Agile Recipe Management for Pharmaceuticals

Import/Export Guide

Release 9.3.4

E39291-02

January 2015

ORACLE

Agile Recipe Management for Pharmaceuticals Import/Export Guide, Release 9.3.4

E39291-02

Primary Author: F. Tabibzade

Contributing Author:

Contributor:

Contents

5 Exporting Data with the User Interface

6 Scenario Examples

Preface

Agile PLM is a comprehensive enterprise PLM solution for managing your product value chain.

Audience

This document is intended for administrators and users of the Agile PLM products.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Related Documents

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 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.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.

Convention	Meaning
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1

About Export/Import Tool

This chapter provides an overview of the Agile RMW Import/Export tool.

Overview

Export and import operations are accomplished in two ways:

- rom 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**, **delete**, or **replace** 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, such as Users, Sites, Views, Roles, Alerts, Workflows, and other Administrator 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.
- Import and Update operations are not supported for Categories without a Primary Key. Attempting an import or update on Categories without a PK will result in an error.
- 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.
- The Pick List has been removed from the Properties file. So, to export Pick List values, provide the database value rather than the value showing in the UI. That is, if a user creates a criteria containing List values as a selection, the db List value name must be provided, instead of the label for searching.

2

Installation and Configuration

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 .bat and .sh 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 (.jar) 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/criteria Properties	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 export or import command files;
- Search criteria values in the template files.

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

3. **For UNIX**, the files to be updated are export.sh or import.sh:

   a. Set BEA=home/bea/Weblogic103/server/lib (Change the path depending upon location of the Weblogic.jar file on your machine)

   b. Set JAVA_HOME=home/bea/jdk160_14_R27.6.5-32 (Change the path depending upon location of this folder on your machine)

   Note: ExIm tool requires the Weblogic.jar file to be present in the home/bea/Weblogic103/server/lib folder to complete its operation. Since ExIm 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 ExIm client on OAS application server.

3

Schema, Criteria, and Criteria Properties Files

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.

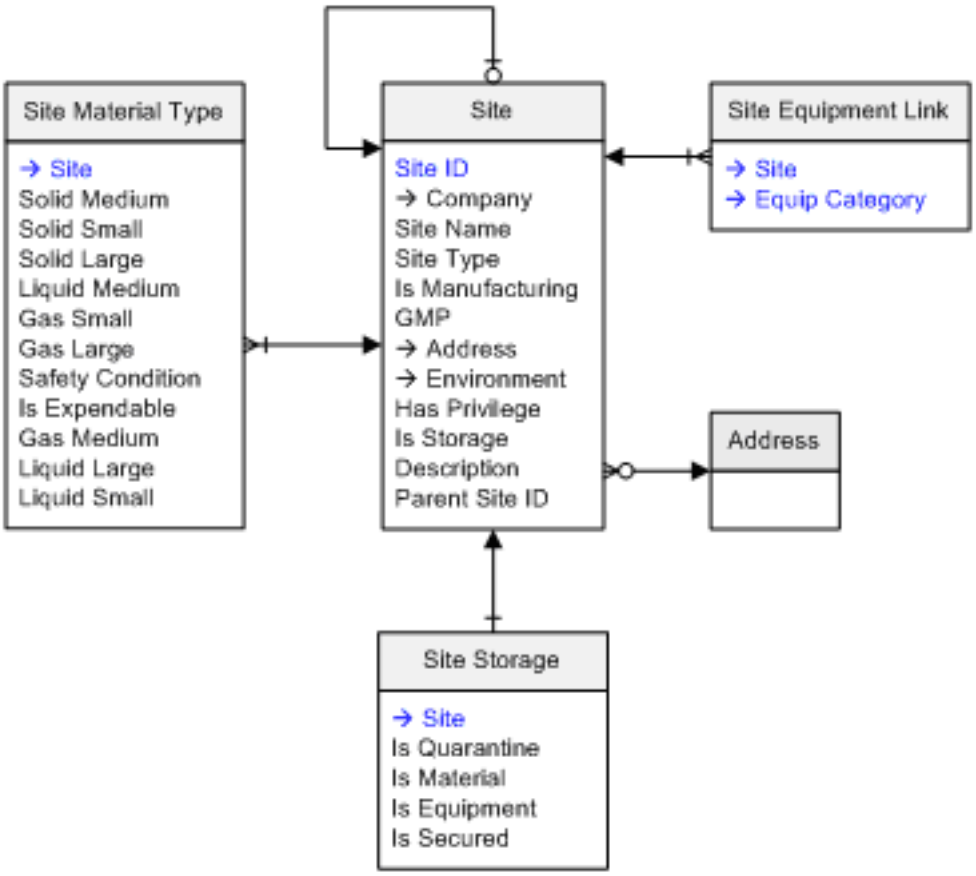
When to Change Values in the Schema File

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>
    <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	<SearchInfo>	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 refid=1, which is the ID of the site category. <SearchCriteria id="1">
Category Name	<CategoryName>	This tag does not contain any value but required for the cfmXML schema.	<CategoryName />

XML Element	Synopsis	Description	Example
Category DB Name	<CategoryDB Name>	This tag contains the database table name of the category.	<CategoryDBName>D_PPL_ADDRESS</CategoryDBName>
Relationship	<Relationship>	The <Relationship> 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.	<Relationship name="Address" refid="2" type="inbound"> <CategoryName></CategoryName> <CategoryDBName></CategoryDBName> <ObjectKey></ObjectKey> </Relationship>
Search Criteria	<Search Criteria>	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 isPrimaryCategory="yes". <SearchCriteria id="2" isPrimaryCategory="yes"> <CategoryName/> <CategoryDBName>D_ORG_SITE</CategoryDBName> </SearchCriteria>

XML Element	Synopsis	Description	Example
Additional Info	<Additional Info>	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.	<AdditionalInfo> <Attribute name= "refreshCache"> <Value dataType= "boolean">no</Valu e> </Attribute> </AdditionalInfo>

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

1. Enter the search criteria value.

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>
      <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	<Search Criteria>	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. 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. Syntax: placeholder=value (no space after the "=" symbol) A search criteria has an ID. It should match with the ID in the schema file.	<SearchCriteria id="2"> … <Attribute name="Site ID"> <Value dataType= "String">$$_SITE_ID </Value> <UnitOfMeasure> </UnitOfMeasure> </Attribute> … </SearchCriteria>

XML Element	Synopsis	Description	Example
Additional Info	<AdditionalInfo>	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. The file names in this tag should not have an extension as the program automatically adds the extension. Also, the file names are optional.	<AdditionalInfo> <Attribute name="schema"> <Value dataType= "file">../schema/Site _ schema</Value> </Attribute> <Attribute name="values"> <Value dataType= "file">../criteria Properties/Site_ criteria </Value> </Attribute>

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
# yyyyy.MMMMM.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
#------------------------------------------------------------------------
```

4

Exporting and Importing Data from the Command Line

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

Export/Import Parameters

These parameters are listed and described in the following tables.

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

Option	Description
-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. 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 -o option
-h	Used to export record from the entire hierarchy tree. 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. 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 -o option.

Import Parameters

Option	Description
-u	Login details for user to access R&MW in <userid>/<Password> format. Password should not contain a "slash" (/).
-d	RMW database name from which the data is being imported. This is the logical name of the database in the R&MW Application being accessed. The database name is found in CFMDBConfig.xml file or RMW login page. The database name is usually 'Production' or 'Development'.
-i	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.
-op	Used to specify the types of transactions that can be performed on the objects present in the database. The transactions are one of the following: UPD, INS, DEL, or RPL. See definitions below.
UPD	Read the import data file specified in the -i parameter. For every object in the file: 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
INS	Read the import data file specified in the -i parameter. For every object in the file, insert a new database object with values from the object in the import file.
DEL	Read the import data file specified in the -i parameter. For every object in the file: 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
RPL	Replace mode: If the data being imported does not exist in the target database, then it is inserted. If the data being imported exists in the target database, then it is updated.

How to Export

To begin exporting information from RMW:

1. Configure the application server information in the export command file.

   For more information, see Updating the Export or Import Command Files.

2. Update the criteria properties template file to export specific information.

   For more information, see Updating Template Files.

3. 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 the import Command File.

   For more information, see Updating the Export or Import Command Files.

2. **Optional:** Update the schema template file for cache refresh.

   For more information, see Updating the Schema template file for Refreshing Cache

3. 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.

4. 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 | RPL>

- For UNIX:

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

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.
- **Replace mode**: If the data being imported does not exist in the target database, then it is inserted. If the data being imported exists in the target database, then it is updated.

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

Batch Exporting and Importing

"BatchML", an XML using tags specified in S-88, is now available for the Work Request, Recipe, and Control Recipe. A typical Recipe may contain Equipment, Materials, or Inventory. When a Recipe is approved in RMW, the approved recipe is exported via BatchML. The same BatchML is imported to Agile PLM 9.3.x and stored against the Item as an Attachment.

Import and Export BatchML is available through the User Interface. Import/Export Batch ML for Work Requests only are available through Web Services in Rel. 9.3.1.1.

Batch Export

Batch Export of a business object exports the business objects and saves different instances in different XML files.

For example, for batch export of a Campaign object, type the command:

BatchExport -d Development -u <username>/<encrypted_pwd> -c <AgileRMW_Home>\exim\templates\criteria\campaign_criteria.xml -o <Output_File_Location>\output\Cam\Cam_{Campaign#Name}.xml

This will create XML file with Campaign Name. The string inside { } will be any attribute -

Cam_{Campaign#Name}.xml

- here the "#" character is not part of the Attribute, it is being used to replace the "space" character in the Attribute name. Campaign Name is the primary key for this object, so for multiple instances of Campaign batch export creates multiple XML files and saves them as cam_{Campaign#Name}.xml

The attribute name should be exactly same as that mentioned in the properties file.

We can export files using patterns. This can export a single business object and save the different instances as different XML files.

Batch File Export

BatchFileExport.bat file is used to do a batch file export of a sequence of objects using properties file.

User can specify multiple criteria and patterns to export using multiple objects at once.

With reference of BatchFileExportREADME.txt file, create batchfileexport.properties in exim\bin folder.

In thebatchfileexport.properties file, specify the criteria:

criteria.keys=1,2...

user=admin/{encrypted password}

db=<db name>

criteria.1.file=<criteria file path>

criteria.1.out=<output file name>

criteria.1.param=<criteria for which you want to export>

criteria.1.seqFileFolder=<sequence file path>

For example:

criteria.1.file=F:/mystarteam/mainline/conformia/pcm/dev/config/migration/criteria/AutoSequence_Criteria.xml

criteria.1.out=F:/dataloader/post/seed/AutoSeq/autoseq_{Name}.xml

criteria.1.param=$$_MET_AUTO_SEQ_NAME=AUDIT_INTERNAL_ID|WF_TASK_ID

criteria.1.seqFileFolder=F:/dataloader/post/seed/AutoSeq/

Likewise, you can mention different criteria in different criteria keys and export more than one object at a time.

This example will export instances based on parameter given for AutoSequence_Criteria object and save them asautoseq_{attribute} file in the specified folder.

The string inside { } will be any attribute -

Cam_{Campaign#Name}.xml

- here the '#' is not part of the Attribute, '#' are used to replace ' '(space) in Attribute name.

Campaign Name is the primary key for campaign object, so for multiple instances of Campaign, batch export creates multiple XML files and saves them as cam_{Campaign#Name}.xml

The attribute name should be exactly same as that mentioned in the properties file.

Execute the command to do BatchFileExport:

BatchFileExport.bat BatchFileExport.properties

Batch Import

We can import data for multiple instances of multiple objects using diferent modes of operation at a time.

With reference of BatchImportREADME.txt file, create batchimport.properties in exim\bin folder.

In thebatchimport.properties file, specify the criteria:

import.keys=0,1

user=admin/{encrypted password}

db=<db name>

import.0.dir=<path of xml files to be imported
import.0.operation=<ins/upd/del/rpl>

For example:

import.0.dir=C:/Agile/Agile9311Build34/AgilePharma/exim/output/menu

import.0.operation=INS

In this example, in the folder menu all the XMLs will be imported in INS mode for menu object.

Execute the command: BatchImport.bat BatchImport.properties

5

Exporting Data with the User Interface

This chapter provides information about Supported Operations and Formats and Searching and Selecting the Export data.

Supported Operations and Formats

The application supports exporting RMW data using RMW's User Interface (UI) or the command-line interface (CLI) with ExIm. 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. ExIm 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

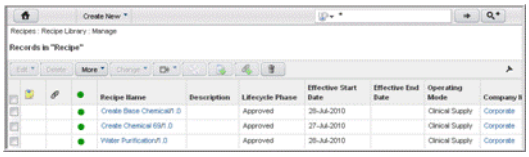
In addition to exporting search results, RMW supports exporting tables of objects that do not have a related object. For example, the table content on all tables under the **History** tab and under the **Notes and Attachments** tab on all objects can be exported in EXCEL, PDF, HTML, or XML formats.

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.

   Records in the Records in "Recipe" are displayed.



4. Select one or more records for export.

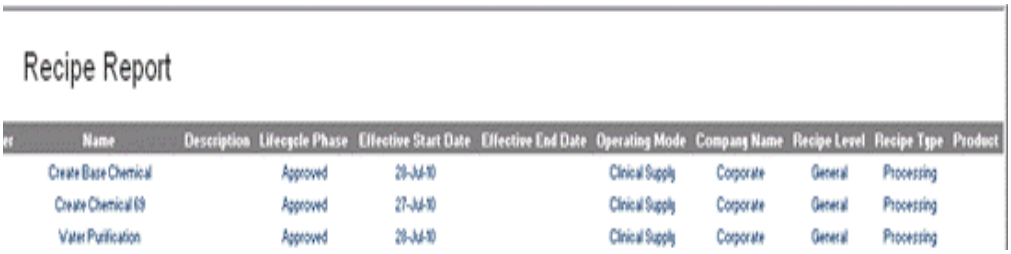
   The Export button in the menu bar is activated.

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:

   Excel sample:



Recipe Report

Name	Description	Lifecycle Phase	Effective Start Date	Effective End Date	Operating Mode	Company Name	Recipe Level	Recipe Type	Product
Create Base Chemical		Approved	28-Jul-10		Clinical Supply	Corporate	General	Processing	
Create Chemical 69		Approved	27-Jul-10		Clinical Supply	Corporate	General	Processing	
Water Purification		Approved	28-Jul-10		Clinical Supply	Corporate	General	Processing	

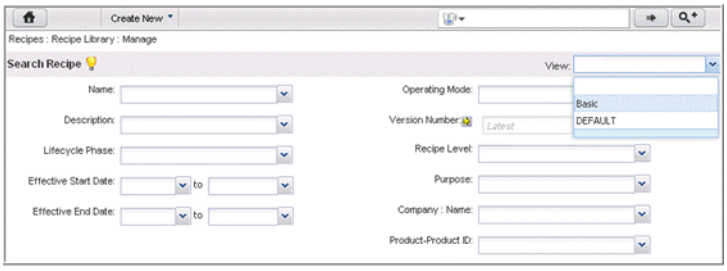
Object + Related Object sample:

```
<?xml version="1.0" encoding="UTF-8" ?>
- <!--
            Generated using SUMS 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
  -->
- <cfmXML>
  - <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>
          - <Attribute name="Version Number">
              <Value dataType="Basic Text">1.0</Value>
              <UnitOfMeasure />
            </Attribute>
```

To Export data using Custom Search:

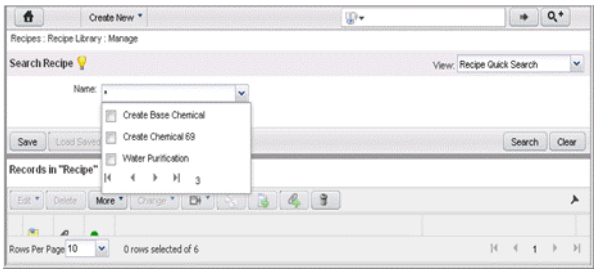
1. Repeat steps 1 and 2 in To export data using Execute Search.
2. In the **Top Navigation** bar, click the Custom Search button.

   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.



3. In the selected View, click the drop-down arrows to select the desired value for each criteria and then click . 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.



4. To continue, select the records you want to 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.

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.

6

Scenario Examples

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 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 export 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** > **Metamodel** > **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** > **ManageRecord.**
4. Click the **Browse** tab.
5. Select category **Base** > **MetaModelFolder** > **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. 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 **ViewName** 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.

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 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** > **Metamodel** > **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 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 and Database as **Development**.
3. Navigate to **Administration**> **Library** > **ManageRecord.**
4. Click the **Browse** tab.
5. Select category **Base** > **MetaModelFolder** > **ViewAttribute.**
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) form 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** >e **Metamodel** > **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** > **ManageRecord.**
4. Click the **Browse** tab.
5. Select category **Base** > **MetaModelFolder** > **ViewAttribute.**
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) form 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 exported view attributes to the 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 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** > **Metamodel** > **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** > **ManageRecord.**
4. Click the **Browse** tab.
5. Select category Base>MetaModelFolder>ViewAttribute and click the Search tab.
6. Provide the required search criteria, you can use the similar criteria values used for export.
7. 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.

<TBD>

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

1. Export the category using the following command:

<migration_home>./export.sh -d Development -u admin/admin -c <migration_home>/templates/category_criteria.xml -o <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** > **Metamodel** > **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** > **Metamodel** > **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 depicts deleting 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** > **Metamodel** > **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

1. Export category attribute using attribute label.

Provide category name using $$_ATTR_LABEL=property. For example, $$_ATTR_LABEL==AlertName

1. 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 -o <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** > **Metamodel** > **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 do not 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** > **Metamodel** > **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** > **Metamodel** > **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.

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**>**SystemFolder**>**IntegrationFolder**>**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 Step 1: Export existing Web service(s) from source RMW application.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 UPD

Step 3: Check the newly added Web service 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** > **ManageRecord.**
4. Click the **Browse** tab.
5. Select category **Base**>**SystemFolder**>**IntegrationFolder**>**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** > **ManageRecord.**
4. Click the **Browse** tab.
5. Select category **Base**>**SystemFolder**>**IntegrationFolder**>**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(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 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>/webservice_operations.xml -op UPD

Step 3: Check the newly added Web service 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 serviceOperation**.
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(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 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>/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(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 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>/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(s) 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_ criteria.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 added Web service 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.

9. Click on the record to verify the record details