development -u admin/admin -i  
<migration_home>/alertsearchview.xml -op DEL
```

Step 3: Verify the deleted view import through RMW Web user interface Meta model UI

1. Open the RMW Web URL in the Web browser.
2. Log in to the application using valid user credentials.
3. Navigate to **Administration > Setup > Meta model > View > Manage**.
4. Provide the view name as search criteria in the View Name column.
5. Click the **Search** button.
6. Check if the imported view not is displayed in the search result window.

Step 4: Check the imported view through RMW Web user interface Library module.

1. Open the RMW Web URL in the Web browser.
2. Log in to the application using valid user credentials.
3. Navigate to **Administration > Library > Manage Record**.
4. Click the **Browse** tab.
5. Select category **Base > Meta Model > Folder View**.
6. Click the **Search** tab.
7. Provide the required search criteria; you can use typical criteria values used for export.
8. Search for the record and verify the imported records shown in the search result window.
9. Click on the view link to verify the details.

View Attributes

View attribute template files are used to insert, update or delete view attributes for a view. When using view attribute schema only view attributes details are exported or imported. Referencing view or category attributes are imported separately and available prior to view attribute import.

Template files:

Criteria file: viewattribute_criteria.xml

Schema file: viewattribute_schema.xml

Properties file: viewattribute_criteria.properties

Using Search Attributes to Export Views

You can use the following search attributes to export a view name, as described here.

- **View Name:** Provide the Web service ID for search attribute `$$_VIEW_NAME`. For example, `$$_VIEW_NAME=Alertsearchview`
- **View Attribute Name:** Provide the service name for search attribute `$$_VIEW_ATTR_NAME`. For example, `$$_VIEW_ATTR_NAME=AlertName`

Adding Additional Attributes to Existing Views

To insert view attribute(s) to the RMW application, the view attribute(s) should be exported from the source RMW application and imported to the destination RMW application in insert mode.

Step 1: Export existing view attribute(s) from source RMW application.

1. Open the viewattribute_criteria.properties file.
2. Specify the search attribute values to export records.
3. Check the preceding section for using different search attributes.
4. Export the records using following command:

```
<migration_home>./export.sh -d Development -u admin/admin -c
<migration_home>/templates/criteria/viewattribute_criteria.xml -o
<migration_home>/alertsearchviewattributes.xml
```

Step 2: Import the exported view attributes to destination RMW application.

1. Import the exported xml using the following command:

```
<migration_home>./import.sh -d Development -u admin/admin -i
<migration_home>/alertsearchviewattributes.xml -op INS
```

Step 3: Check the imported view attributes through RMW Web user interface Meta model UI.

1. Open the RMW Web URL in Internet Explorer Web browser.
2. Log in to the application using valid user credentials.

3. Navigate to **Administration > Setup > Meta model > View > Manage**.
4. Provide the view name as search criteria in the View Name column. Click the **Search** button.
5. Check if the imported view is displayed in the search result.
6. Click on the view name to view the search and result attribute used in the view.
7. Click on the respective tab to view the view attributes details.

Step 5: Check the imported view attributes through RMW Web user interface Library module.

1. Open the RMW Web URL in Internet Explorer Web browser.
2. Log in to the application using valid user credentials and Database as **Development**.
3. Navigate to **Administration> Library > Manage Record**.
4. Click the **Browse** tab.
5. Select category **Base > Meta Model Folder > View Attribute**.
6. Click the **Search** tab.
7. Provide the required search criteria, you can use the typical criteria values used for export.
8. Search for the record and verify the imported records shown in the search result window.

Updating Existing View Attributes

To update view attribute(s) to the RMW application, the view attribute(s) should be exported from the source RMW application and imported to the destination RMW application in update mode.

Step 1: Export existing view attribute(s) from source RMW application.

1. Open the viewattribute_criteria.properties file.
2. Specify the search attribute values to export records.
3. Check the preceding section for using different search attributes.
4. Export the records using the following command:

```
<migration_home>./export.sh -d Development -u admin/admin -c  
<migration_home>/templates/criteria/viewattribute_criteria.xml -o  
<migration_home>/alertsearchviewattributes.xml
```

Step 2: Import exported view attributes to the destination RMW application

1. Import the exported xml using the following command:

```
<migration_home>./import.sh -d Development -u admin/admin -i  
<migration_home>/alertsearchviewattributes.xml -op UPD
```

Step 3: Verify the imported view attributes through RMW Web user interface UI

1. Open the RMW Web URL in Internet Explorer Web browser.
2. Log in to the application using valid user credentials and Database as **Development**.
3. Navigate to **Administration > Setup > Meta model > View > Manage**.
4. Provide the view name as search criteria in the **View Name** column.

5. Click the **Search** button.
6. Check if the imported view is displayed in the search result.
7. Click on the view name to view the view details.
8. Click on the respective tab to view the view attributes details.

Step 6: Check the imported view attributes through RMW Web user interface Library module.

1. Open the RMW Web URL in Internet Explorer Web browser.
2. Log in to the application using valid user credentials and Database as **Development**.
3. Navigate to **Administration > Library > Manage Record**.
4. Click the **Browse** tab.
5. Select category **Base > Meta Model Folder > View Attribute**.
6. Click the **Search** tab.
7. Provide the required search criteria, you can use typical criteria values used for export.
8. Search for the record and verify the imported records shown in the search result window.

Deleting Existing View Attributes From Views

To delete existing view attribute(s) to RMW application, the view attribute(s) should be imported to destination the RMW application in delete mode.

Step 1: Export existing view attribute(s) from source RMW application

1. Open the viewattribute_criteria.properties file.
2. Specify the search attribute values to export records.
3. Check the preceding section for using different search attributes.
4. Export the records using following command:

```
<migration_home>./export.sh -d Development -u admin/admin -c
<migration_home>/templates/criteria/viewattribute_criteria.xml -o
<migration_home>/alertsearchviewattributes.xml
```

Step 2: Import the exported view attributes to destination RMW application.

Import the exported xml using the following command:

```
<migration_home>./import.sh -d Development -u admin/admin -i
<migration_home>/alertsearchviewattributes.xml -op DEL
```

Step 3: Check the imported view attributes through RMW Web user interface Meta model UI.

1. Open the RMW Web URL in Internet Explorer Web browser.
2. Log in to the application using valid user credentials and Database as **Development**.
3. Navigate to **Administration > Setup > Meta model > View > Manage**.
4. Provide the view name as search criteria in the 'View Name' column. Hit the search button.
5. Check if the imported view is displayed in the search result.

6. Click on the view name to view the search and result attribute used in the view.
7. Click on the respective tab to view the view attributes details.
8. Verify the deleted attribute removed from the view.

Step 4: Check the imported view attributes through RMW Web user interface Library module.

1. Open the RMW Web URL in Internet Explorer Web browser.
2. Log in to the application using valid user credentials.
3. Navigate to **Administration > Library > Manage Record**.
4. Click the **Browse** tab.
5. Select category **Base > Meta Model Folder > View Attribute**.
6. Click the **Search** tab.
7. Provide the required search criteria, you can use the similar criteria values used for export.
8. Search for the record and verify the imported records are not shown in the search result window.

Categories

Using the category schema, refer table Business Objects and related templates to find the files related to the category.

Inserting a New Category

You can export an existing category from one RMW application (source) to another RMW application (destination). The example shows how to export the category Alert Definition from the source RMW application to the destination RMW application.

Step 1: Export existing category from the source.

1. Update the CFMConfig.xml to point to the source RMW application.
2. Open the category_criteria.properties file.
3. Specify values for any of the search attributes to export the category.
4. Export category using category name.
Provide the category name using `$$_CAT_NAME=property`. For example,
`$$_CAT_NAME=AlertDefinition`
5. Export the category using the following command:

```
<migration_home>./export.sh -d Development -u admin/admin -c  
<migration_home>/templates/category_criteria.xml <output file  
location>/alertdefinition.xml
```

Step 2: Import exported category to the destination.

1. Update the CFMConfig.xml to point to the source RMW application.
2. Use the following command to import the category to the destination:

```
<migration_home>./import.sh -d Development -u admin/admin -i <input
file location>/alertdefinition.xml -op INS
```

Step 3: Verify the category import through RMW Web user interface.

1. Open the RMW Web URL in the Web browser.
2. Log in to the application using valid user credentials.
3. Navigate to **Administration > Setup > Meta model > Category > Manage**.
4. Provide the category name as search criteria in the **Name** column.
5. Click the **Search** button.
6. Check if the imported category is displayed in the search result.

Updating an Existing Category

The example shows how to export the category 'Alert Definition' from the source RMW application, after making some modification to the existing category and importing the same category to the destination RMW application.

Step 1: Export existing category from the source.

Refer the preceding section on how to export category.

Step 2: Import exported category to the destination.

1. Update the CFMConfig.xml to point to the source RMW application.
2. Use the following command to import the category to the destination.

```
<migration_home>./import.sh -d Development -u admin/admin -i <input
file location>/alertdefinition.xml -op UPD
```

Step 3: Verify the category import through RMW Web user interface.

1. Open the RMW Web URL in the Web browser.
2. Log in to the application using valid user credentials.
3. Navigate to **Administration > Setup > Meta model > Category > Manage**.
4. Provide the category name as search criteria in the **Name** column.
5. Click the **Search** button.
6. Check if the imported category is displayed in the search result.
7. Click on the category name to view the view details.
8. Check the category details for the changes you have made.
9. Refresh the cache if you do not see the changes. Refer the cache refresh section to see how to refresh the cache.

Deleting an Existing Category

This example shows how to delete the category Alert Definition from the destination RMW

application.

Step 1: Export existing category from the source.

Refer the preceding section on how to export a category.

Step 2: Import exported category to the destination.

Use the following command to import the category to the destination:

```
<migration_home>./import.sh -d Development -u admin/admin -i <input  
file location>/alertdefinition.xml -op DEL
```

Step 3: Verify the category import through RMW Web user interface.

1. Open the RMW Web URL in the Web browser.
2. Log in to the application using valid user credentials.
3. Navigate to **Administration > Setup > Meta model > Category > Manage**.
4. Provide the view name as search criteria in the **Name** column.
5. Click the **Search** button. The deleted category will not be displayed in the search result.

Adding Additional Attributes To Existing Categories

Using the Category Attribute schema refer the table Business Objects and related templates to find the files related to Category Attribute.

Exporting category attributes of an existing category from one RMW application to importing these attributes to the same category in another RMW application. The same category should exist in both the application. Assume these additional attributes are not available in the destination RMW application category.

Step 1: Export category attributes.

1. Update the CFMConfig.xml to point to the source RMW application.
2. Open the categoryattribute_criteria.properties file.
3. Specify value for any of the search attribute to export the view attribute.
4. Export category attributes using category name.
Provide the category name using `$$_OWNER_CAT_NAME=property`. For example,
`$$_OWNER_CAT_NAME=AlertDefinition`
5. Export category attribute using attribute label.
Provide category name using `$$_ATTR_LABEL=property`. For example,
`$$_ATTR_LABEL=AlertName`
6. Export the category attributes using the following command:

```
<migration_home>/bin/export.sh -d Development -u admin/admin -c  
<migration_home>/templates/categoryattribute_criteria.xml <output file  
location>/alertdefinition_attributes.xml
```

Step 2: Import exported category attributes to the destination.

1. Update the CFMConfig.xml to point to the source RMW application.
2. Use the following command to import the category attributes to the destination:

```
<migration_home>./import.sh -d Development -u admin/admin -i <input
file location>/alertdefinition_attributes.xml -op INS
```

Step 3: Verify the category attributes import through RMW Web user interface.

1. Open the RMW Web URL in Internet Explorer Web browser.
2. Log in to the application using valid user credentials.
3. Provide the user credentials and select **Development** as the DB name.
4. Navigate to **Administration > Setup > Meta model > Category > Manage**.
5. Provide the category name as search criteria in the **Name** column.
6. Click the **Search** button.
7. Check if the imported category is displayed in the search result.
8. Click on the category name to view the category details.
9. In category details window check the category attributes added.
10. Refresh the cache if you don't see the changes. Refer the cache refresh section to see how to refresh the cache.

Updating Existing Category Attributes

Exporting category attributes of an existing category from one RMW application to importing these attributes to the same category in another RMW application. The same category should exist in both the application. Assume these additional attributes are available in the destination RMW application category. (Update operation will insert the attributes if not found in the destination).

Step 1: Export category attributes.

Refer the preceding section to see how to export category attributes.

Step 2: Import exported category attributes to the destination.

Use the following command to import the category attributes to the destination:

```
<migration_home>./import.sh -d Development -u admin/admin -i <input
file location>/alertdefinition_attribute.xml -op UPD
```

Step 3: Verify the category attribute import through RMW Web user interface.

1. Open the RMW Web URL in Internet Explorer Web browser.
2. Log in to the application using valid user credentials.
3. Provide the user credentials and select **Development** as the DB name.
4. Navigate to **Administration > Setup > Meta model > Category > Manage**.
5. Provide the category name as search criteria in the **Name** column.

6. Click the **Search** button.
7. Check if the category is displayed in the search result.
8. Click on the category name to view the category details.
9. In category details window check the attributes updated.
10. Refresh the cache if you do not see the changes. Refer the cache refresh section to see how to refresh the cache.

Deleting Existing Category Attributes

Exporting category attributes of an existing category from one RMW application to importing these attributes to the same category in another RMW application. The same category should exist in both the application. Assume these additional attributes are available in the destination RMW application category.

Step 1: Export category attributes.

Refer the preceding section to see how to export category attributes.

Step 2: Import exported category attributes to the destination.

1. Update the CFMConfig.xml to point to the source RMW application.
2. Use the following command to import the category attributes to the destination:

```
<migration_home>./import.sh -d Development -u admin/admin -i <input  
file location>/alertdefinition_attribute.xml -op DEL
```

Step 3: Verify the category attributes import through RMW Web user interface.

1. Open the RMW Web URL in Internet Explorer Web browser.
2. Log in to the application using valid user credentials.
3. Navigate to **Administration > Setup > Meta model > Category > Manage**.
4. Provide the category name as search criteria in the **Name** column.
5. Click the **Search** button.
6. Check if the category is displayed in the search result.
7. Click on the category name link to view the category details.
8. In the View Details window, check if the category attributes are deleted from the category.
9. Refresh the cache if you do not see the changes. Refer the cache refresh section to see how to refresh the cache.

Buttons

Inserting New Buttons

Using Button Schema refer the table Business Objects and related templates to find the files related to Button.

Exporting existing button from one RMW application (source) to another RMW application (destination). The example shows how to export the button 'Add Roles' from source RMW application to the destination RMW application.

Step 1: Export existing button from the source.

1. Update the CFMConfig.xml to point to the source RMW application.
2. Open the button_criteria.properties file.
3. Specify value for any of the search attribute to export the button.
 1. Export button using button name. Provide the button name using `$$_BUTTON_NAME=property`. For example, `$$_BUTTON_NAME=AddRoles`
 2. Export All Buttons using component ID. Provide the component ID using `$$_COMPONENT_ID=property`. For example, `$$_COMPONENT_ID=PCM_EDIT_CONFIRM_SA_BTNBAR`
4. Export the button using the following command:

```
<migration_home>./export.sh -d Development -u admin/admin -c
<migration_home>/templates/button_criteria.xml <output file
location>/addrolesbutton.xml
```

Step 2: Import exported button to the destination.

1. Update the CFMConfig.xml to point to the source RMW application.
2. Use the following command to import the button to the destination:


```
<migration_home>./import.sh -d Development -u admin/admin -i <input
file location>/addrolesbutton.xml -op INS
```

Step 3: Verify the button import through RMW Web user interface.

1. Open the RMW Web URL in Internet Explorer Web browser.
2. Log in to the application using valid user credentials.
3. Navigate to **Administration > Library > Manage Record**.
4. Click the **Browse** tab.
5. Select **Base > System Folder > Authorization Folder > Button** category.
6. Click the **Search** tab.
7. Provide the button name as search criteria in the **Button Name** column.
8. Click the **Search** button.

9. Check if the imported button is displayed in the search result.

Updating Existing Buttons

Using the Button Schema refer the table Business Objects and template files to know the files related to Button]).

Exporting existing button from one RMW application (source) to another RMW application (destination). The example shows how to export the button 'Add Roles' from source RMW application to the destination RMW application.

Step 1: Export existing button from the source.

1. Update the CFMConfig.xml to point to the source RMW application.
2. Open the button_criteria.properties.
3. Specify value for any of the search attribute to export the button.
 1. Export button using button name. Provide the button name using `$$_BUTTON_NAME=property`. For example, `$$_BUTTON_NAME=AddRoles`
 2. Export All Buttons using component ID. Provide the component ID using `$$_COMPONENT_ID=property`. For example, `$$_COMPONENT_ID=PCM_EDIT_CONFIRM_SA_BTNBAR`
4. Export the button using the following command:

```
<migration_home>./export.sh -d Development -u admin/admin -c
<migration_home>/templates/button_criteria.xml <output file
location>/addrolesbutton.xml
```

Step 2: Import exported button to the destination.

1. Update the CFMConfig.xml to point to the source RMW application.
2. Use the following command to import the button to the destination:

```
<migration home>./import.sh -d Development -u admin/admin -i <input
file location>/addrolesbutton.xml -op UPD
```

Step 3: Verify the button import through RMW Web user interface.

1. Open the RMW Web URL in Internet Explorer Web browser.
2. Log in to the application using valid user credentials.
3. Navigate to **Administration > Library > Manage Record**.
4. Click the **Browse** tab.
5. Select **Base > System Folder > Authorization Folder > Button** category.
6. Click the **Search** tab.
7. Provide the button name as search criteria in the Button Name column.
8. Click the **Search** button.
9. Check if the imported button is displayed in the search result.
10. Click on the button name to view the details.

Deleting Existing Buttons

Using the Button Schema refer the table Business Objects and related templates to know the files related to Button]).

Step 1: Export existing button from the source.

1. Update the CFMConfig.xml to point to the source RMW application.
2. Open the button_criteria.properties.
3. Specify value for any of the search attribute to export the button.
 1. Export button using button name. Provide the button name using `$$_BUTTON_NAME=property`. For example, `$$_BUTTON_NAME=AddRoles`
 2. Export All Buttons using component ID. Provide the component ID using `$$_COMPONENT_ID=property`. For example, `$$_COMPONENT_ID=PCM_EDIT_CONFIRM_SA_BTNBAR`
4. Export the button using the following command:

```
<migration_home>./export.sh -d Development -u admin/admin -c
<migration_home>/templates/button_criteria.xml <output file
location>/addrolesbutton.xml
```

Step 2: Import exported button to the destination.

1. Update the CFMConfig.xml to point to the source RMW application.
2. Use the following command to import the button to the destination:


```
<migration home>./import.sh -d Development -u admin/admin -i <input
file location>/addrolesbutton.xml -op DEL
```

Step 3: Verify the button import through RMW Web user interface.

1. Open the RMW Web URL in Internet Explorer Web browser.
2. Log in to the application using valid user credentials.
3. Navigate to **Administration > Library > Manage Record**.
4. Click the **Browse** tab.
5. Select **Base > System Folder > Authorization Folder > Button** category.
6. Click the **Search** tab.
7. Provide the button name as search criteria in the **Button Name** column.
8. Click the **Search** button.
9. Check if the imported button is not displayed in the search result.

Renaming Existing Buttons

The Button Name is a primary key in the button table. To rename an existing button, you must delete the existing button and insert a new button.

The Button Callback has reference to Button. So before deleting the button the referencing button callbacks have to be updated to point to new button. Following are the steps involved in renaming a button.

The example shows how to rename an existing button 'Add Roles' to 'Add New Roles'. Expectation is, the button is renamed to 'Add New Roles' in the source RMW application and a button named 'Add Roles' exist in the destination RMW application

Step 1: Export existing button from destination RMW application.

1. Open the button_criteria.properties file.
2. Specify a value for any of the search attributes to export the button.
Export the button using the button name. Provide the button name using `$$_BUTTON_NAME=property`. For example, `$$_BUTTON_NAME=AddRoles`
3. Export the button using the following command:

```
<migration_home>./export.sh -d Development -u admin/admin -c  
<migration_home>/templates/button_criteria.xml <output file  
location>/addrolesbutton.xml
```

Step 2: Export renamed button from source RMW application.

1. Open the button_criteria.properties file.
2. Specify a value for any of the search attributes to export the button.
Export button using button name. Provide the button name using `$$_BUTTON_NAME=property`. For example, `$$_BUTTON_NAME=AddNewRoles`
3. Export the button using the following command:

```
<migration_home>./export.sh -d Development -u admin/admin -c  
<migration_home>/templates/button_criteria.xml <output file  
location>/addnewrolesbutton.xml
```

Step 3: Import the exported button to the destination RMW application.

Use the following command to import the button to the destination:

```
<migration_home>./import.sh -d Development -u admin/admin -i <input  
file location>/addnewrolesbutton.xml -op INS
```

Step 4: Verify the newly imported button through RMW Web user interface.

1. Open the RMW Web URL in Internet Explorer Web browser.
2. Log in to the application using valid user credentials.
3. Navigate to **Administration > Library > Manage Record**.
4. Click the **Browse** tab.
5. Select **Base > System Folder > Authorization Folder > Button** category.

6. Click the **Search** tab.
7. Provide the button name as search criteria in the Button Name column.
8. Click the **Search** button.
9. Check if the imported button is displayed in the search result window.

Step 5: Export the related button callbacks (s) from source RMW application.

1. Open the buttoncallback_criteria.properties file.
2. Specify the search attribute values to export the button callbacks.
3. Export all button callbacks for the given button. Provide the value for \$\$_BUTTON_NAME search attribute. For example, \$\$_BUTTON_NAME=AddNewRoles
4. Export the button callbacks using following command:

```
<migration_home>./export.sh -d Development -u admin/admin -c
<migration_home>/templates/criteria/buttoncallback_criteria.xml -o
<migration_home>/addnewrolesbuttoncallbacks.xml
```

Step 6: Import the exported button callbacks into the destination RMW application in update mode.

Use the following command to import the button to the destination:

```
<migration_home>./import.sh -d Development -u admin/admin -i <input
file location>/addnewrolesbuttoncallbacks.xml -op UPD
```

Step 7: Check the updated button callbacks through RMW Web user interface.

1. Open the RMW Web URL in Internet Explorer Web browser.
2. Log in to the application using valid user credentials.
3. Navigate to **Administration > Library > Manage Record**.
4. Click the **Browse** tab.
5. Select **Base > System Folder > Authorization Folder > Button Callback** category.
6. Click the **Search** tab.
7. Provide the button name as search criteria in the **Button Name** column.
8. Click the **Search** button.
9. Check if the imported button callbacks are displayed in the search result window.

Step 8: Import the old button (exported in Step 1) into destination RMW application in delete mode.

Use the following command to import the button to the destination:

```
<migration home>./import.sh -d Development -u admin/admin -i <input
file location>/addrolesbutton.xml -op DEL
```

Step 9: Verify the deleted button through RMW Web user interface.

1. Open the RMW Web URL in Internet Explorer Web browser.
2. Log in to the application using valid user credentials.

3. Navigate to **Administration > Library > Manage Record**.
4. Click the **Browse** tab.
5. Select **Base > System Folder > Authorization Folder > Button** category.
6. Click the **Search** tab.
7. Provide the button name as search criteria in the Button Name column.
8. Click the **Search** button.
9. Verify that the deleted button is not displayed in the search result window.

Component Buttons

The Component Button template files can be used to link or unlink existing buttons to UI components. Existing links can be exported from one RMW application to another RMW application.

Note This procedure using the Component Button template exports or imports only the link between the component and button. When importing a component button link, the related button and component should be available in the destination RMW application. Components and buttons should be exported and imported separately.

Template files:

Criteria file: component_button_link_criteria.xml

Schema file: component_button_link_schema.xml

Properties file: component_button_link_criteria.properties

Using Search Attributes to Export Button-Component Links

You can use the following search attributes to export a button-component link, as described here.

- **Component ID:** To export all button-component links for the given component ID(s).
Provide the value for the `$$_COMPONENT_ID` search attribute. For example,
`$$_COMPONENT_ID=CMPT_USR_EDIT_ADD_SITE_ROLE_LIST`
- **Screen ID:** To export all button-component links across all components for the given screen ID(s).
Provide the value for the `$$_SCREEN_ID` search attribute. For example,
`$$_SCREEN_ID=PCM_SCR_ALT_ADD_EDIT_12`
- **Callback ID:** to export all button-component links for the given button callback ID.
Provide the value for the `$$_BUTTON_CALLBACK_ID` search attribute. For example,
`$$_BUTTON_CALLBACK_ID=PCM_USR_EDIT_ADD_SITE`
- **Button Name:** To export all button-component links for the given button.
Provide the value for the `$$_BUTTON_NAME` search attribute. For example,
`$$_BUTTON_NAME=AddaSite`

Linking Existing Buttons to a Component

To link existing buttons to a component in the destination RMW application, export the link from the source RMW application and import them to the destination RMW application.

Perform the following steps to export and import component-button links:

Step 1: Export existing component button link(s) from source RMW application.

1. Open the component_button_link_criteria.properties file.
2. Specify the search attribute values to export the button component link. For details, see [Using Search Attributes to Export Button-Component Links](#) 46.
3. Export the links using the following command:

```
<migration_home>./export.sh -d Development -u admin/admin -c
<migration_home>/templates/criteria/component_button_link_criteria.xml
-o <migration_home>/button_complinks.xml
```

Step 2: Import the exported links to the destination RMW application, using the following command:

```
<migration_home>./import.sh -d Development -u admin/admin -i
<migration_home>/button_complinks.xml -op INS
```

Step 3: Check the newly added component button links through the RMW Web user interface.

1. Open the RMW Web URL in the Web browser.
2. Log in to the application using valid user credentials.
3. Navigate to **Administration > Library > Manage Record**.
4. In the **Browse** tab, select category **Base > System Folder > Authorization Folder > Component Button**.
5. Click the **Search** tab.
6. Provide the required search criteria. You can use typical criteria values used for export, such as Button Callback ID, Component ID, and so on.
7. Search for the record and verify that the newly imported button-component links are shown in the search results.
8. Click on the record to verify the details of the imported record.

Web Services

The Web service templates are used to insert, update and delete Web services and related information from one RMW application to another RMW application.

Note When exporting Web service details using Web service templates, only the Web service related information is exported. Related integration application setup data will not be exported, this should be exported and imported separately, and should be available in the destination RMW application before importing Web service.

Template files:

Criteria file: webservices_criteria.xml

Schema file: webservices_schema.xml

Properties file: webservices_criteria.properties

Using Search Attributes to Export Web Services

You can use the following search attributes to export web services, as described here.

- **Web Service ID:** Provide the Web service ID for search attribute \$\$_WEBSERVICE_ID. For example, \$\$_WEBSERVICE_ID=WeighScaleService
- **Service Name:** Provide the service name for search attribute \$\$_SERVICE_NAME. For example, \$\$_SERVICE_NAME=WeighScaleService

Inserting New Web Services

To insert a new Web service to the destination RMW application, export those Web service(s) from the source RMW application and import them to the destination RMW application in insert mode.

Perform the following steps to export and import Web services:

Step 1: Export existing Web service (s) from source RMW application

1. Open the webservices_criteria.properties file.
2. Specify the search attribute values to export the Web services.
3. Check the following section for using different search attributes.
4. Export the Web services using the following command:

```
<migration_home>./export.sh -d Development -u admin/admin -c
<migration_home>/templates/criteria/Webservice_criteria.xml -o
<migration_home>/webservices.xml
```

Step 2: Import the exported Web services to destination RMW application

1. Import the exported xml using the following command:

```
<migration_home>./import.sh -d Development -u admin/admin -i
```

```
<migration_home>/webservices.xml -op INS
```

Step 3: Check the newly added Web service through RMW Web user interface.

1. Open the RMW Web URL in the the Web browser.
2. Log in to the application using valid user credentials.
3. Navigate to **Administration > Library > Manage Record**.
4. Click the **Browse** tab.
5. Select category **Base>System Folder>Integration Folder>Web services**.
6. Click the **Search** tab.
7. Provide the required search criteria, you can use the similar criteria values used for export, like, Web service ID, service name, and so on.
8. Search for the record and verify the newly imported Web services shown in the search result.
9. You can click on the Web service ID to view details.

Updating Existing Web Services

To update a Web service in the destination RMW application, export those Web service(s) from the source RMW application and import them to the destination RMW application in update mode.

Perform the following steps to export and import Web services.

Step 1: Export existing Web services from source RMW application

1. Open the webservices_criteria.properties file.
2. Specify the search attribute values to export the button callbacks.
3. Export the Web services using the following command:

```
<migration_home>./export.sh -d Development -u admin/admin -c
<migration_home>/templates/criteria/websevice_criteria.xml -o
<migration_home>/webservices.xml
```

Step 2: Import the exported Web services to destination RMW application

1. Import the exported xml using the following command:

```
<migration_home>./import.sh -d Development -u admin/admin -i
<migration_home>/webservices.xml -op UPD
```

Step 3: Check the newly added Web services through RMW Web user interface

1. Open the RMW Web URL in the Web browser.
2. Log in to the application using valid user credentials.
3. Navigate to **Administration > Library > Manage Record**.
4. Click the **Browse** tab.
5. Select category **Base>System Folder>Integration Folder>Webservices**.
6. Click the **Search** tab.

7. Provide the required search criteria, you can use the similar criteria values used for export, like, Web service ID, service name, and so on.
8. Search for the record and verify the newly imported Web services shown in the search result.
9. You can click on the Web service ID to view details. Check if the modified information is updated in the destination RMW application.

Deleting Existing Web Services

To delete existing Web services in the destination RMW application, export those Web services and import them to the destination RMW application in delete mode.

Perform the following steps to export and import Web services:

Step 1: Export existing Web service (s) from source RMW application

1. Open the webservices_criteria.properties file.
2. Specify the search attribute values to export the button callbacks.
3. Export the Web services using the following command:

```
<migration_home>./export.sh -d Development -u admin/admin -c  
<migration_home>/templates/criteria/webservice_criteria.xml -o  
<migration_home>/webservices.xml
```

Step 2: Import the exported Web services to destination RMW application

1. Import the exported xml using the following command:

```
<migration_home>./import.sh -d Development -u admin/admin -i  
<migration_home>/webservices.xml -op DEL
```

Step 3: Check the deleted Web service through RMW Web user interface

1. Open the RMW Web URL in the the Web browser.
2. Login to the application using valid user credentials.
3. Navigate to **Administration > Library > Manage Record**.
4. Click the **Browse** tab.
5. Select category **Base>System Folder>Integration Folder>Web services**.
6. Click the **Search** tab.
7. Provide the required search criteria, you can use the similar criteria values used for export, like, Web service ID, service name, and so on.
8. Search for the record and verify the deleted Web services are not shown in the search result.

Web Service Operations

The Web service operation templates are used to insert, update or delete Web service operations and related information from one RMW application to another RMW application. When exporting Web service operation details using Web service templates, not all Web service operation related information are exported. The related integration application setup information, Web service to which this Web service operation belongs, and so on) will not be exported. This should be exported and imported separately, and should be available in the destination RMW application before importing a Web service.

Template files:

Criteria file: webservice_operation_criteria.xml

Schema file: webservice_operation_schema.xml

Properties file: webservice_operation_criteria.properties

Using Search Attributes to Export Web Service Operations

You can use the following search attributes to export a Web Service operation, as described here.

- **Web Service ID:** Provide the Web service ID for search attribute \$\$_WEBSERVICE_ID. For example, \$\$_WEBSERVICE_ID=WeighScaleService
- **Web Service Name:** Provide the service name for search attribute \$\$_SERVICE_NAME. For example, \$\$_SERVICE_NAME=WeighScaleService
- **Operation Name:** Provide the operation name for search attribute \$\$_OPERATION_NAME. For example, \$\$_OPERATION_NAME=printLabel

Inserting New Web Service Operations

To insert new Web service operation to the destination RMW application, export those Web service operation(s) from the source RMW application and import them to the destination RMW application in insert mode.

Follow these steps to export and import Web services.

Step 1: Export existing Web service operation(s) from source RMW application

1. Open the webservice_operation_criteria.properties file.
2. Specify the search attribute values to export the Web service operations.
3. Check the preceding section for using different search attributes.
4. Export the Web services operations using the following command:

```
<migration_home>./export.sh -d Development -u admin/admin -c
<migration_home>/templates/criteria/webservice_operation_criteria.xml
-o <migration_home>/webservice_operations.xml
```

Step 2: Import the exported Web service operations to destination RMW application

1. Import the exported xml using the following command:

```
<migration_home>./import.sh -d Development -u admin/admin -i  
<migration_home>/webservice_operations.xml -op UPD
```

Step 3: Check the newly added Web service operations through RMW Web user interface

1. Open the RMW Web URL in the Internet Explorer Web browser.
2. Log in to the application using valid user credentials.
3. Navigate to **Administration > Library > Manage Record**.
4. Click the **Browse** tab.
5. Select category **System Folder > Integration Folder> Web service Operation**.
6. Click the **Search** tab.
7. Provide the required search criteria, you can use the similar criteria values used for export, like, Web service ID, service name, and so on.
8. Search for the record and verify the newly imported Web service operations shown in the search result window.
9. Click on the Web service operation to verify the Web service operation details.

Updating Existing Web Service Operations

To update existing Web service operation in the destination RMW application, export those Web service operation(s) from the source RMW application and import them to the destination RMW application in update mode.

Follow the following steps to export and import Web services.

Step 1: Export existing Web service operation(s) from source RMW application

1. Open the webservice_operation_criteria.properties file.
2. Specify the search attribute values to export the Web service operations.
3. Check the preceding section for using different search attributes.
4. Export the Web services operations using the following command:

```
<migration_home>./export.sh -d Development -u admin/admin -c  
<migration_home>/templates/criteria/webservice_operation_criteria.xml  
-o <migration_home>/webservice_operations.xml
```

Step 2: Import the exported Web service operations to destination RMW application

1. Import the exported xml using the following command:

```
<migration_home>./import.sh -d Development -u admin/admin -i  
<migration_home>/webservice_operations.xml -op UPD
```

Step 3: Check the updated Web service operations through RMW Web user interface

1. Open the RMW Web URL in the Internet Explorer Web browser.

2. Login to the application using valid user credentials.
3. Navigate to **Administration > Library > Manage Record**.
4. Click the **Browse** tab.
5. Select category **System Folder > Integration Folder > Web service Operation**.
6. Click the **Search** tab.
7. Provide the required search criteria, you can use the similar criteria values used for export, like, Web service ID, service name, and so on.
8. Search for the record and verify the updated Web service operations shown in the search result window.
9. Click on the Web service operation to verify the Web service operation details.

Deleting Existing Web Service Operations

To update existing Web service operation in the destination RMW application, import the Web service operation xml to destination RMW application in delete mode.

Follow the following steps to export and import Web services.

Step 1: Export existing Web service operation(s) from source RMW application

1. Open the webservice_operation_criteria.properties file.
2. Specify the search attribute values to export the Web service operations. Check the preceding section for using different search attributes.
3. Export the Web services operations using following command:

```
<migration_home>./export.sh -d Development -u admin/admin -c
<migration_home>/templates/criteria/webservice_operation_criteria.xml
-o <migration_home>/webservice_operations.xml
```

Step 2: Import the exported Web service operations to destination RMW application

1. Import the exported xml using the following command:

```
<migration_home>./import.sh -d Development -u admin/admin -i
<migration_home>/webservice_operations.xml -op DEL
```

Step 3: Check the deleted service operations through RMW Web user interface

1. Open the RMW Web URL in Internet Explorer Web browser.
2. Log in to the application using valid user credentials.
3. Navigate to **Administration> Library > Manage Record**.
4. Click the **Browse** tab.
5. Select category **System Folder > Integration Folder > Web service Operation**.
6. Click the **Search** tab.
7. Provide the required search criteria, you can use the similar criteria values used for export, like, Web service ID, service name, and so on.

8. Search for the record and verify the deleted Web service operations are not shown in the search result window.

Action-to-Web-Service Operations

Action to Web service operation templates can be used to insert, update, or delete the Web service operation – button callback. This link is used to link an existing ‘button callback’ to an existing ‘Web service operation’.

This procedure using ‘Action – Web service link’ template, exports or imports only the link between a Web service operation and button callback. The referenced Web service operation and button callback should be available in the destination RMW application prior to import.

Template files:

Criteria file: `action_webserviceoperation_criteria.xml`

Schema file: `action_webserviceoperation_schema.xml`

Properties file: `action_webserviceoperation_criteria.properties`

Using Search Attributes to Export Action to Web Service Links

You can use the following search attributes to export an action-web service link, as described here.

- **Button Callback ID:** To export all Web service operation action links for the button callback ID.
Provide the value for the `$$_BUTTON_CALLBACK_ID` search attribute. For example,
`$$_BUTTON_CALLBACK_ID=PCM_INT_PRINT_LABEL`
- **Web service operation name:** To export all Web service operation action links for the Web service operation.
Provide the value for the `$$_OPERATION_NAME` search attribute. For example,
`$$_OPERATION_NAME=printLabel`

Linking Button Callback To Web Service Operation

To link existing buttons to a component in the destination RMW application, export the links from the source RMW application and import them to the destination RMW application.

Perform the following steps to export and import component button links:

Step 1: Export existing ‘Web service operation-action link’ from source RMW application

1. Open the `action_webserviceoperation_criteria.properties` file.
2. Specify the search attribute values to export the Web service operation-action link.
3. Export the links using the following command:

```
<migration_home>./export.sh -d Development -u admin/admin -c  
<migration_home>/templates/criteria/action_webserviceoperation_criteri
```

```
a.xml -o <migration_home>/action_wsoperations.xml
```

Step 2: Import the exported Web service operation to action link to destination RMW application, using the following command:

```
<migration_home>./import.sh -d Development -u admin/admin -i  
<migration_home>/action_wsoperations.xml -op INS
```

Step 3: Check the newly inserted Web service operation action link through RMW Web user interface

1. Open the RMW Web URL in the Web browser.
2. Log in to the application using valid user credentials.
3. Navigate to **Administration > Library > Manage Record**.
4. Click the **Browse** tab.
5. Select category **Base >System Folder>Integration Folder>Action to Web service Operation**.
6. Click the **Search** tab.
7. Provide the required search criteria, you can use the similar criteria values used for export, like, button call back ID, Web service operation ID, and so on.
8. Search for the record and verify that the newly added records are not shown in the search results.
 1. Click on the record to verify the record details.

