Oracle® Fusion Middleware

Creating Schemas with the Repository Creation Utility

12*c* (12.1.3)

E48352-02

August 2014

Documentation for database and system administrators that describes how to create database schemas for Oracle Fusion Middleware products.

ORACLE®

Oracle Fusion Middleware Creating Schemas with the Repository Creation Utility, 12*c* (12.1.3)

E48352-02

Contents

3 Running Repository Creation Utility from the Command Line

A Understanding Repository Creation Utility Screens

B   Understanding Repository Creation Utility Schemas, IDs, and Tablespaces

C   Extending Repository Creation Utility to Configure Custom Application Repositories

D   Troubleshooting Repository Creation Utility

Preface

The *Creating Schemas with the Repository Creation Utility* book contains overview information and usage instructions for Oracle Repository Creation Utility (RCU).

Intended Audience

This guide is intended for users who are installing Oracle Fusion Middleware products and are comfortable running some system administration operations, such as creating users and groups, adding users to groups, and installing operating system patches on the computer where Oracle Fusion Middleware products will be installed. Users on UNIX systems need `root` access to run some scripts.

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

For additional information, see the following manuals:

- *Planning an Installation of Oracle Fusion Middleware*

  This document contains important information about planning and preparing for Oracle Fusion Middleware product installations.

- *Installing and Configuring the Oracle Fusion Middleware Infrastructure*

  The Repository Creation Utility is included with the Oracle Fusion Middleware Infrastructure distribution.

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

What's New in This Guide

The following topics introduce the new and changed features of the Repository Creation Utility and other significant changes that are described in this guide, and provides pointers to additional information. This document is the new edition of the formerly titled *Oracle Fusion Middleware Repository Creation Utility User's Guide*.

New and Changed Features for 12*c* (12.1.3)

Repository Creation Utility 12*c* (12.1.3) includes the following new and changed schema creation features:

- Schema creation can be performed in two phases, so that users without the necessary database permissions can generate scripts for schema creation, and users with the necessary database privileges can execute the scripts to complete schema creation.

  For more information, see Section 1.2.1.

- The command line syntax is also updated with the addition of the phased schema creation.

  See Chapter 3.

- A new schema called Service Table enables other Fusion Middleware applications to obtain schema information.

  For more information, see Section 1.2.4

Other Significant Changes in this Document for 12*c* (12.1.3)

For 12*c* (12.1.3), this guide has been updated in several ways. Following are the sections that have been added or changed.

- Added a section to describe the different phases of schema creation. See Section 1.2.1.

- Added a section to describe how to organize your schemas using custom prefixes depending on your specific topology. See Section 1.2.5.

- Revised the section on obtaining RCU; for 12*c* (12.1.3) RCU is no longer available as a stand-alone product, but instead is included with the Oracle Fusion Middleware Infrastructure distribution. Section 2.1.

- Revised the section on starting RCU. Since the method of obtaining RCU is different, the location from where you start RCU is also different. See Section 2.2.

- Modified the schema creation instructions. Two section on schema creation are now available, based on the database privileges of the user. See Section 2.3.1 and Section 2.3.2.

- Modified the silent schema creation instructions. Sections on generating system load scripts and data load from the command line are added. See Section 3.5 and Section 3.6.

1

Understanding Repository Creation Utility

This chapter introduces you to the concepts and features you should be familiar with before using the repository Creation Utility (RCU).

Many of the Oracle Fusion Middleware components require the existence of schemas in a database prior to installation. These schemas are created and loaded in your database using the Repository Creation Utility (RCU).

This chapter contains the following content:

- Section 1.1, "Verifying RCU System and Database Requirements"
- Section 1.2, "Preparing for Schema Creation"
- Section 1.3, "Using RCU with Java Access Bridge (Windows Only)"

1.1 Verifying RCU System and Database Requirements

This section contains links to important information about supported platforms for RCU, certified databases, and database configuration information. Read this information carefully before you obtain and run RCU.

1.1.1 Verifying Supported Platforms

To see the platforms on which you can run RCU, review the "RCU Supported Platforms" section in the *Oracle Fusion Middleware System Requirements and Specifications* document.

1.1.2 Finding a Certified Database

For a list of certified databases that can be used with RCU, refer to the certification document for your release, located on the *Oracle Fusion Middleware Supported System Configurations* page.

1.1.3 Verifying Your Database Configuration

Before you begin using RCU, review the "Repository Creation Utility (RCU) Requirements" section in the *Oracle Fusion Middleware System Requirements and Specifications* document.

This section contains important information about general and component-specific database requirements that should be met before you run RCU.

Note that not all schemas are supported on all databases. Make sure you have read the information in this section carefully so that you configure a certified database that supports the schemas you need for your Fusion Middleware components.

1.1.4 Verifying Additional Requirements for IBM DB2 Databases

In addition to the typical space and configuration database requirements, IBM DB2 databases also have the following special requirements:

- On IBM DB2 databases running on Linux operating systems, there is a limitation with regards to the length of the schema names.

- One database operating system user must be created for each schema that is created in an IBM DB2 database.

For more information, refer to the "RCU Prerequisites for IBM DB2 Databases" section in the *Oracle Fusion Middleware System Requirements and Specifications* document.

1.2 Preparing for Schema Creation

This section contains important information and concepts regarding schema creation and organization.

The following topics are covered:

- Section 1.2.1, "Understanding System Load and Product Load"

- Section 1.2.2, "Granting Permissions to Users for Querying System Load Objects"

- Section 1.2.3, "Understanding Custom Prefixes"

- Section 1.2.4, "Understanding the Service Table Schema"

- Section 1.2.5, "Planning Your Schema Creation"

- Section 1.2.6, "Integrating Components Using Declarative XML"

1.2.1 Understanding System Load and Product Load

Schema creation in RCU is performed in multiple phases; each phase requires a different level of access to your database.

- System Load Phase

  During the system load phase, RCU will create the necessary tablespaces and schemas and also the `schema_version_registry`, if it is not already present. One entry for each component will be created in `schema_version_registry`, and the entry will have the appropriate access and status set to "LOADED" in the `schema_version_registry` table.

  These actions must be performed by someone with SYS or SYSDBA privileges; authentication credentials must be provided on the Database Connection Details screen when running RCU.

  If you do not have the necessary privileges, you can select **Prepare Scripts for System Load** on the Create Repository screen. This will generate a SQL script containing all the same SQL statements and blocks that would have been called if RCU were to execute the actions for the selected components. After the script is generated, a user with the necessary SYS or SYSDBA privileges can execute the script to complete the system load phase.

  After the system load phase is complete, any user can then run RCU again to complete the schema creation by performing the product load phase.

> **Note:** If you need to generate the script for system load, you can only create schemas on Oracle and Oracle EBR databases; the system load script is not supported on any other database.
>
> If you are performing the system load with full SYS or SYSDBA privileges, then you can create the schemas on any certified database.

- Product Load Phase

   During the product load phase, RCU will create procedures, functions, tables, indexes and other objects within schemas and run any action that does not require DBA access. Any non DBA user or the REGISTRYOWNER user can be used for this step.

- Optional Post-Product Load Phase

   This optional step is needed for components that must execute product load scripts requiring DBA privileges.

1.2.2 Granting Permissions to Users for Querying System Load Objects

If you want to give a user with limited privileges the ability to query the system load objects, you must grant the user the following before they can query the system load:

> **Note:** This user will be used for connecting to the system for queries, but the generated scripts from the system load phase must be run by someone with DBA privileges.

```
grant select_catalog_role to user;
grant select any dictionary to user;
grant create session to user;
grant select on schema_version_registry to user;
```

> **Note:** You may encounter a "Table or view does not exist" error message when you execute the last command; this can be safely ignored.

After performing the system load, grant the following to the same user in order for them to perform data load:

```
grant REGISTRYACCESS to user;
```

1.2.3 Understanding Custom Prefixes

Schemas in your database can be grouped together using custom prefixes.

> **Note:** For important information regarding custom prefixes in IBM DB2 databases, refer to "Size Limit for Schema Prefixes" in the *Oracle Fusion Middleware System Requirements and Specifications* document.

The prefix is prepended to and separated from the schema name with an underscore (_) character, as shown below:

prefix_schemaname

Prefixes:

- Can only contain alpha-numeric characters; no spaces or other special characters.

- Must begin with a letter.

- Must not be longer than 12 characters

The default prefix used by RCU is DEV; if DEV has already been used, then RCU will default to DEV1, then DEV2, and so on. Prefixes are used to create and organize logical groups of schemas. For example, you may want to create a test version of the Metadata Services (schema name MDS) called TEST_MDS; then, when are ready for your production version, you can create a second version of the schema called PROD_MDS. Both TEST_MDS and PROD_MDS may reside on the same or separate databases.

You are only allowed to use a prefix once per schema within a single database. For example, if you had a version of the Metadata Services schema called DEV_MDS, then you can not use the DEV prefix again to create another version of the Metadata Services schema (for example, DEV_MDS2).

If you want to create another version of the schema using the same prefix, you must first drop the existing schema and then create the schema again.

The mapping between the prefixes and schemas is maintained in schema_version_ registry.

1.2.4 Understanding the Service Table Schema

The Service Table schema is a special schema that is installed automatically whenever RCU is run. The service table stores basic schema configuration information (for example, schema prefixes and passwords) that can then be accessed and used by other Oracle Fusion Middleware components during domain creation.

For example, the configuration wizard has screens which you can configure to use the data stored in the service table when RCU was run. After you provide the service table schema credentials, the data from the service table is used to populate the fields on the screen, thus saving you the need to manually populate that data yourself.

Once created, service tables are used to wire Oracle Fusion Middleware components together. For more information, see "Cross-Component Wiring" in *Administering Oracle Fusion Middleware*.

1.2.5 Planning Your Schema Creation

This section provides examples to help you understand how schemas can be grouped together and distributed depending on your specific environment. The following examples are provided:

- Organizing Schemas on a Single Database for a Single Domain

- Organizing Schemas on Multiple Databases for a Single Domain

- Organizing Schemas on a Single Database for Multiple Domains

- Organizing Schemas on Multiple Databases for Multiple Domains

Example 1 Organizing Schemas on a Single Database for a Single Domain
Figure 1–1 shows a set of schemas in a single database being used by a single WebLogic domain. This is the simplest scenario in which all schemas using the DEV prefix are grouped together and used by this single WebLogic domain.
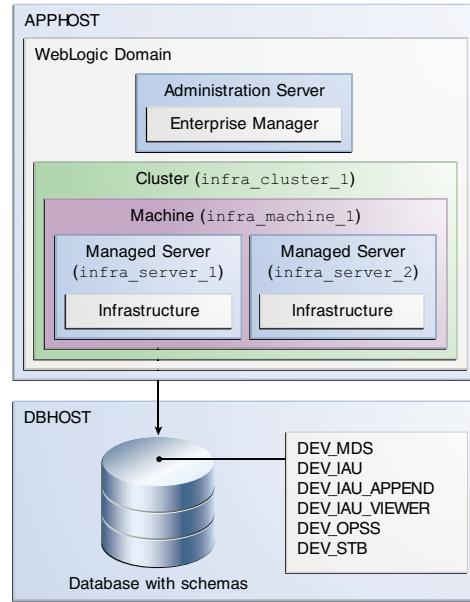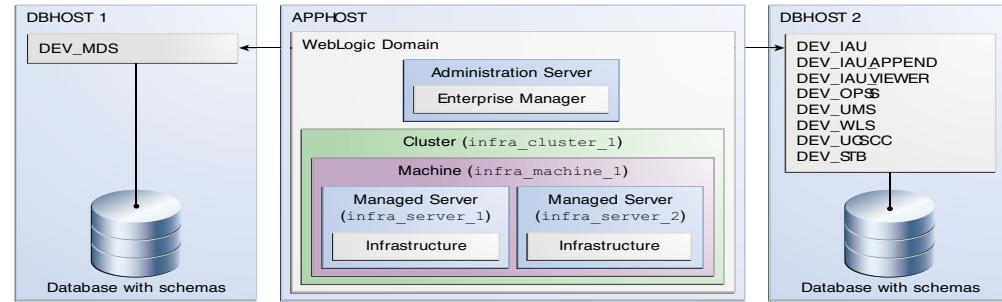
Figure 1–1   Schemas on a Single Database for a Single Domain

Example 2   Organizing Schemas on Multiple Databases for a Single Domain
Figure 1–2 shows a single set of schemas distributed on multiple databases being used by a single WebLogic domain.
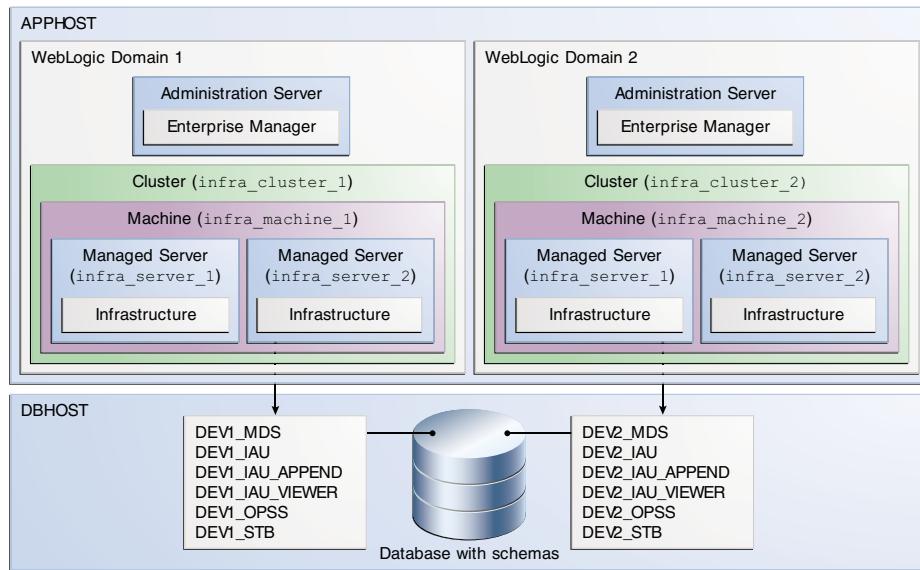
Figure 1–2   Schemas on Multiple Databases for a Single Domain

Note that the same schema prefix (in this case, DEV) can be used to group these related schemas together, even across multiple databases.

Example 3   Organizing Schemas on a Single Database for Multiple Domains
Figure 1–3 shows how schemas on a single database should be grouped for multiple domains.
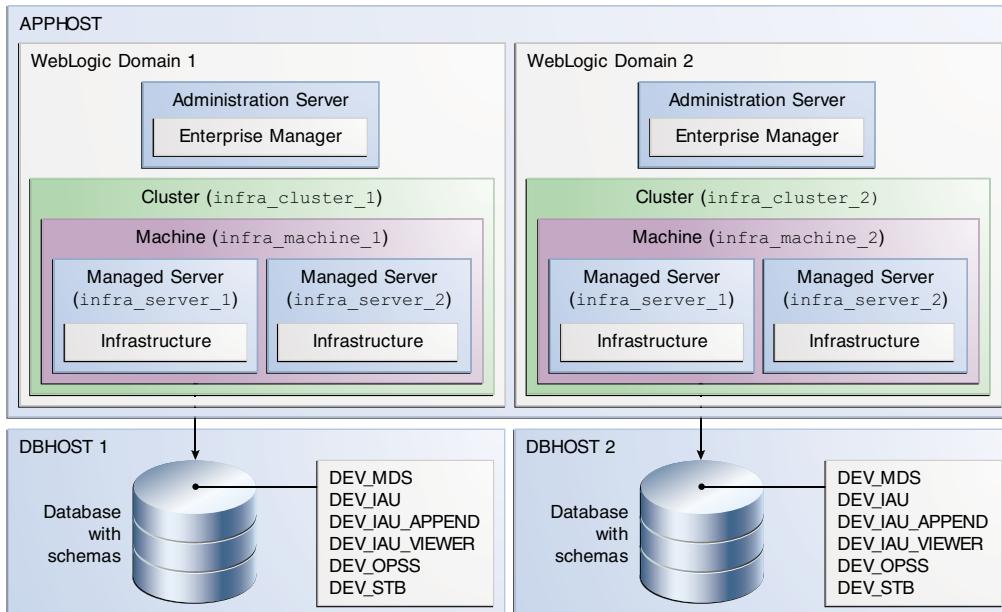
Figure 1–3   Schemas on a Single Database for Multiple Domains

In this example, the prefixes are grouped together by using DEV1 for one set of schemas (used by WebLogic Domain 1), and DEV2 for the second set (used by WebLogic Domain 2).

It is not possible to share a single set of schemas across multiple domains; each domain must have its own set of schemas.

Example 4 Organizing Schemas on Multiple Databases for Multiple Domains

Figure 1–4 shows one way to organize schemas on multiple databases for use with multiple WebLogic domains.

Figure 1–4   Schemas on Multiple Databases for Multiple Domains

Note that in this scenario it is possible to have separate domains on the same host use schemas with the same name and prefix (DEV), since the schemas are located on different databases.

1.2.6 Integrating Components Using Declarative XML

RCU provides extensibility with XML DTDs. Using these DTDs, component owners can integrate their components and prerequisites with RCU by providing a configuration file that adheres to the provided DTD.

For more information, refer to Appendix C, "Extending Repository Creation Utility to Configure Custom Application Repositories".

1.3 Using RCU with Java Access Bridge (Windows Only)

Java Access Bridge enables assistive technologies, such as JAWS screen reader, to read Java applications running on the Windows platform. Assistive technologies can read Java-based interfaces, such as Oracle Universal Installer and Oracle Enterprise Manager.

1.3.1 Install Java Access Bridge

To install Java Access Bridge:

1. Download Java Access Bridge from the following URL:

   `http://java.sun.com/javase/technologies/accessibility/accessbridge/`
2. Install Java Access Bridge.

3. Copy the `access-bridge.jar` and `jaccess-1_4.jar` from your installation location to the `jre\lib\ext` directory.

4. Copy the `WindowsAccessBridge.dll`, `JavaAccessBridge.dll`, and `JAWTAccessBridge.dll` files from your installation location to the `jre\bin` directory.

5. Copy the `accessibility.properties` file to the `jre\lib` directory.

1.3.2 Configure RCU to Use Java Access Bridge

To configure RCU to use Java Access Bridge after you complete the installation, set the system variable `ORACLE_OEM_CLASSPATH` to point to the installed Java Access Bridge files:

1. Display **System** in the Control Panel.

2. Select the **Advanced** tab.

3. Click the **New** button under the System Variable list. The New System Variable dialog appears.

4. In the Variable Name field, enter `ORACLE_OEM_CLASSPATH`.

5. In the Variable Value field, enter the full path to `access-bridge.jar` and `jaccess-1_4.jar`.

   Use a semicolon to separate the two paths. Do not use quotes or character spaces.

6. Click **OK**.

2

Obtaining and Running Repository Creation Utility

This chapter contains instructions for obtaining and running the Repository Creation Utility (RCU).

This chapter contains the following sections:

- Section 2.1, "Obtaining RCU"
- Section 2.2, "Starting RCU"
- Section 2.3, "Creating Schemas"
- Section 2.4, "Dropping Schemas"

2.1 Obtaining RCU

In 12*c* (12.1.3), RCU is available with the Oracle Fusion Middleware Infrastructure distribution. For information about how to install this software and obtain RCU, see *Installing and Configuring the Oracle Fusion Middleware Infrastructure*.

2.2 Starting RCU

After Oracle Fusion Middleware Infrastructure is installed, start RCU from the *ORACLE_HOME*/`oracle_common/bin` directory.

> **Note:** If you are running RCU using a non-English database, you will need to set the following language environment variables: LANG, LC_ALL, and NLS_LANG. Use the environment commands that are appropriate for your environment.
>
> For example, for UNIX operating systems running csh enter the following:
>
> ```
> setenv LANG en_US.UTF8
> setenv LC_ALL $LANG
> setenv NLS_LANG american_america
> ```

On Linux operating systems:

```
cd ORACLE_HOME/oracle_common/bin
./rcu
```

On Windows operating systems:

```
cd ORACLE_HOME\oracle_common\bin
rcu.bat
```

RCU provides a command line interface in situations where Xserver is not available or you have access to telnet terminals without display capabilities. The command line interface also allows you to embed RCU from command line scripts or with some Oracle Fusion Middleware components (for example, Enterprise Manager).

For more information using the CLI, see Chapter 3.

2.3 Creating Schemas

This section contains the following:

- Section 2.3.1, "Creating Schemas as a User with Full SYS or SYSDBA Privileges"
- Section 2.3.2, "Creating Schemas as a User With Limited Database Privileges"
- Section 2.3.3, "Verifying Schema Version Numbers"

2.3.1 Creating Schemas as a User with Full SYS or SYSDBA Privileges

If you are a user with full SYS or SYSDBA privileges, and are able to provide valid authentication credentials for database access, follow the instructions in Table 2–1 to create schemas.

Click on the screen name to see more detailed information for that screen. Unless otherwise noted, click **Next** to continue to the next screen.

Table 2–1   Schema Creation Steps for Full-Privilege Users

Screen	Description
Welcome	This screen introduces you to RCU. You can choose to skip this screen each time you start RCU by selecting **Skip This Page Next Time**.
Create Repository	Select **Create Repository**, then select **System Load and Product Load**.
	If you do not have full SYS or SYSDBA privileges on the database, select **Prepare Scripts for System Load** and follow the schema creation instructions in Section 2.3.2.
Database Connection Details	Specify your database connection credentials.
	Remember that if you are creating schemas on an IBM DB2 database, you must have already created one operating system user for each schema you want to create. See Section 1.1.4 for more information.
	Click **Next** when you have specified your credentials. A separate dialog window will appear while RCU checks connectivity and some database prerequisites. When the database checking has passed without errors, click **OK** to dismiss the dialog window and go to the next screen.
Select Components (for Create Operation)	Select the components for which you want to create schemas, and specify a prefix to group them together.
	You must remember the prefix and schema names for the components you are installing; you will need this information during the configuration phase of your product installation. Oracle recommends that you write these values down.

Table 2–1 (Cont.) Schema Creation Steps for Full-Privilege Users

Screen	Description
Schema Passwords	Specify the passwords for your schema owners.
	You must remember the passwords you enter on this screen; you will need this information during the configuration phase of your product installation. Oracle recommends that you write these values down.
Custom Variables	This screen appears only if the Oracle home from which you started RCU contains Oracle Data Integrator or Oracle SOA Suite.
	■ If Oracle Data Integrator is present, you will see the Custom Variables for Oracle Data Integrator.
	■ If Oracle SOA Suite is present, you will see the Custom Variables for Oracle SOA Suite.
	■ If both are present, you will see the custom variables for both products.
Map Tablespaces	Use this screen to configure the desired tablespace mapping for the schemas you want to create.
	When you click **Next**, a separate dialog window will appear asking you to confirm that you want to create these tablespaces. Click **OK** to proceed and dismiss the dialog window.
	A second dialog window will appear showing the progress of tablespace creation. After this is complete, click **OK** to dismiss this window and go to the next screen.
Summary (for Create Operation)	Verify the information on this screen, then click **Create** to begin schema creation.
Completion Summary (for Create Operation)	Review the information on this screen to verify that the operation was completed successfully. Click **Close** to complete the schema creation and dismiss RCU.

2.3.2 Creating Schemas as a User With Limited Database Privileges

If you are a user without SYS or SYSDBA privileges, or you are unable to provide valid authentication credentials for database access, follow the instructions in Table 2–2 to create schemas.

Table 2–2 Schema Creation Steps for Limited-Privilege Users

Screen	Description
Welcome	This screen introduces you to RCU. You can choose to skip this screen each time you start RCU by selecting **Skip This Page Next Time**.
Create Repository	Select **Create Repository**, then select **Prepare Scripts for System Load**.
	See Section 1.2.1 for more information about these operations.
Database Connection Details	Specify your database connection credentials.
	Click **Next** when you have specified your credentials. A separate dialog window will appear while RCU checks connectivity and some database prerequisites. When the database checking has passed without errors, click **OK** to dismiss the dialog window and go to the next screen.
	NOTE: Performing system load and product load separately is only supported on Oracle or Oracle EBR databases.

Table 2–2   (Cont.) Schema Creation Steps for Limited-Privilege Users

Screen	Description
Select Components (for Create Operation)	Select the components for which you want to create schemas, and specify a prefix to group them together.
	You must remember the prefix and schema names for the components you are installing; you will need this information during the configuration phase of your product installation. Oracle recommends that you write these values down.
Map Tablespaces	Use this screen to configure the desired tablespace mapping for the schemas you want to create.
	When you click **Next**, a separate dialog window will appear asking you to confirm that you want to create these tablespaces. Click **OK** to proceed and dismiss the dialog window.
	A second dialog window will appear showing the progress of tablespace creation. After this is complete, click **OK** to dismiss this window and go to the next screen.
Summary (for Create Operation)	Verify the information on this screen, then click **Generate** to begin script generation.
	By default, the scripts are located in the `ORACLE_HOME`/`oracle_common/rcu/log/logdir.`*`date_timestamp`* directory. If you want to specify a different location, click **Browse** and select a location on your system.
	The names of the scripts generated are `script_systemLoad.sql` and `script_postDataLoad.sql`.
N/A	After the scripts are created, someone with SYS or SYSDBA privileges should execute the script:
	1. Login to SQL*Plus.
	2. Enter the following command to execute the `script_systemLoad.sql` script (replace the path to the script if you chose to save it in a custom location):
	`@`*`ORACLE_HOME`*`/oracle_common/rcu/log/logdir.`*`date_timestamp`*`/script_systemLoad.sql`
	3. While the script is running, the user will be prompted to set the schema password for each schema being created. Specify the passwords as prompted.
	You will be returned to your system prompt when the script has been executed.
N/A	After the script is executed, start RCU again and perform the product load phase to complete schema creation.
Welcome	This screen introduces you to RCU. You can choose to skip this screen each time you start RCU by selecting **Skip This Page Next Time**.
Create Repository	Select **Create Repository**, then select **Perform Product Load**.
	See Section 1.2.1 for more information about these operations.
Database Connection Details	Specify your database connection credentials.
	Click **Next** when you have specified your credentials. A separate dialog window will appear while RCU checks connectivity and some database prerequisites. When the database checking has passed without errors, click **OK** to dismiss the dialog window and go to the next screen.

Table 2–2   (Cont.) Schema Creation Steps for Limited-Privilege Users

Screen	Description
Select Components (for Create Operation)	From the drop-down list, select the prefix you specified earlier in this procedure.
	Then, select the components for which you want to perform the product load.
Schema Passwords	Specify the passwords for your schema owners.
	You must remember the passwords you enter on this screen; you will need this information during the configuration phase of your product installation. Oracle recommends that you write these values down.
Summary (for Create Operation)	Verify the information on this screen, then click **Create** to begin schema creation.
Completion Summary (for CreateAtion)	Review the information on this screen to verify that the operation was completed successfully. Click **Close** to complete the schema creation and dismiss RCU.

2.3.3 Verifying Schema Version Numbers

When the schemas are created in your database, RCU creates and maintains a table called `schema_version_registry`. This table contains schema information such as version number, component name and ID, date of creation and modification, and custom prefix.

To verify that the schemas are installed properly, run the following query after login in to SQL*Plus:

```
select comp_name, version from schema_version_registry;
```

The `comp_name` argument retrieves the name of the component, and `version` retrieves the version number. For example:

```
SQL> select comp_name, version from schema_version_registry;

COMP_NAME
--------------------------------------------------------------------------------
VERSION
-----------------------------
Audit Service
12.1.2.0.0

Audit Service Append
12.1.2.0.0

Audit Service Viewer
12.1.2.0.0


COMP_NAME
--------------------------------------------------------------------------------
VERSION
-----------------------------
Metadata Services
12.1.2.0.0

Oracle Platform Security Services
12.1.2.0.0
```

```
Service Table
12.1.2.0.0


COMP_NAME
--------------------------------------------------------------------------------
VERSION
------------------------------
Call Control
12.1.2.0.0

User Messaging Service
12.1.2.0.0

WebLogic Services
12.1.2.0.0


9 rows selected.
```

2.4 Dropping Schemas

To drop schemas from the database, start RCU (see Section 2.2), then follow the instructions in Table 2–3.

Click on the screen name to see more detailed information for that screen. Unless otherwise noted, click **Next** to continue to the next screen.

Table 2–3   RCU Screens and Description for Dropping Schemas

Screen	Instructions and Action Required
Welcome	This screen introduces you to RCU. You can choose to skip this screen each time you start RCU by selecting **Skip This Page Next Time**.
Create Repository	Select **Drop Repository**.
Database Connection Details	Specify the connection details for your database, then click **Next**.
	A separate dialog window will appear while RCU checks connectivity and some database prerequisites. When the database checking as passed without errors, click **OK** to dismiss the dialog window and go to the next screen.
Select Components (for Drop Operation)	Select the prefix and the schemas you want to drop, then click Next.
	A separate dialog window will appear asking you to verify that you want to drop the selected schemas. Click **OK** to dismiss this window.
	A second dialog window appears while RCU checks the prerequisites for the schemas you are dropping. After this is complete, click **OK** to dismiss this window and go to the next screen.
Summary (for Drop Operation)	Review the information on this screen, then click **Drop** to drop the schemas.
Completion Summary (for Drop Operation)	Note the location of the log files, then click **Close** to dismiss the screen.

2.4.1 Dropping Shared Tablespaces

Tablespaces that are shared among multiple schemas will not be dropped. For example, if you created both the Audit Services (for example, DEV_IAU) and Metadata Services (for example, DEV_MDS) schemas, both schemas would use the temporary tablespace DEV_IAS_TEMP (see Section A.8.1, "Default Tablespace Mappings").

If you then drop the DEV_IAU schema, the DEV_IAS_TEMP tablespace would not be dropped since it is also being used by the DEV_MDS schema.

2.4.2 Dropping Schemas and Deleting Datafiles (Windows Only)

If you used RCU to drop a schema from a Windows-based database, and you want to recreate the dropped schema, you will have to manually delete datafiles that were not automatically removed when the schema was dropped.

Navigate to the DB_HOME\oradata directory and manually delete any remaining datafiles before recreating the schema. Oracle recommends that you check this directory for any remaining datafiles before you attempt to recreate any dropped schema.

3

Running Repository Creation Utility from the Command Line

This chapter describes how to run RCU from the command line.

The command-line interface (CLI) is necessary for integration with both the Oracle Fusion Middleware installer and Enterprise Manager during application deployment. Additionally, you can use the CLI in cases where Xserver is not configured or if you are using a telnet terminal that does not have proper display capabilities.

This chapter contains the following topics:

- Section 3.1, "Command Line Syntax and Parameters"
- Section 3.2, "Using the -silent Command"
- Section 3.3, "Using the -interactive Command"
- Section 3.4, "Creating a Repository from the Command Line"
- Section 3.5, "Generating a System Load Script From the Command Line"
- Section 3.6, "Loading Data Into the Repository From the Command Line"
- Section 3.7, "Dropping a Repository from the Command Line"
- Section 3.8, "RCU Environment Variables"

3.1 Command Line Syntax and Parameters

The syntax for the RCU command line interface is:

```
rcu mode operation {parameters}
```

There are two modes (`-silent` and `-interactive`) and four operations (`-createRepository`, `-generateScript`, `-dataLoad`, and `-dropRepository`) so the command can be written out as follows:

```
rcu [-silent | -interactive] [-createRepository | -generateScript | -dataLoad |
-dropRepository] {parameters}
```

The tables in this section describe each of the modes, operations, and parameters:

- Table 3–1, " RCU Modes Available from the Command Line"
- Table 3–2, " RCU Operations Available from the Command Line"
- Table 3–3, " RCU Command Line Parameters and Descriptions"

Table 3–1 lists and describes the RCU modes available from the command line.

Table 3–1   RCU Modes Available from the Command Line

Command	Description
`-silent`	Run RCU with minimal or no interaction from the command line.
	For more information, see Section 3.2.
`-interactive`	Run the RCU graphical interface. This is the default if neither `-silent` nor `-interactive` is specified.
	This command (whether specified or not) allows you to pre-populate certain screens with information as specified from the command line. You can pre-populate the Create Repository and Database Connection Details.
	For more information, see Section 3.3.

Table 3–2 lists and describes the RCU operations available from the command line.

Table 3–2   RCU Operations Available from the Command Line

Command	Description
`-createRepository`	Run RCU with either `-silent` or `-interactive` to create a repository.
	For more information, see Section 3.4.
`-generateScript`	Run RCU with either `-silent` or `-interactive` to generate a script for system load.
	For more information, see Section 3.5.
`-dataLoad`	Run RCU with either `-silent` or `-interactive` to load data into the repository.
	For more information, see Section 3.6.
`-dropRepository`	Run RCU with either `-silent` or `-interactive` to drop a repository.
	For more information, see Section 3.7.

Table 3–3 lists and describes the various command line parameters.

Table 3–3   RCU Command Line Parameters and Descriptions

Parameter	Required?	Valid for Which Operation?	Description
-compInfoXMLLocation	No	All	Full path to the location of the ComponentInfo.xml file. The default location is *ORACLE_HOME*/oracle_common/rcu/config (on UNIX operating systems) or *ORACLE_HOME*\oracle_common\rcu\config (on Windows operating systems).
-storageXMLLocation	No	All	Full path to the location of the StorageInfo.xml file. The default location is *ORACLE_HOME*/oracle_common/rcu/config (on UNIX operating systems) or *ORACLE_HOME*\oracle_common\rcu\config (on Windows operating systems).
-databaseType	No	All	Type of database to which you are connecting.
			If you are performing the system load and product load concurrently as a user with full DBA privileges, the valid options are ORACLE, SQLSERVER, IBMBD2, MYSQL, or EBR.
			If you do not have permissions to perform system load and product load concurrently and need to generate a script, you can only use the ORACLE and EBR database types.
			For more information about system load and product load, see Section 1.2.1.
-connectString	Yes	All	Credentials for connecting to your database.
			For Oracle or EBR-enabled databases, use the following format:
			host:*port*:*sid*
			For all other database types, use:
			server_name/*host*:*port*:*database_name*
-edition	No	All	Edition name. This is only valid if you specify databaseType=EBR.
-dbUser	Yes	All	Database user name (for example, the default user name on Oracle databases is SYS).
-dbRole	No	All	Database user role (for example, SYSDBA for the SYS user on Oracle databases).
-unicodeSupport	No	-createRepository -dropRepository	Specify Yes or No for unicode support. Default is Yes. This is only valid if you specify databaseType=SQLSERVER.
-skipCleanupOnFailure	No	-createRepository -generateScript -dataLoad	Whether or not you want to skip the schema cleanup if schema creation fails. Valid values are Yes or No. The default is No.
-scriptLocation	No	-generateScript	Specify the location to save the generated script for system load repository.
-useSamePasswordForAllSchemaUsers	No	-createRepository -generateScript -dataLoad	Whether or not you want to use the same password for all schemas. Valid values are true or false. The default is false.

Table 3–3   (Cont.) RCU Command Line Parameters and Descriptions

Parameter	Required?	Valid for Which Operation?	Description
-selectDependentsForComponents	No	-createRepository -generateScript -dataLoad	Whether or not you want to have RCU automatically select dependent schemas for your components. Valid values are `true` or `false`. The default is `false`.
-variables	No	All	Comma separated variables in the format *variablename=value*. See Section 3.8 for a list of RCU environment variables.
-schemaPrefix	No	All	Prefix for your schemas. For more information about schema prefixes, refer to Section 1.2.5 and Section A.4.

-component	Yes	All	ID of the component(s) you want to add or drop. To specify a single component, use the format:
			
			`-component component_ID`
			
			To specify multiple components, use the format:
			
			`-component component_ID -component component_ID`
			
			For a list of valid component (schema) IDs, refer to Appendix B. Multiple components are sometimes necessary because of dependencies; for example, you cannot create the Oracle Platform Security Services (`OPSS`) schema if the Audit Services (`IAU`) schemas does not already exist. In this case, you must specify both schemas:
			
			`-component OPSS -component IAU`

-tablespace	Yes	-createRepository -generateScript	Tablespace name of the component. This tablespace must exist in the database if you are using this option.
-tempTablespace	Yes	-createRepository -generateScript	TEMP tablespace name of the component. This tablespace must exist in the database if you are using this option.

3.2 Using the -silent Command

Specify `-silent` if you want to run RCU with minimal interaction once you have entered the commend. You must specify all mandatory command line parameters in the command. For example:

```
rcu -silent -createRepository -connectString database_connect_string -dbUser
-database_user -component component
```

In this scenario, RCU will prompt you for the database and component schema passwords from the command line. If you specify multiple components, you will be prompted for the passwords in the order in which the components are specified.

If you want to avoid all interaction from the command line, you can create a text file containing all the necessary passwords (one password per line) and then use the `-f` option to pass this password file to RCU. For example, if you create a file called `passwordfile.txt`, you can use the command below:

```
rcu -silent -createRepository -connectString database_connect_string -dbUser
-database_user -component component1_name -component component2_name -f <
passwordfile.txt
```

The `passwordfile.txt` file would contain, in order:

```
database_password
component1_schema_password
component2_schema_password
```

It is important to make sure that the passwords in the file are specified in the same order as the components on the command line.

Once the installation is complete the password file must be removed. The passwords are maintained in cleartext format and therefore present a security risk if the password file is left in place after installation.
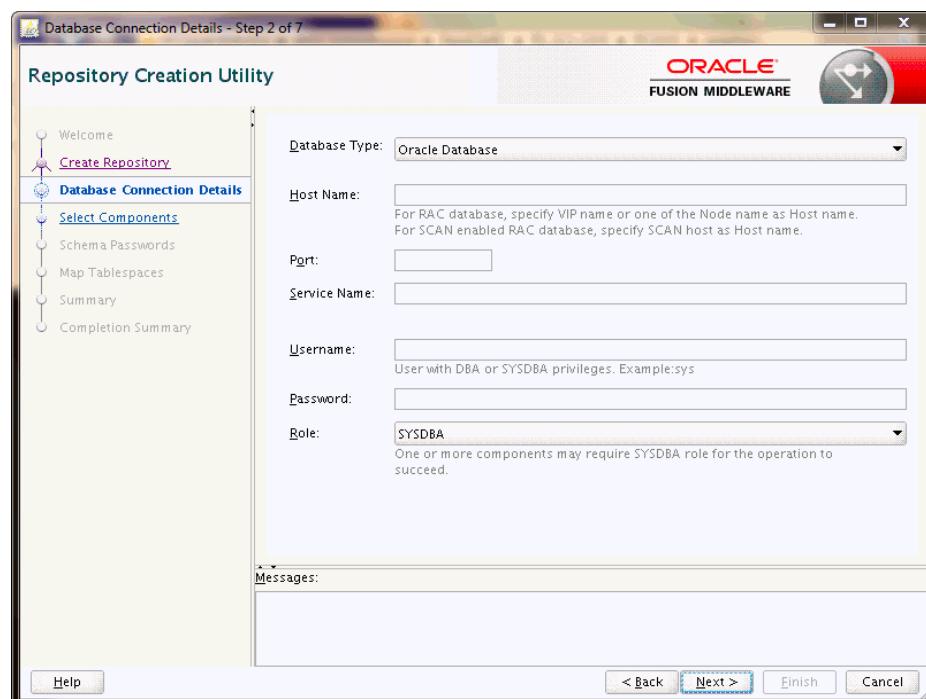
3.3 Using the -interactive Command

Specify `-interactive` to run the RCU graphical interface. This is the default if neither `-silent` nor `-interactive` is specified.

You can specify information from the command line that would be populated in the graphical interface to expedite your RCU operation. For example, if you run RCU with the following command:

```
./rcu
```
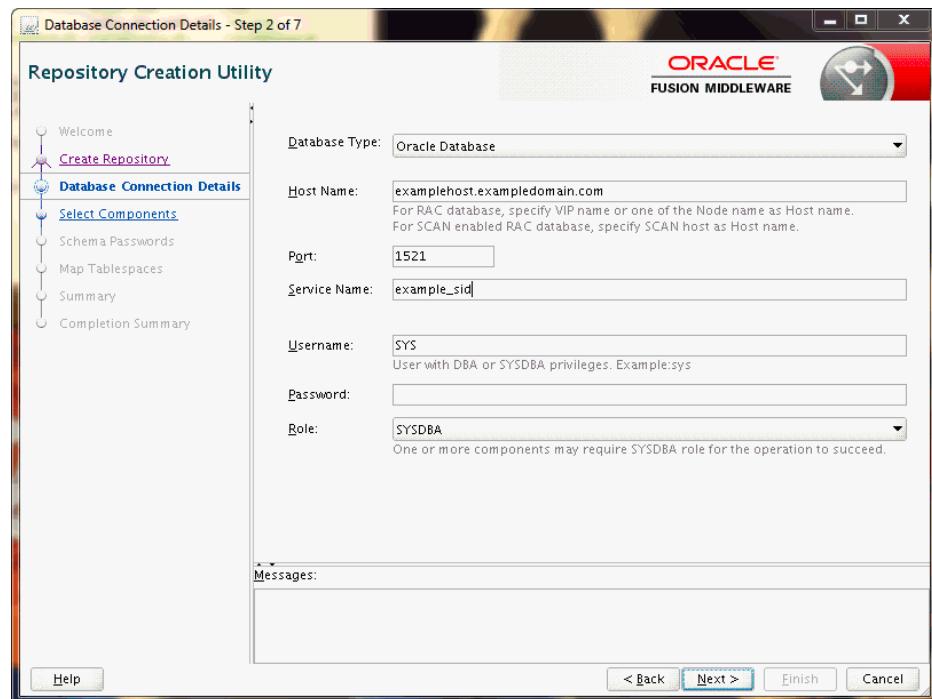
The Database Connection Details page contains blank fields, as shown below:

But if you run RCU with a few parameters from the command line:

```
./rcu -interactive -createRepository -connectString
examplehost.exampledomain.com:1521:example_sid -dbUser SYS -component MDS
```

The Database Connection Details page contains the information supplied from the command line:

3.4 Creating a Repository from the Command Line

The full syntax for the RCU command line interface to create a repository is shown below:

```
rcu [-silent | -interactive] -createRepository
   [-compInfoXMLLocation omponentInfo.xml_file_location]
   [-storageXMLLocation Storage.xml_file_location]
   [-databaseType [ORACLE|EBR]]
   -connectString database_connect_string
   [-edition edition_name]
   -dbUser database_username
   [-dbRole database_user_role]
   [-unicodeSupport [Yes|No]]
   [-skipCleanupOnFailure [Yes|No]]
   [-useSamePasswordForAllSchemaUsers [true|false]]
   [-selectDependentsForComponents [true|false]]
   [-variables variablename=value]
   [-schemaPrefix schema_prefix]
   -component component_ID
   -tablespace component_tablespace_name
   -tempTablespace component_temp_tablespace_name
```

In order to work properly, make sure that the parameters are specified in the same order that they are listed. For example, do not specify the -compInfoXMLLocation parameter before the -component parameter.

When specifying the -component, you must use the correct component IDs, which are listed in Appendix B.

Before you create any schemas, you must be aware of and specify all dependencies for the component you are loading. For example, the SOAINFRA schema depends on the MDS and ORASDPM schemas; if you try to load the SOAINFRA schema without specifying both the MDS and ORASDPM schemas, or if the MDS and ORASDPM schemas do not already exist in the database, RCU will stop before any loading takes place.

Below is a sample command to create the OPSS schema on a UNIX operating system:

```
./rcu -silent -createRepository -databaseType ORACLE -connectString
examplehost.exampledomain.com:1521:exampleSID -dbUser sys -dbRole sysdba
-schemaPrefix TEST -component OPSS -component MDS
```

3.5 Generating a System Load Script From the Command Line

The full syntax for the RCU command line interface to generate a system load script is shown below:

```
rcu [-silent | -interactive] -generateScript
   [-compInfoXMLLocation omponentInfo.xml_file_location]
   [-storageXMLLocation Storage.xml_file_location]
   [-databaseType [ORACLE|EBR]]
   -connectString database_connect_string
   [-edition edition_name]
   -dbUser database_username
   [-dbRole database_user_role]
   [-skipCleanupOnFailure [Yes|No]]
   [-scriptLocation script_location]
   [-useSamePasswordForAllSchemaUsers [true|false]]
   [-selectDependentsForComponents [true|false]]
   [-variables variablename=value]
   [-schemaPrefix schema_prefix]
   -component component_ID
   -tablespace component_tablespace_name
   -tempTablespace component_temp_tablespace_name
```

In order to work properly, make sure that the parameters are specified in the same order that they are listed. For example, do not specify the -compInfoXMLLocation parameter before the -component parameter.

When specifying the -component, you must use the correct component IDs, which are listed in Appendix B.

Before you create a script, you must be aware of and specify all component dependencies. For example, the SOAINFRA schema depends on the MDS and ORASDPM schemas; if you specify the SOAINFRA schema without specifying both the MDS and ORASDPM schemas, or if the MDS and ORASDPM schemas do not already exist in the database, RCU will stop before competing the operation.

Below is a sample command to create a system load script on a UNIX operating system:

```
./rcu -silent -generateScript -databaseType ORACLE -connectString
examplehost.exampledomain.com:1521:exampleSID -dbUser sys -dbRole sysdba
-scriptLocation /home/Oracle/Products/Oracle_Home/oracle_
common/rcu/log/logdir.date_timestamp -schemaPrefix TEST -component OPSS -component
MDS
```

3.6 Loading Data Into the Repository From the Command Line

The full syntax for the RCU command line interface to load data into a repository is shown below:

```
rcu [-silent | -interactive] -dataLoad
   [-compInfoXMLLocation omponentInfo.xml_file_location]
   [-storageXMLLocation Storage.xml_file_location]
   [-databaseType [ORACLE|EBR]]
   -connectString database_connect_string
   [-edition edition_name]
   -dbUser database_username
   [-dbRole database_user_role]
   [-skipCleanupOnFailure [Yes|No]]
   [-useSamePasswordForAllSchemaUsers [true|false]]
   [-selectDependentsForComponents [true|false]]
   [-variables variablename=value]
   [-schemaPrefix schema_prefix]
   -component component_ID
```

Below is a sample command to perform a data load on a UNIX operating system:

```
./rcu -silent -dataLoad -databaseType ORACLE -connectString
examplehost.exampledomain.com:1521:exampleSID -dbUser sys -dbRole sysdba
-schemaPrefix TEST -component OPSS -component MDS
```

3.7 Dropping a Repository from the Command Line

The full syntax for the RCU command line interface to drop a repository is shown below:

```
rcu [-silent | -interactive] -dropRepository
   [-compInfoXMLLocation ComponentInfo.xml_file_location]
   [-storageXMLLocation Storage.xml_file_location]
   [-databaseType [ORACLE|EBR]]
   -connectString database_connect_string
   [-edition edition_name]
   -dbUser database_username
   [-dbRole database_user_role]
   [-unicodeSupport [Yes|No]]
   [-variables variablename=value]
   [-schemaPrefix schema prefix]
   -component component_ID
```

In order to work properly, make sure that the parameters are specified in the same order that they are listed. For example, do not specify the -compInfoXMLLocation parameter before the -component parameter.

You must also be aware of schema dependencies when dropping schemas (see Appendix B). For example, several schemas require the MDS schema to be present; if you choose to drop the MDS schema, then all the schemas that require the MDS schema will stop working.

Below is a sample command to drop the OPSS schema on a UNIX operating system:

```
./rcu -silent -dropRepository -databaseType ORACLE -connectString
examplehost.exampledomain.com:1521:exampleSID -dbUser sys -dbRole sysdba
-schemaPrefix TEST -component OPSS
```

3.8 RCU Environment Variables

Table 3–4 shows the variables picked up by RCU from the environment. If the environment variable is not set, then RCU uses the default value.

Table 3–4   RCU Environment Variables

Variable	Default	Description
RCU_LOG_LOCATION	*ORACLE_HOME*/oracle_ common/rcu/log (UNIX operating systems) *ORACLE_HOME*\oracle_ common\rcu\log (Windows operating systems)	Location of the RCU log file.
RCU_TIMESTAMP_LOG_DIR	true	Determines whether or not a directory with the format *logdir.yyyy-dd_hh-mm* is created for the RCU log file. Set this variable to true or false.
RCU_LOG_NAME	rcu.log	Name of the RCU log file.
RCU_LOG_LEVEL	ERROR	Determines the RCU log level. Set this variable to one of SEVERE, ERROR, NOTIFICATION, or TRACE.

A
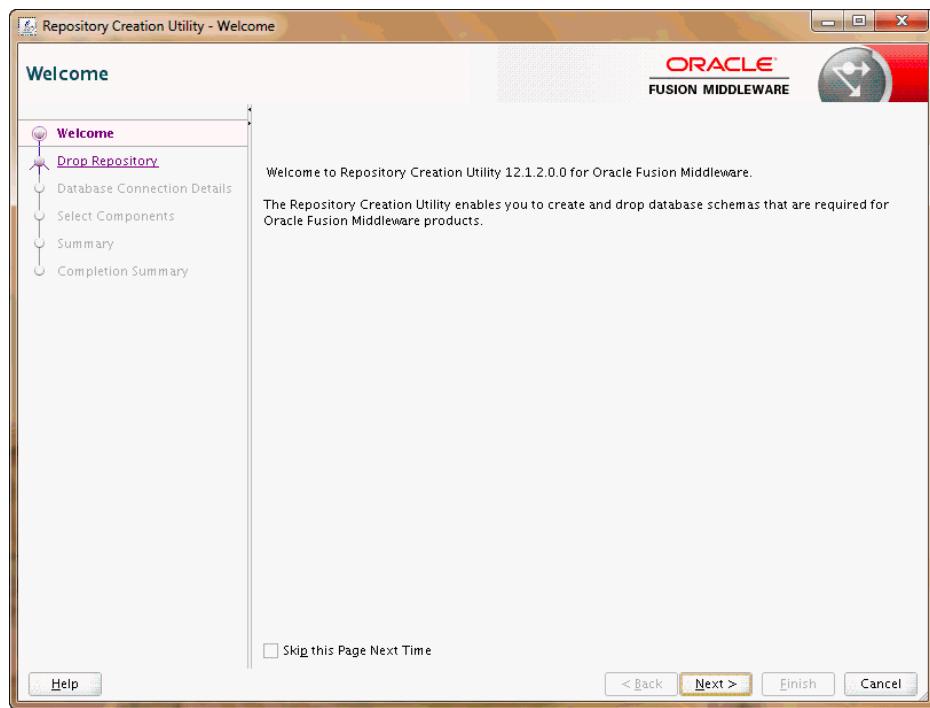
Understanding Repository Creation Utility Screens

This appendix contains screenshots and descriptions for all of the Repository Creation Utility screens.

The following sections are included:

- Section A.1, "Welcome"
- Section A.2, "Create Repository"
- Section A.3, "Database Connection Details"
- Section A.4, "Select Components (for Create Operation)"
- Section A.5, "Select Components (for Drop Operation)"
- Section A.6, "Schema Passwords"
- Section A.7, "Custom Variables"
- Section A.8, "Map Tablespaces"
- Section A.9, "Summary (for Create Operation)"
- Section A.10, "Summary (for Drop Operation)"
- Section A.11, "Completion Summary (for Create Operation)"
- Section A.12, "Completion Summary (for Drop Operation)"
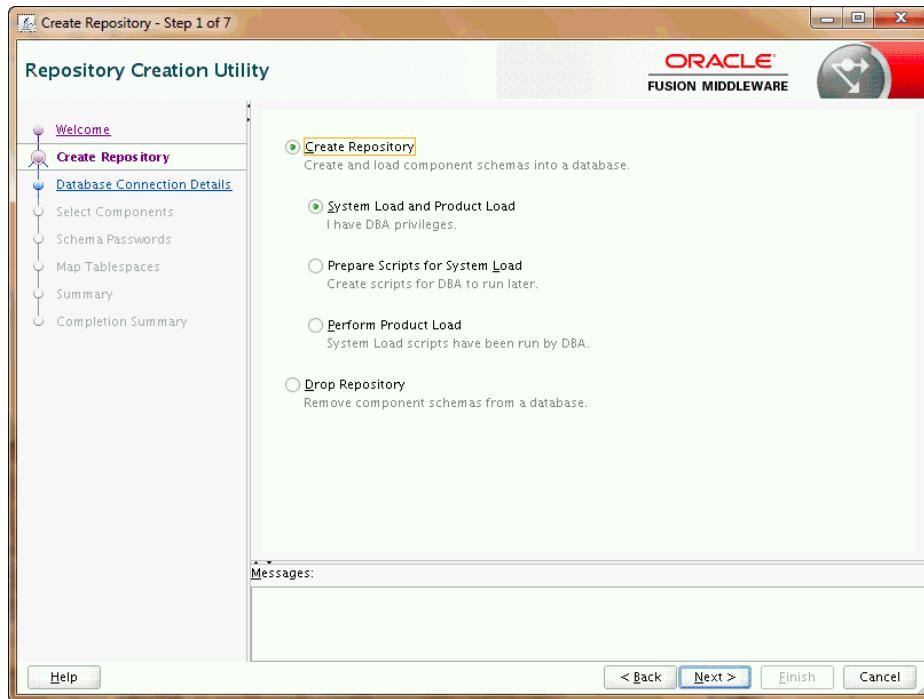
A.1 Welcome

This is the first screen that appears when RCU is started. Note the navigation pane on the left that summarizes the tasks that RCU will help you complete. Each item in the navigation pane represents a specific screen that will prompt you for information required to create or drop your schemas.

Click **Skip This Page Next Time** if you do not want to see the Welcome screen the next time you start RCU.

A.2 Create Repository

Use this screen to select the action you want to perform.

■ Create Repository

■ Drop Repository

A.2.1 Create Repository

Select **Create Repository** to create component schemas in the database. The following table describes the options available:
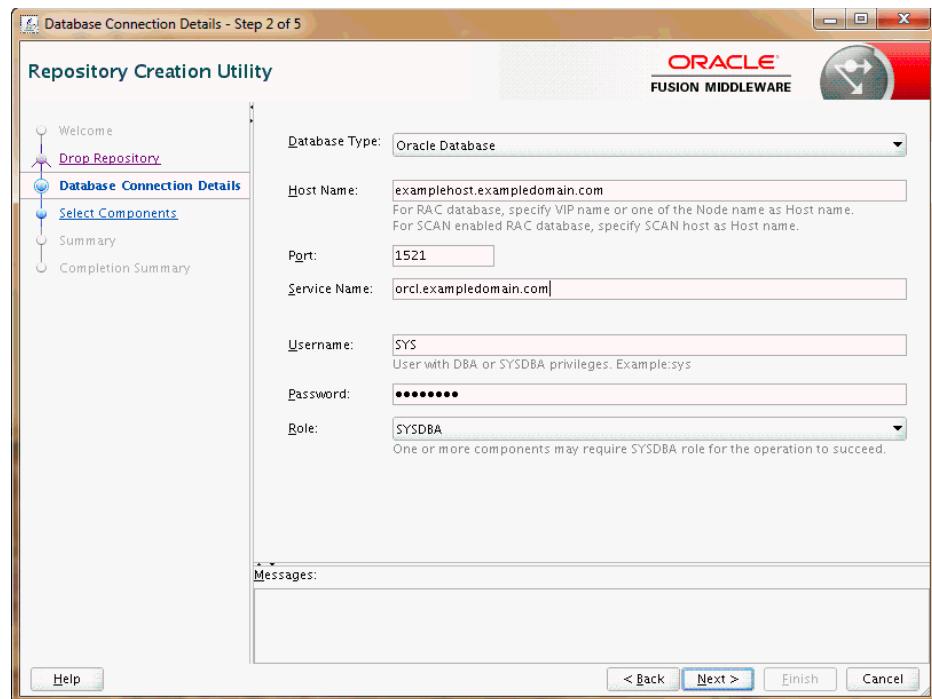
Option	Description
System Load and Product Load Concurrently	Select this option to perform both System Load and Product Load operations.
	Users must have DBA or SYSDBA permissions to select this option.
	This option can be performed on any certified database.
Prepare Script for System Load	Select this option to perform actions that require DBA or SYSDBA permissions:
	■ Creating tablespaces and schemas.
	■ Create the `schema_version_registry` (if not already present).
	■ Create entries in `schema_version_registry` for each selected component, set the proper access permissions, and set the status of the component to "LOADED" in the `schema_version_registry` table.
	This option is only supported on Oracle and Oracle EBR databases.

Option	Description
Perform Product Load Only	Select this option to load and create procedures, functions, tables, indexes and other objects within schemas and run any action that does not require DBA access.
	Any non-DBA user or the REGISTRYOWNER user can select this option.

A.2.2 Drop Repository

Select **Drop Repository** to remove component schemas from the database.
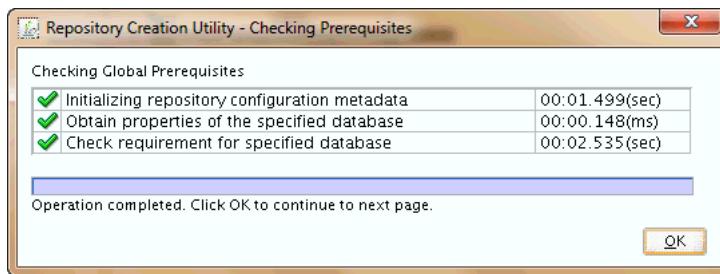
A.3 Database Connection Details

Use this screen to specify the connection credentials to the database in which you will be creating or dropping your schemas. Select one of the following depending on your database:

■ Specifying Connection Credentials for Oracle Databases and Oracle Databases with Edition-Based Redefinition

■ Specifying Connection Credentials for MySQL Databases

■ Specifying Connection Credentials for Microsoft SQL Server Databases

■ Specifying Connection Credentials for IBM DB2 Databases

Click **Next** when you are finished entering the connection credentials for your database. The following screen appears, indicating the progress of the installer establishing the connection with the specified database:

If an error occurs while the connection is being established, the error message(s) appear in the Messages field on the Database Connection Details screen.

Specific database requirements for the various schemas can be found in the *Oracle Fusion Middleware System Requirements and Specifications* document.

For certified database versions, see the *System Requirements and Supported Platforms for Oracle Fusion Middleware 11gR1* document, which is available on the Oracle Fusion Middleware Supported System Configurations page.

A.3.1 Specifying Connection Credentials for Oracle Databases and Oracle Databases with Edition-Based Redefinition

For Oracle databases and Oracle databases with edition-based redefinition, specify the following connection credentials:

Field	Description
Host Name	Enter the name of the server where your database is running. Use the following format: `examplehost.exampledomain.com` For Oracle RAC databases, specify the VIP name or one of the node names in this field.
Port	Enter the port number for your database. The default port number for Oracle databases is 1521.
Service Name	Specify the service name for the database. Typically, the service name is the same as the global database name. If you are unsure what the service name for your database is, you can obtain it from the `SERVICE_NAMES` parameter in the database's initialization parameter file. If the initialization parameter file does not contain the `SERVICE_NAMES` parameter, then the service name is the same as the global database name, which is specified in the `DB_NAME` and `DB_DOMAIN` parameters. For Oracle RAC databases, specify the service name of one of the nodes in this field. For example: `examplehost.exampledomain.com`
Username	Enter the user name for your database. The default user name is `SYS`.
Password	Enter the password for your database user.
Role	Select the database user's role from the drop-down list: ■ Normal ■ SYSDBA

A.3.2 Specifying Connection Credentials for MySQL Databases

For MySQL databases, specify the following connection credentials:

Field	Description
Host Name	Enter the host name, IP address, or complete server name in host\server format of the server where your database is running.
Port	Enter the port number for your database.
Database Name	Specify the name of your database.
Username	Specify the name of a user with administrator privileges.
Password	Enter the password for your database user.

A.3.3 Specifying Connection Credentials for Microsoft SQL Server Databases

For Microsoft SQL Server databases, specify the following connection credentials:

Field	Description
Unicode Support	Select Yes or No from the drop-down list.
Server Name	Enter the host name, IP address, or complete server name in host\server format of the server where your database is running.
Port	Enter the port number for your database.
Database Name	Specify the name of your database.
Username	Specify the name of a user with administrator privileges.
Password	Enter the password for your database user.
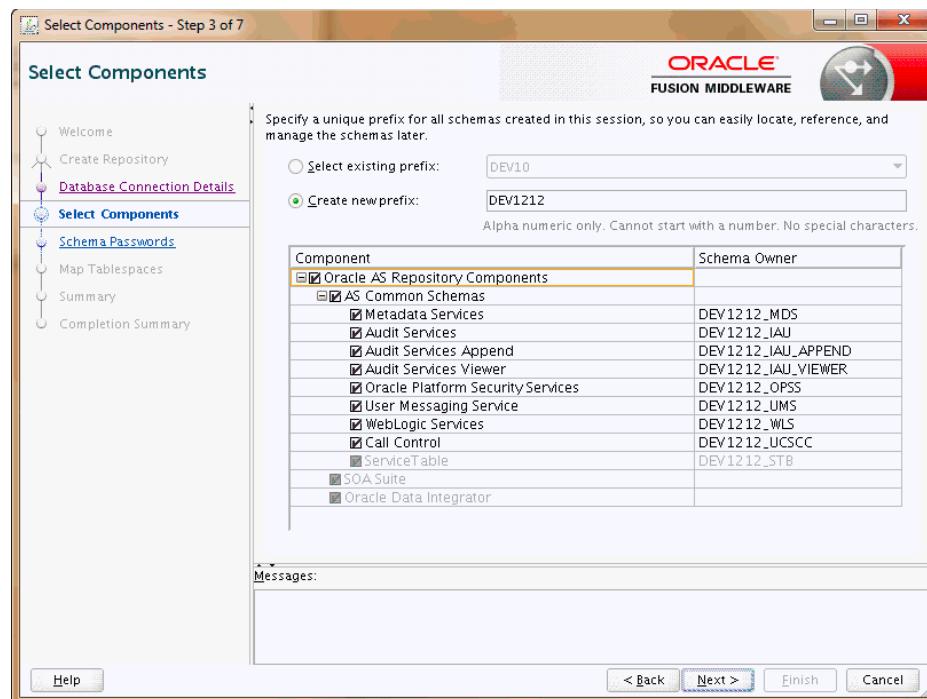
A.3.4 Specifying Connection Credentials for IBM DB2 Databases

For IBM DB2 databases, specify the following connection credentials:

Field	Description
Server Name	Enter the host name, IP address, or complete server name in host\server format of the server where your database is running.
Port	Enter the port number for your database.
Database Name	Specify the name of your database.
Username	Specify the name of a user with DB Owner privileges. The default user name for IBM DB2 databases is db2admin.
Password	Enter the password for your database user.

A.4 Select Components (for Create Operation)

Below is the Select Components screen if you selected **Create** on the Create Repository screen.
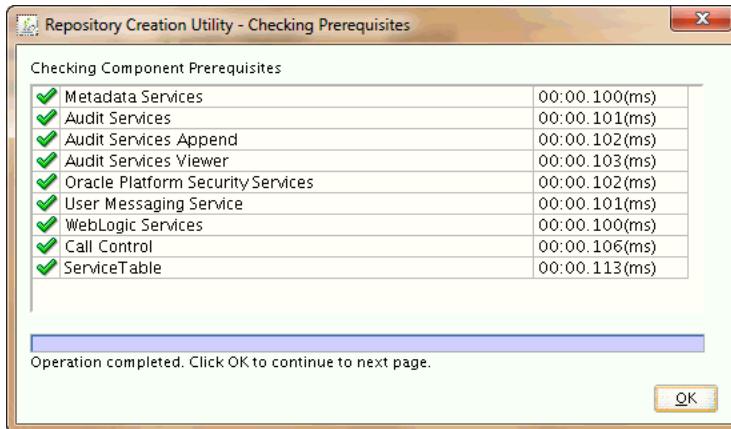
The following table describes the fields on this screen.

Field	Description
Select existing prefix	Use this option to create schemas using an existing prefix which you can select from the drop-down list. If you select this option, you can only create schemas that do not already have this prefix in the database (for example, if DEV_MDS already exists, you will not be able to select the MDS schema on the page to create it again).
Create new prefix	Use this option to create a new custom prefix that can be used to group your schema together. The prefix name must be a minimum of one character in length and cannot exceed 12 alphanumeric characters (0-9, a-z, or A-Z) in length (not including the underscore character). Prefixes should not start with a number. No whitespace or special characters are allowed.
	You can edit the schema name or prefix by clicking the item you want to change in the "Schema Owner" column and editing the table cell directly.
	For more information about custom prefixes, see Section 1.2.3.

Field	Description
Component	Use the "Component" column in the table to select the component schema you want to create.
	When you select a component, any other components that may be required by the component you select are also selected. For example, if you select **Oracle Platform and Security Services**, then the **Audit Services** schema is also automatically selected if it has not already been selected. The **Audit Services** schema is required by the **Oracle Platform and Security Services** schema.
	If a component has a plus sign (+) next to its name, then there are sub components available. Click on the plus sign (+) to expand the category to view all sub components. If you want to select a component with all its subcomponents, click on the top-most box with the plus sign (+).
	If you are creating component schemas on an IBM DB2 database, see important information in Section 1.1.4.
Schema Owner	This column shows the name of the schema owner for each component (custom prefix and schema name). You can edit the schema owner value by clicking the item you want to change in this column and editing the table cell directly.

> **Note:**   You must remember the Schema Owner name for the components you are installing; you will need this information during the configuration phase of your Oracle Fusion Middleware product installation. Oracle recommends that you write this value down.

Click **Next** when you are finished specifying your prefix, schema names, and selecting components. The following screen appears, indicating the progress of component prerequisite checking before the schemas are created.
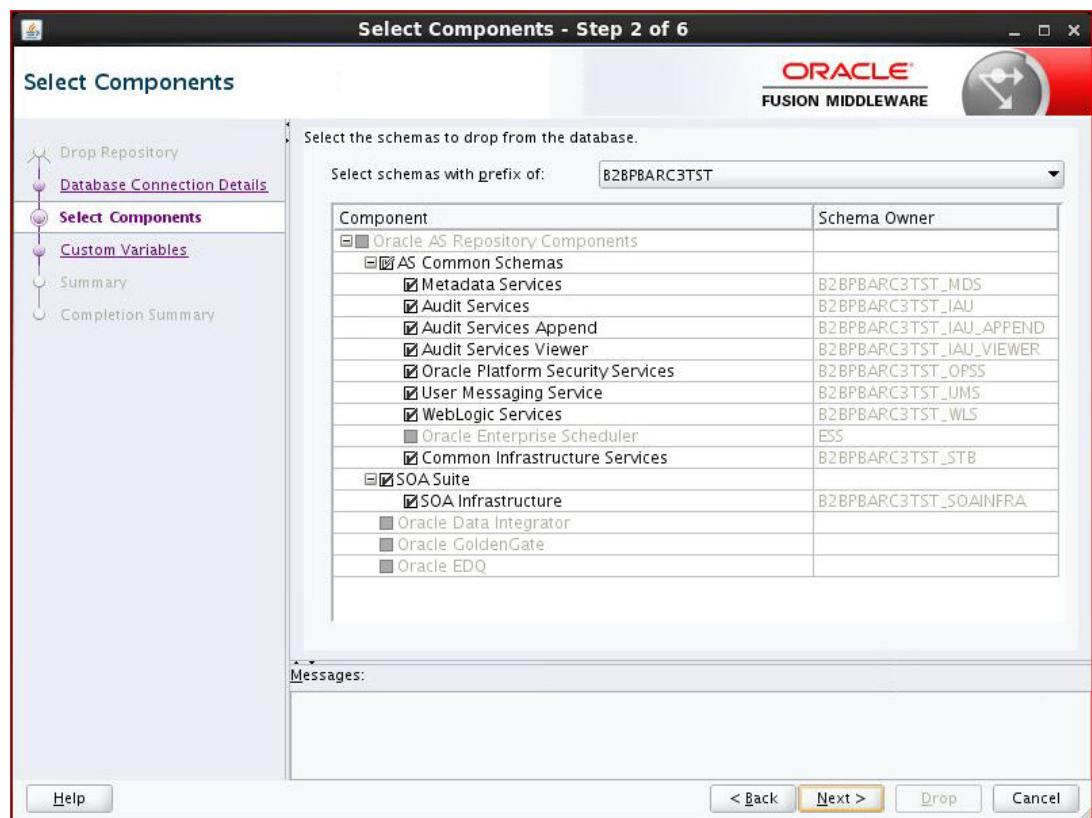
If an error occurs during the prerequisite checking, the error message(s) appear in the Messages field on the Select Components screen.

Click **OK** to dismiss this screen.

A.5 Select Components (for Drop Operation)
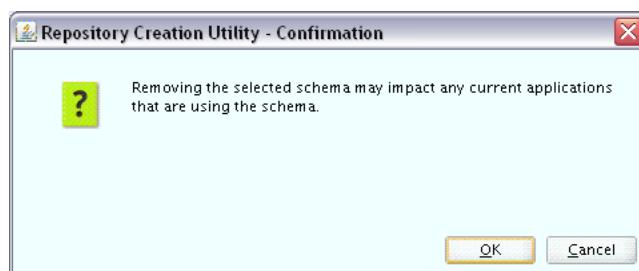
Below is the Select Components screen if you selected **Drop** on the Create Repository screen.
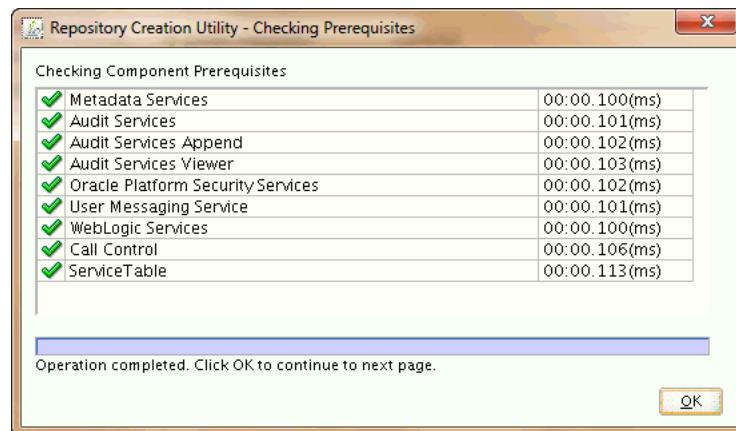
The following table describes the fields on this screen:

Field	Description
Select schemas with prefix of	Use the drop-down list in this field to select an existing custom prefix - all schemas associated with this prefix will be removed.
Component	Select the component schemas you want to remove. **NOTE:** Unlike create operations, schema dependencies are not handled automatically by RCU for drop operations. When you remove a schema, you must also remove the schemas which depend on the schema you are removing. For example, the **Audit Services** schema is required by the **Oracle Platform Security Services** schema; if you remove the **Audit Services** schema, the **Oracle Platform Security Services** schema will no longer work.
Schema Owner	This column shows the schema owner (custom prefix and schema name). For drop operations, this column cannot be edited.

Click **Next** when you are finished. The following screen appears:

Click **OK** to continue. The following screen appears:
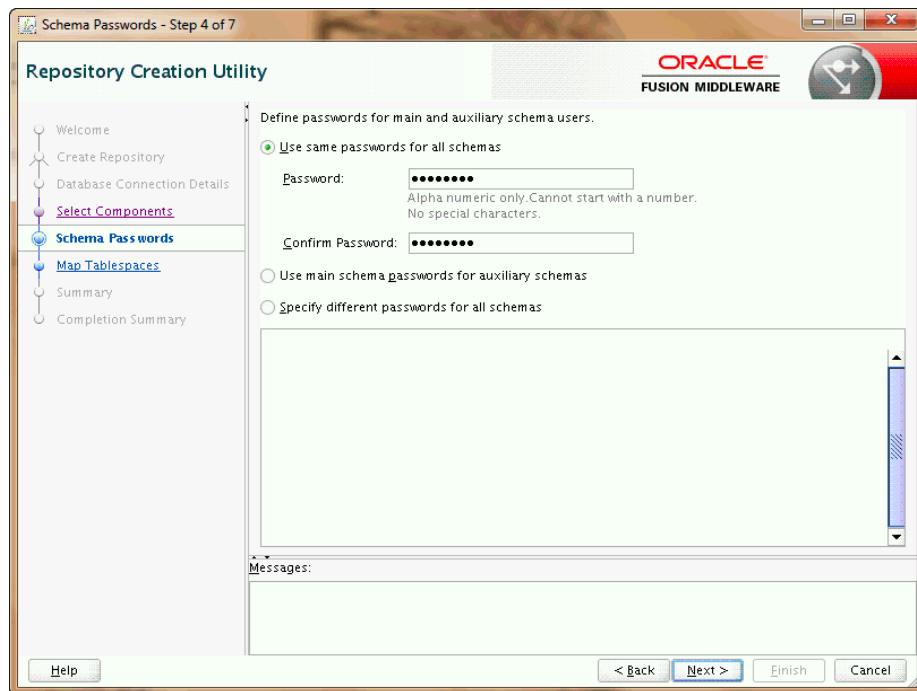
If an error occurs during the prerequisite checking, the error message(s) appear in the Messages field on the Select Components screen.

Click **OK** to dismiss this screen.

A.6 Schema Passwords

Below is the Schema Passwords screen.

There are three ways to specify schema passwords on this screen; they are described in the following table:

Option	Description
Use same password for all schemas	Select this option if you want to use a single password for all schemas and their auxiliary schemas. In the Password field, enter your password. Enter your password again in the Confirm Password field.
Use main schema passwords for auxiliary schemas	Select this option if you want to specify different passwords for the main schemas, but still have the same password used for their respective auxiliary schemas. If you select this option, only the main schemas will be visible in the table. For each schema, you must enter each schema's password in the Schema Password column in the table, and enter the same password in the Confirm Password column.
Specify different passwords for all schemas	Select this option if you want to specify unique passwords for the main schemas and auxiliary schemas. If you select this option, all main schemas and auxiliary schemas will be visible in the table. For each schema and auxiliary schema, you must enter the password in the Schema Password column in the table, and enter the same password in the Confirm Password column.

> **Note:** You must remember the passwords you enter on this screen; you will need this information during the configuration phase of your Oracle Fusion Middleware product installation. Oracle recommends that you write these values down.
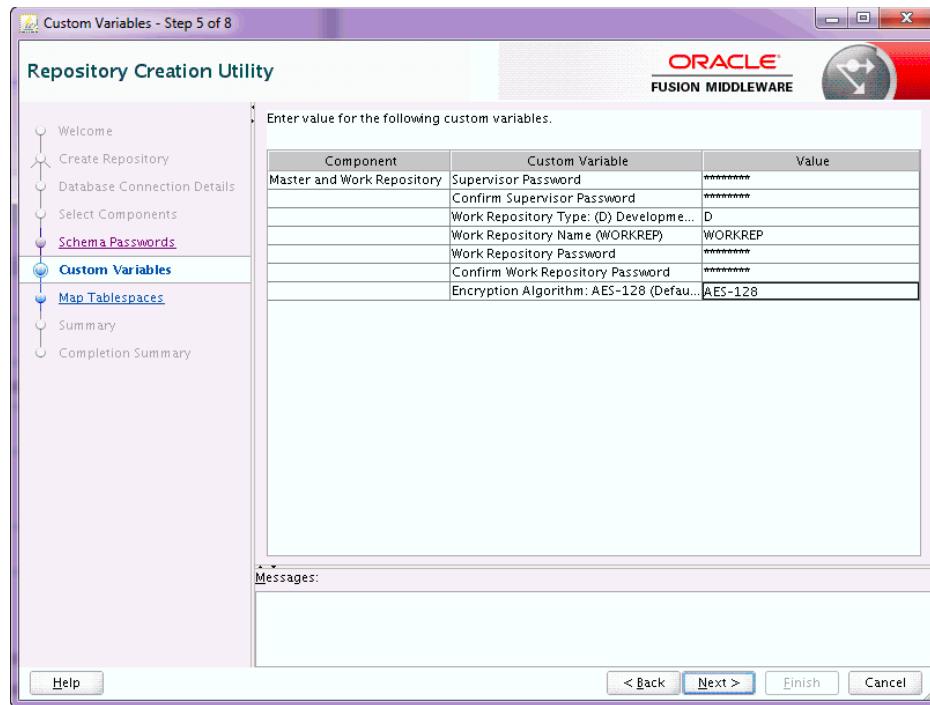
A.7 Custom Variables

Custom variables are required by some products for additional product configuration information. This screen only appears if Oracle Data Integrator or Oracle SOA Suite are detected in the Oracle home in which RCU was started.

■ If Oracle Data Integrator is detected, the Custom Variables for Oracle Data Integrator will be visible.

■ If Oracle SOA Suite is detected, the Custom Variables for Oracle SOA Suite will be visible.

■ If both Oracle Data Integrator and Oracle SOA Suite are detected in the Oracle home, both sets of custom variables will appear on this screen.

A.7.1 Custom Variables for Oracle Data Integrator

Below is the Custom Variables screen for Oracle Data Integrator:
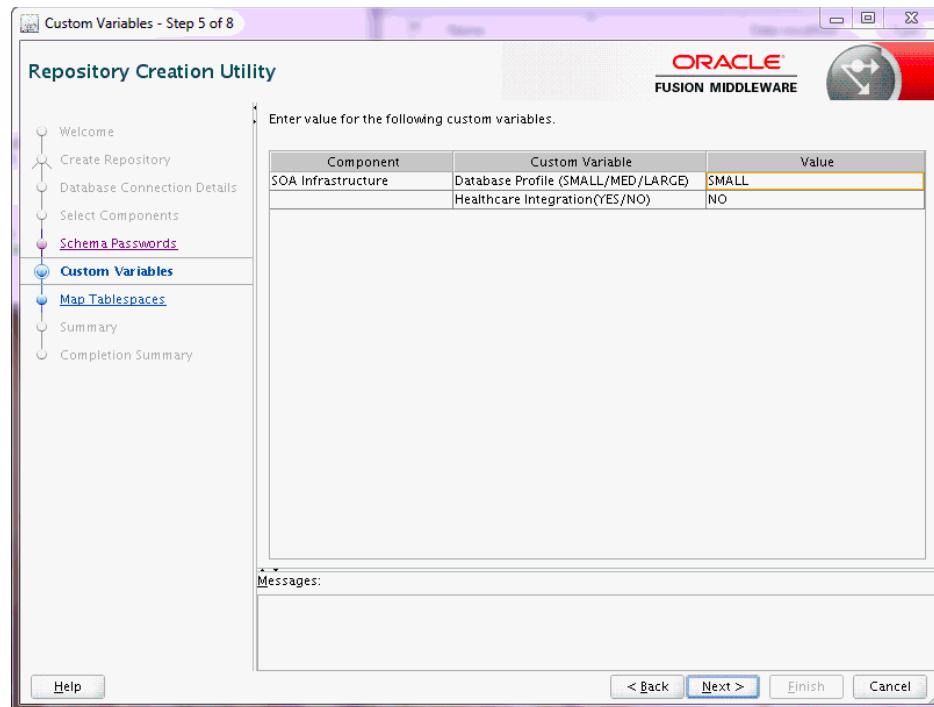
The custom variables for Oracle Data Integrator are described in the following table:

Variable	Description
Supervisor Password	Password of the supervisor user. You must confirm this password on the following line.
Work Repository Type	Specify how the Work Repository will be used:
	■ Use **Development (D)** for creating a development repository. This type of repository allows management of design-time objects such as data models and projects (including interfaces, procedures, etc.) A development repository also includes the run-time objects (scenarios and sessions). This type of repository is suitable for development environments.
	■ Use **Execution (E)** for creating an execution repository: This type of repository only includes run-time objects (scenarios, schedules and sessions). It allows launching and monitoring of data integration jobs in Operator Navigator. Such a repository cannot contain any design-time artifacts. Designer Navigator cannot be used with it. An execution repository is suitable for production environments.
Work Repository Name	A unique name for the Work Repository (for example: `DEVWORKREP1`).
Work Repository Password	Provide a password for the Work Repository. If you provide a password, you must confirm the password on the following line.
Encryption Algorithm	Select the encryption algorithmm, either `AES-128` or `AES-256`. The default is `AES-128`.

A.7.2 Custom Variables for Oracle SOA Suite
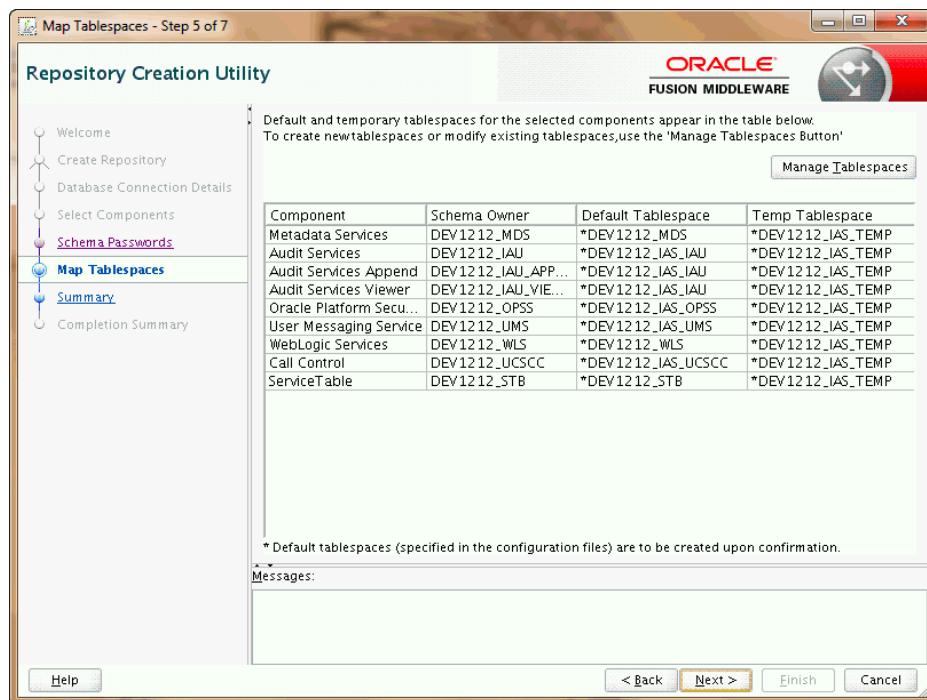
Below is the Custom Variables screen for Oracle SOA Suite:

The custom variables for Oracle SOA Suite are described in the following table:

Variable	Description
Database Profile	Specify the database profile you want to use for Oracle SOA Suite. This determines the size of the SOA database.
	The default value is SMALL.
Healthcare Integration	Specify whether or not you want to enable Healthcare Integration for Oracle SOA Suite.
	The default value is NO.

A.8 Map Tablespaces

This screen only appears if you selected the **Create** option on the Create Repository screen. The following topics are covered:
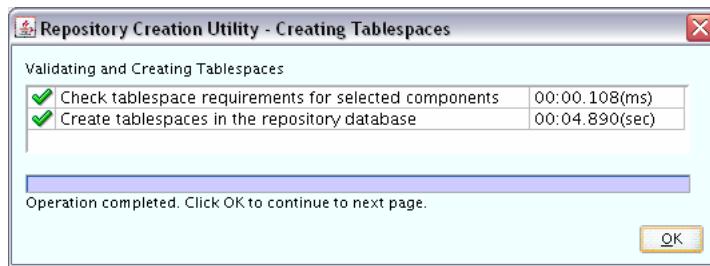
- Section A.8.1, "Default Tablespace Mappings"
- Section A.8.2, "Changing Default and Temporary Tablespaces"
- Section A.8.3, "Viewing and Changing Additional Tablespaces"
- Section A.8.4, "Managing Tablespaces and Datafiles"

Click **Next** when you are finished with your tablespace information. The following screen appears, asking you to confirm the creation of tablespaces for any new schemas.

> **Note:** RCU only creates tablespaces for those components associated with RCU.

Click **OK** to continue. The following screen appears, indicating the progress of the tablespace creation.

Click **Stop** to cancel tablespace creation. When the tablespaces are created, click **OK** to dismiss this window.

A.8.1 Default Tablespace Mappings

The default tablespace mapping for each component are shown in Appendix B.

In the "Default Tablespace" and "Temp Tablespace" columns, you can click on the tablespace cell to select from a list of available additional tablespace names.

A.8.2 Changing Default and Temporary Tablespaces

To change the default tablespace for a component, select the tablespace name in the Default Tablespace column, then select the tablespace name you want to use from the drop-down list. You can have your components use as many or as few tablespaces as desired to suit your configuration.

To change the temporary tablespace for a component, select the tablespace name in the Temp Tablespace column, then select the tablespace name you want to use from the drop-down list.

A.8.3 Viewing and Changing Additional Tablespaces

Some components have additional tablespaces associated with their schemas. If this is the case, the **Additional Tablespaces** button will appear on this screen. If none of the selected components have additional tablespaces, then this button will not appear.
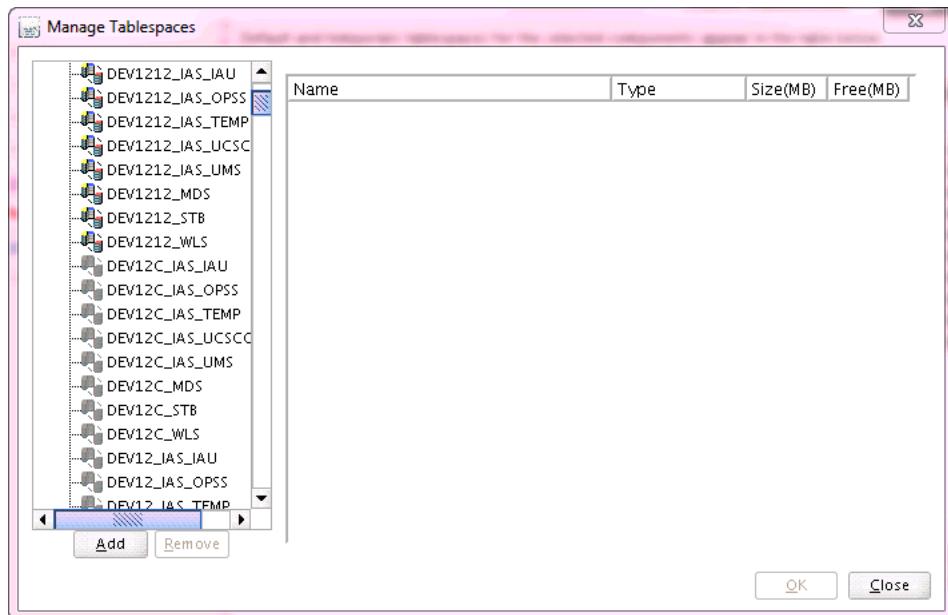
To view additional tablespaces associated with the selected components, click the **Additional Tablespaces** button. Only those components with additional tablespaces as defined in the configuration files will appear on this screen.

To change the tablespace you want to use for a component, click in the "Tablespace Name" column and select the tablespace you want to use from the drop-down list.

Click **OK** when you are finished.

A.8.4 Managing Tablespaces and Datafiles

To manage your tablespaces and datafiles, click the **Manage Tablespaces** button. You will see a screen similar to the following:

The following topics are covered in this section:

- Section A.8.4.1, "Adding, Modifying, and Removing Tablespaces"

- Section A.8.4.2, "Adding, Modifying, and Removing Datafiles"

A.8.4.1 Adding, Modifying, and Removing Tablespaces

Only tablespaces that will be created by RCU can be modified or removed. Tablespaces that existed before RCU was launched are visible on this screen but are grayed out and cannot be modified or removed.

Only tablespaces that are used by a component are created. You can specify a new tablespace here, but unless it is actually used by a component it will not be created.

To modify a tablespace, select the tablespace name on the left-hand portion of the screen, and edit the fields as described in the following table:

Field	Description
Name	Edit the tablespace name this field to change the name of your tablespace.
Type	Specify whether you want this tablespace to be a temporary tablespace or permanent tablespace.
Block Size (KB)	Specify the block size (in Kilobytes) to be used for data retrieval.
Storage Type	Select **Use Bigfile Tablespace** if you want to create a bigfile tablespace; this is typically used if you have single large files instead of multiple small files. Select **Use Automatic Segment Space Management** if you want to use bitmaps to manage the free space within segments.

To add a tablespace, click **Add** and specify the same details as above (for modifying a tablespace) for your new tablespace.

To remove a tablespace, select the tablespace name from the navigation tree, then click **Remove**. This tablespace will not get created.
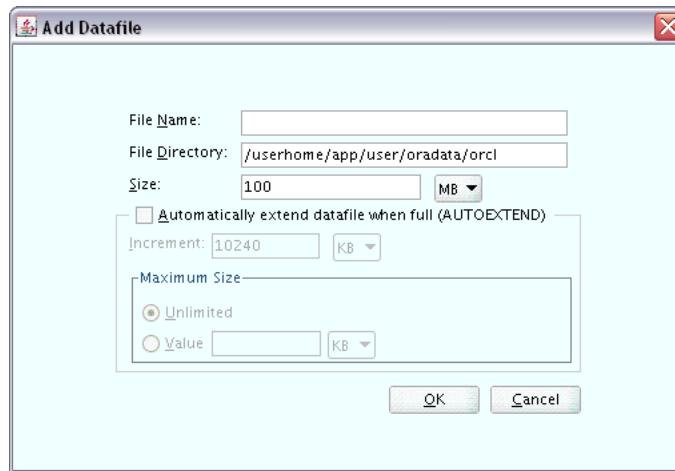
A.8.4.2 Adding, Modifying, and Removing Datafiles

In the Datafiles section, specify the datafiles that make up the selected tablespace.

To add a datafile, click the icon with the plus sign (+):

The Add Datafile screen appears:

Provide the information described in the following table:

Field	Description
File Name	Specify the name of the datafile. **NOTE:** Datafile names with a dash or hyphen (-) character are not permitted.
File Directory	Specify the location where this datafile will reside.
Size	Specify the initial size of the datafile. Use the drop-down list to specify the size in kilobytes (KB), megabytes (MB), or gigabytes (GB).
Automatically extend datafile when full (AUTOEXTEND)	Select **Automatically extend datafile when full (AUTOEXTEND)** if you want to automatically extend the size of your datafile when it becomes full. In the "Increment" field, specify the size by which your datafile should be increased each time it becomes full. Use the drop-down list to specify the size in kilobytes (KB), megabytes (MB), or gigabytes (GB). If you want to limit maximum size of the datafile, specify this value in the "Maximum Size" field.

Similarly, to modify or edit a datafile, select the icon next to the datafile name you want to edit, then click the icon with the pencil:
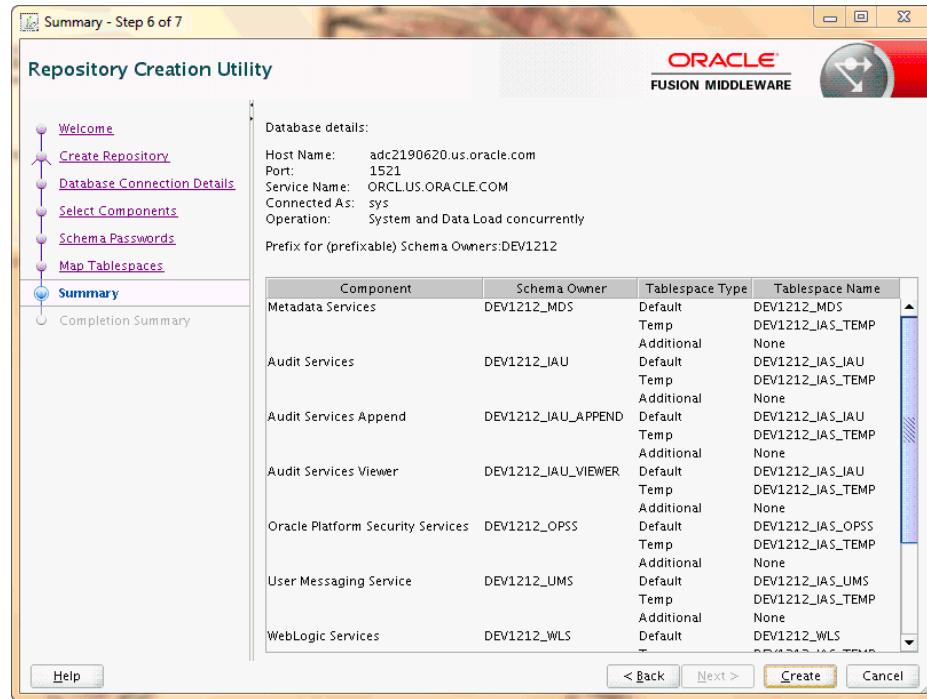
To delete a datafile, select the icon next to the datafile name you want to delete, then click the icon with the "X":
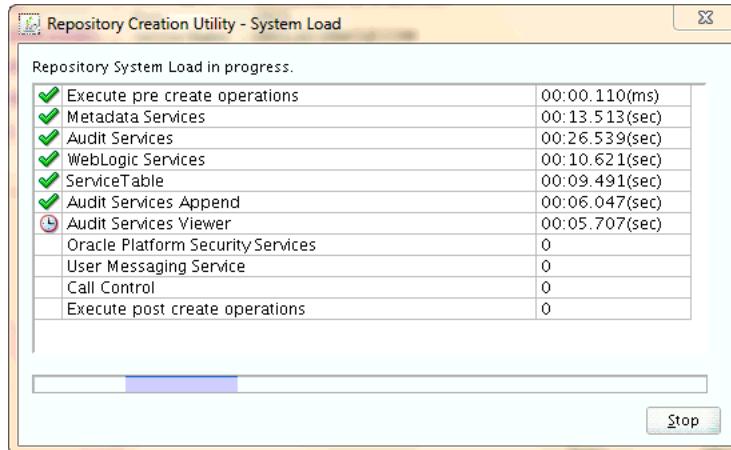
A.9 Summary (for Create Operation)

Below is the Summary screen if you selected **Create** on the Create Repository screen.

Review the information on this screen, and click **Create** to begin schema creation. The operations summarized on this page will be performed when you click **Create**.

While the schemas are being created, you will see the following progress screen:

Click **Stop** if you want to stop creating the schemas.
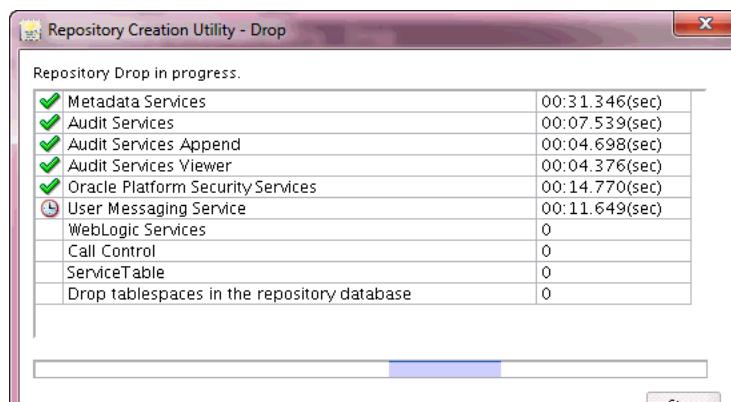
A.10 Summary (for Drop Operation)

Below is the Summary screen if you selected **Drop** on the Create Repository screen.

Review the information on this screen, and click **Drop** to begin the operations summarized on this page.
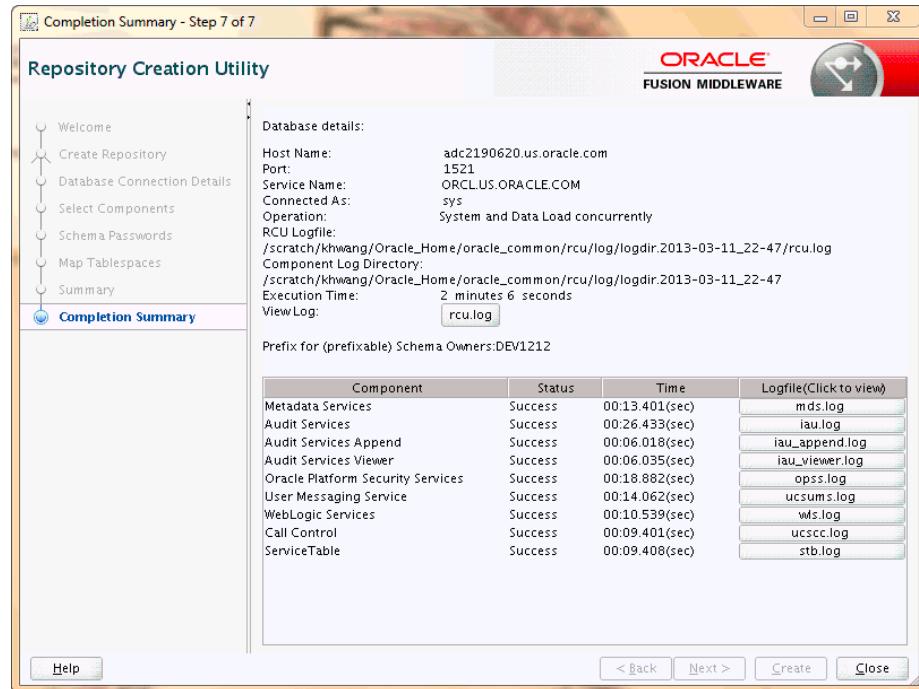
While the schema(s) are being dropped, you will see the following progress screen:

Click **Stop** if you want to cancel the operation.

A.11 Completion Summary (for Create Operation)

Below is the Completion Summary screen if you selected **Create** on the Create Repository screen.

This screen contains information about the log files that were created from this RCU operation. You can click on the name of a particular log file to view the contents of that file.
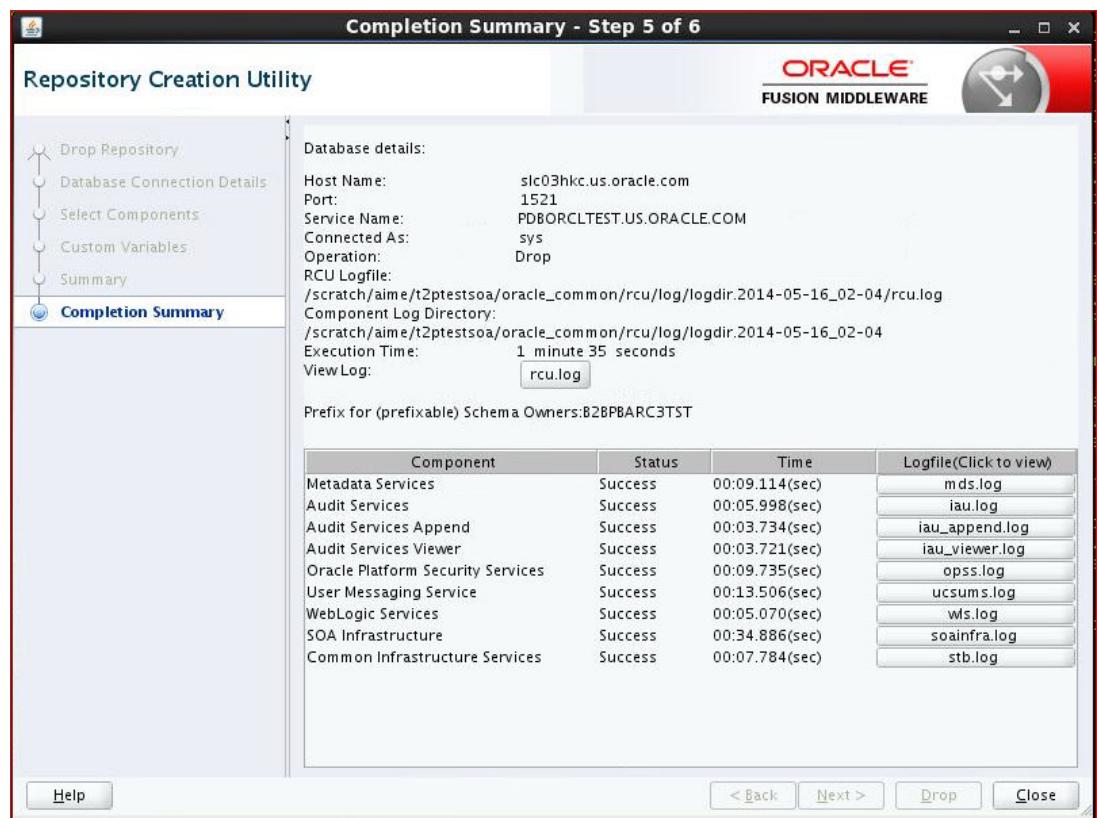
If there were any problems encountered during schema creation, you can troubleshoot the issue using the log files. For more information, refer to Section D.2.

If errors are encountered during a Create operation, or if a Create operation fails for any component, the **Cleanup for failed components** checkbox appears on this page and is selected by default. If selected, RCU will perform cleanup operations for the component that failed during the Create operation. If you choose not to select this checkbox, you can cleanup the failed component at a later time by performing a Drop operation for the failed component(s).

Review the information on this screen, then click **Close** to dismiss this screen.

A.12 Completion Summary (for Drop Operation)

Below is the Completion Summary screen if you selected **Drop** on the Create Repository screen.

Note the log file names for each component that are visible in the "Logfile" column.

The main RCU log (`rcu.log`) and component log files are written to the following directory on UNIX operating systems:

ORACLE_HOME/oralce_common/rcu/log/logdir.*date_timestamp*

On Windows operating systems:

ORACLE_HOME\oracle_common\rcu\log\logdir.*date_timestamp*

Click **rcu.log** to view the contents of the main RCU log in a separate window.

If there were any problems encountered during schema creation, you can troubleshoot the issue using the log files. For more information, see Appendix D, "Troubleshooting Repository Creation Utility".

Review the information on this screen, then click **Close** to dismiss this screen.

B

Understanding Repository Creation Utility Schemas, IDs, and Tablespaces

This appendix lists the available schemas that can be created using RCU, and also their component IDs and dependencies.

Table B–1 lists the schemas along with their component IDs, tablespace mappings, and dependencies.

The "Schema Owner" is the name of the schema that you will see in RCU and is also the name you must provide in the Oracle Fusion Middleware Configuration Wizard on the Configure JDBC Component Schema screen, prepended by the custom prefix.

The "Component ID" is the value you must specify with the `-component` parameter when you are creating or dropping schemas using the command line.

> **Note:** Not all schemas are supported on all database types. For more information, refer to "Verifying Requirements for Repository Creation Utility" in the *Oracle Fusion Middleware System Requirements and Specifications* document.

Table B–1 Schema Component IDs, Tablespace Mappings, and Dependencies

Component	Schema Owner	Component ID	Default Tablespace	Temp Tablespace	Dependencies
Metadata Services	`MDS`	`MDS`	`MDS`	`IAS_TEMP`	None
Audit Services	`IAU`	`IAU`	`IAS_IAU`	`IAS_TEMP`	Audit Services Append (`IAU_APPEND`)
					Audit Services Viewer (`IAU_VIEWER`)
Audit Services Append	`IAU_APPEND`	`IAU_APPEND`	`IAS_IAU`	`IAS_TEMP`	None
Audit Services Viewer	`IAU_VIEWER`	`IAU_VIEWER`	`IAS_IAU`	`IAS_TEMP`	None
Oracle Platform Security Services	`OPSS`	`OPSS`	`IAS_OPSS`	`IAS_TEMP`	Audit Services (`IAU`)
					Audit Services Append (`IAU_APPEND`)
					Audit Services Viewer (`IAU_VIEWER`)

Component	Schema Owner	Component ID	Default Tablespace	Temp Tablespace	Dependencies
User Messaging Service	UMS	UCSUMS	IAS_UMS	IAS_TEMP	Metadata Services (MDS)
WebLogic Services	WLS	WLS	WLS	IAS_TEMP	None
Call Control	UCSCC	UCSCC	IAS_UCSCC	IAS_TEMP	Metadata Services (MDS)
Oracle Enterprise Scheduler	ESS	ESS	ESS	IAS_TEMP	Metadata Services (MDS)
					Audit Services (IAU)
					Audit Services Append (IAU_ APPEND)
					Audit Services Viewer (IAU_ VIEWER)
					Oracle Platform Security Services (OPSS)
Common Infrastructure Services (formerly Service Table)[1]	STB	STB	STB	IAS_TEMP	None

Table B–1   (Cont.) Schema Component IDs, Tablespace Mappings, and Dependencies

Component	Schema Owner	Component ID	Default Tablespace	Temp Tablespace	Dependencies
Master and Work Repository	`ODI`	`ODI`			
SOA Infrastructure	`SOAINFRA`				Metadata Services (`MDS`)
					Audit Services (`IAU`)
					Audit Services Append (`IAU_APPEND`)
					Audit Services Viewer (`IAU_VIEWER`)
					Oracle Platform Security Services (`OPSS`)
					User Messaging Service (`UMS`)
					WebLogic Services (`WLS`)
Managed File Transfer	`MFT`		`MFT`	`IAS_TEMP`	Metadata Services (`MDS`)
					Audit Services (`IAU`)
					Audit Services Append (`IAU_APPEND`)
					Audit Services Viewer (`IAU_VIEWER`)
					Oracle Platform Security Services (`OPSS`)
					User Messaging Service (`UMS`)
					Oracle Enterprise Scheduler (`ESS`)

[1]   For more information about the Service Table schema, see Section 1.2.4.

C

Extending Repository Creation Utility to Configure Custom Application Repositories

This chapter describes the configuration XML files that can be used to generate custom schema creation and deletion scripts.

RCU provides an XML-based framework for component owners to plug in your schema creation and deletion scripts into RCU. This chapter provides some details of the configuration XML files and script-writing guidelines that are used to integrate your components with RCU.

The following topics are covered in this appendix:

- Section C.1, "RCU Integration Options"
- Section C.2, "RCU Configuration Files"
- Section C.3, "RCU Script Writing Guidelines"

C.1 RCU Integration Options

RCU provides the following options for integrating component scripts:

- RCU JDBC Engine Compliant SQL*Plus Scripts
- Pure JDBC Scripts
- SQL*Plus Scripts
- External Processes
- Java Code Using JavaAction

RCU JDBC Engine Compliant SQL*Plus Scripts is the recommended option for integrating component scripts. SQL*Plus and External Processes are only intended for integrating Legacy/Classic components such as Oracle Portal 10*g* or Identity Management. Components that have a dependency on SQL*Plus scripts cannot be loaded with RCU when running from the installed Oracle home. They can only be used when running RCU from CD.

C.1.1 RCU JDBC Engine Compliant SQL*Plus Scripts

The RCU JDBC Engine emulates a set of SQL*Plus features over JDBC. This set is broad enough to cover the requirements of schema creation. Your component teams can integrate existing SQL*Plus scripts with a few minor changes.

The RCU JDBC Engine parses the SQL*Plus script to get individual statements and then runs each statement over JDBC. Command line arguments to scripts and substitution using DEFINE variables are supported. Script can be nested (for example,

one script can call other scripts). Component teams can specify list of expected errors and fatal errors to RCU through configuration files and RCU would interpret these when running the scripts.

These scripts are easy to maintain and use as they can be run in SQL*Plus in development environment. However, Oracle recommends that the RCU JDBCEngine tool is also used in your development environment to ensure that these scripts run properly when integrated with RCU.

C.1.2 Pure JDBC Scripts

This option is recommended for non-Oracle databases (for Oracle databases, RCU JDBC Engine Compliant SQL*Plus scripts should be used). Contents of the script file should be a valid PL/SQL block, which can be called with `Connection.prepareCall()` or `Connection.createStatement()`. Standard JDBC Bind variables with '?' convention are supported.

Some disadvantages of this option are:

- No nested scripts, which can mean a larger number of scripts.

- May require a more significant re-work for component teams to re-write the scripts in this format.

- Difficult to maintain as every DDL statement has to be wrapped with in EXECUTE IMMEDIATE.

- Cannot be run using SQL*Plus in development environment.

- Less useful error support since the whole block would fail in case of any errors.

Below is an example:

```
<Action TYPE="JDBC" PERCENT_PROGRESS="20">
   <ValidIf DBTYPE="ORACLE" />
   <Command TYPE="INLINE">DROP USER %SCHEMA_USER% CASCADE</Command>
</Action>
```

And a second example:

```
<Action TYPE="Java" PERCENT_PROGRESS="100">
   <Command TYPE="METHOD">
      oracle.ias.version.SchemaVersionUtil:utilSetComponentValid
   </Command>

   <Parameters>
      <Parameter TYPE="String">MDS</Parameter>
    </Parameters>
</Action>
```

C.1.3 SQL*Plus Scripts

This option is mainly for the consumption of legacy components that need to be loaded from RCU.

Example:

```
<Action TYPE="SQLPlus" PERCENT_PROGRESS="100">
   <Command TYPE="SCRIPT">%SCRIPT_HOME%/oid/scripts/seedldap.sql</Command>
   <IgnorableErrors>
      <Error Type="ORA-01918">user name does not exist</Error>
   </IgnorableErrors>
</Action>
```

And a second example:

```
<Action TYPE="JDBCSqlScript" PERCENT_PROGRESS="20">
   <ValidIf DBTYPE="ORACLE" />
   <Command TYPE="SCRIPT">%SCRIPT_HOME%/mds/sql/mds_user.sql</Command>
   <Parameters>
      <Parameter TYPE="CmdLine">%SCHEMA_USER%</Parameter>
      <Parameter TYPE="CmdLine">%SCHEMA_PASSWORD%</Parameter>
      <Parameter TYPE="CmdLine">%DEFAULT_TABLESPACE%</Parameter>
      <Parameter TYPE="CmdLine">%TEMPORARY_TABLESPACE%</Parameter>
   </Parameters>
</Action>
```

C.1.4 External Processes

This option is provided only for those components that have their own configuration tool for schema creation, like OPCA (Oracle Portal 10*g*). This is not a recommended option for any new component, as this option cannot make use of RCU error handling framework.

Example:

```
<Action TYPE="HostCmd">
<Command TYPE="SCRIPT">%RCU_
HOME%/rcu/integration/cdb/config/bin/configure</Command>
<Parameters>
   <Parameter TYPE="ProcessInput">%JDBC_CONNECT_STRING%</Parameter>
   <Parameter TYPE="ProcessInput">%DBADMIN_USER%</Parameter>
   <Parameter TYPE="ProcessInput">%DBADMIN_PASSWORD%</Parameter>
   <Parameter TYPE="ProcessInput">%PREFIX_NAME%</Parameter>
   <Parameter TYPE="ProcessInput">%SCHEMA_USER%</Parameter>
   <Parameter TYPE="ProcessInput">%SCHEMA_PASSWORD%</Parameter>
   <Parameter TYPE="ProcessInput">%DEFAULT_TABLESPACE%</Parameter>
   <Parameter TYPE="ProcessInput">%TEMPORARY_TABLESPACE%</Parameter>
</Parameters>
</Action>
```

C.1.5 Java Code Using JavaAction

This option is provided to components that have Java code, which can accept a JDBC connection and execute SQL statements. This is generally used when huge amounts of data has to be seeded or LOBs need to be created.

Example:

```
<Action TYPE="Java">
   <Command TYPE="METHOD">
      oracle.ias.version.SchemaVersionUtil:utilCreateRegistryEntry
   </Command>

   <Parameters>
      <Parameter TYPE="Connection"></Parameter>
      <Parameter TYPE="String">%SCHEMA_USER%</Parameter>
   </Parameters>
</Action>
```

A second example:

```
<Action TYPE="Java">
   <Command TYPE="METHOD">oracle.webdb.config.PortalConfigAssistant:main</Command>
   <Parameters>
```

```
        <Parameter TYPE="StringArray">-mode PORTAL -s %SCHEMA_USER% -p %DBADMIN_
PASSWORD% -c %DB_HOSTNAME%:%DB_PORTNUMBER%:%DB_SERVICE% -silent -verbose -owa -u
%DEFAULT_TABLESPACE% -t %TEMPORARY_TABLESPACE% -d %SCHEMA_USER%_DOC -l %SCHEMA_
USER%_LOG -in %SCHEMA_USER%_IDX -demo -report -voh %RCU_HOME% -log %RCU_LOG_
LOCATION% -oh %SQLPLUS_HOME% -mrc %PREFIX_NAME% -rcu </Parameter>
    </Parameters>
</Action>
```

C.2 RCU Configuration Files

RCU provides the following configuration files types for component integration:

- Section C.2.1, "XML DTDs Defined by RCU"

- Section C.2.2, "Component Repository Configuration File"

- Section C.2.3, "Component List Configuration File"

- Section C.2.4, "Soft-Prerequisite Support"

- Section C.2.5, "Default Tablespaces Configuration File"

C.2.1 XML DTDs Defined by RCU

This section describes the XML DTDs defined by RCU:

- Component Descriptor Configuration File

- Repository Configuration File

- Master List of Supported Components

- Storage Attributes Configuration File

C.2.1.1 Component Descriptor Configuration File

Each component owner would provide a configuration file adhering to following DTD, which lists the pre-requisites and actions:

The Component Descriptor configuration file is called `ComponentInfo.dtd` and is located in the *ORACLE_HOME*/oracle_common/rcu/config (on UNIX operating systems) or *ORACLE_HOME*\oracle_common\rcu\config (on Windows operating systems) directory:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!ENTITY % commonDTD SYSTEM "RCUCommon.dtd">
%commonDTD;
<!ELEMENT ComponentInfo (Display, PrefixSettings, Component*,
PrerequisiteDescriptor*, ExecutionDescriptor?, FatalErrors?, IgnorableErrors?)>
<!ATTLIST ComponentInfo
        VERSION CDATA #REQUIRED
        TYPE CDATA #REQUIRED
        RESOURCE_BUNDLE_PACKAGE CDATA #IMPLIED>
<!ELEMENT PrefixSettings (DetectQuery*)>
<!ATTLIST PrefixSettings
        USE_SCHEMA_PREFIX (TRUE|FALSE) "TRUE"
        USE_TABLESPACE_PREFIX (TRUE|FALSE) "TRUE">
<!ELEMENT Component (ValidIfSet?, ValidIf?, Display, RepositoryConfigFile?,
DetectQuery*, SchemaVersion?, SchemaUser?, AdditionalSchemaUser*, Dependents?,
DatabaseName?, Tablespaces?, CustomVariables?)>
<!ATTLIST Component
        ID CDATA #REQUIRED
        PROGRESS_UNITS CDATA #IMPLIED
```

```
              IS_GROUPING_COMPONENT  (TRUE│FALSE) "FALSE"
              DEFAULT_SELECTED (TRUE│FALSE) "FALSE"
              CHILD_OF CDATA #IMPLIED
              MANDATORY_SELECTED (TRUE│FALSE) "FALSE">
<!ELEMENT Display (#PCDATA)>
<!ATTLIST Display
              NLS_ID CDATA #IMPLIED>
<!ELEMENT RepositoryConfigFile (#PCDATA)>
<!ELEMENT DetectQuery (#PCDATA)>
<!ATTLIST DetectQuery
              OPERATION (CREATE│DROP) 'CREATE'
              TYPE (ORACLE│SQLSERVER│IBMDB2│EBR│MYSQL) 'ORACLE'>
<!ELEMENT SchemaVersion (#PCDATA)>
<!ELEMENT SchemaUser (#PCDATA)>
<!ATTLIST SchemaUser
              USER_EDITABLE (TRUE│FALSE) "TRUE"
              PREFIXABLE (TRUE│FALSE) "TRUE"
              IS_CREATED  (TRUE│FALSE) "TRUE"
              MAX_LENGTH CDATA "30">
<!ELEMENT AdditionalSchemaUser (#PCDATA)>
<!ATTLIST AdditionalSchemaUser
              STARTS_WITH_SCHEMA_USER (TRUE│FALSE) "TRUE" >
<!ELEMENT Dependents (Dependent*)>
<!ELEMENT Dependent (#PCDATA)>
<!ATTLIST Dependent
              COMPONENT_ID CDATA #REQUIRED
              ALT_COMPONENT_ID CDATA #IMPLIED>
<!ELEMENT DatabaseName (#PCDATA)>
<!ELEMENT Tablespaces (Tablespace*)>
<!ATTLIST Tablespace TYPE (DEFAULT_TABLESPACE│TEMPORARY_TABLESPACE│ADDITIONAL_
TABLESPACE1│ADDITIONAL_TABLESPACE2│ADDITIONAL_TABLESPACE3│ADDITIONAL_
TABLESPACE4│ADDITIONAL_TABLESPACE5│ADDITIONAL_TABLESPACE6│ADDITIONAL_
TABLESPACE7│ADDITIONAL_TABLESPACE8│ADDITIONAL_TABLESPACE9│ADDITIONAL_
TABLESPACE10│ADDITIONAL_TABLESPACE11│ADDITIONAL_TABLESPACE12│ADDITIONAL_
TABLESPACE13│ADDITIONAL_TABLESPACE14│ADDITIONAL_TABLESPACE15│ADDITIONAL_
TABLESPACE16│ADDITIONAL_TABLESPACE17│ADDITIONAL_TABLESPACE18│ADDITIONAL_
TABLESPACE19│ADDITIONAL_TABLESPACE20│ADDITIONAL_TABLESPACE21│ADDITIONAL_
TABLESPACE22│ADDITIONAL_TABLESPACE23│ADDITIONAL_TABLESPACE24│ADDITIONAL_
TABLESPACE25│ADDITIONAL_TABLESPACE26│ADDITIONAL_TABLESPACE27│ADDITIONAL_
TABLESPACE28│ADDITIONAL_TABLESPACE29│ADDITIONAL_TABLESPACE30│ADDITIONAL_
TABLESPACE31) "DEFAULT_TABLESPACE">
<!ELEMENT Tablespace (Prompt, TablespaceName)>
<!ELEMENT Prompt (#PCDATA)>
<!ATTLIST Prompt NLS_ID CDATA #IMPLIED>
<!ELEMENT TablespaceName (#PCDATA)>
<!ELEMENT CustomVariables (Variable*)>
<!ATTLIST Variable
              TYPE (STRING│NUMBER│PASSWORD│EXISTING_PASSWORD│FILE) "STRING"
              MAX_LENGTH CDATA "30"
              MIN_LENGTH CDATA "0"
              OPERATION (CREATE│DROP│BOTH) "BOTH"
              DEFAULT_VALUE CDATA "">
<!ELEMENT Variable (Name,Display)>
<!ELEMENT Name (#PCDATA)>
```

C.2.1.2 Repository Configuration File

The Repository configuration file is called `RepositoryConfig.dtd` and is located in the *ORACLE_HOME*/oracle_common/rcu/config (on UNIX operating systems) or *ORACLE_HOME*\oracle_common\rcu\config (on Windows operating systems) directory:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!ENTITY % commonDTD SYSTEM "RCUCommon.dtd">
%commonDTD;
<!ELEMENT RepositoryConfig (PrerequisiteDescriptor*, ExecutionDescriptor,
DeleteDescriptor?)>
<!ATTLIST RepositoryConfig
        COMP_ID CDATA #REQUIRED>
<!ELEMENT DeleteDescriptor (Action*)>
```

C.2.1.3 Master List of Supported Components

RCU maintains a master list of supported components, which contains entries for each supported component. Every time a new component is added, the master list of supported components is updated with the reference of the XML integration file provided by component owner.

This configuration file is called `RCUCommon.dtd` and is located in the *ORACLE_HOME*/oracle_common/rcu/config (on UNIX operating systems) or *ORACLE_HOME*\oracle_common\rcu\config (on Windows operating systems) directory:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!ELEMENT PrerequisiteDescriptor (DBPrerequisiteSet*, DBPrerequisite*)>
<!ATTLIST PrerequisiteDescriptor
        TYPE (CREATE|DROP|REGISTER|DEREGISTER|SYSTEMLOAD|DATALOAD) 'CREATE'>
<!ELEMENT DBPrerequisiteSet (ValidIfSet?, ValidIf?, PrereqSetErrorMsg?,
DBPrerequisite*)>
<!ATTLIST DBPrerequisiteSet
         OPERATOR (OR|AND) "OR"
         SOFT (TRUE|FALSE) "FALSE">
<!ELEMENT DBPrerequisite (ValidIfSet?, ValidIf?, PrereqIdentifier, PrereqValue,
PrereqErrorMsg?)>
<!ATTLIST DBPrerequisite
PREREQ_TYPE
(InitParameter|DBOption|Java|DBComponent|DBVersion|DBObject|CustomSQL|TablespaceFr
eeMB) "CustomSQL"
        DATA_TYPE (STRING|NUMBER|VERSION) "STRING"
        COMPARE_OPERATOR (EQ|GT|LT|NE|GE|LE|COMPARE_VERSION) "EQ"
        SOFT (TRUE|FALSE) "FALSE">

<!ELEMENT PrereqIdentifier (#PCDATA)>
<!ELEMENT PrereqValue (#PCDATA)>
<!ELEMENT PrereqSetErrorMsg (#PCDATA)>
<!ATTLIST PrereqSetErrorMsg
        NLS_ID CDATA #IMPLIED>
<!ELEMENT PrereqErrorMsg (#PCDATA)>
<!ATTLIST PrereqErrorMsg
        NLS_ID CDATA #IMPLIED>
<!ATTLIST PrereqValue
        UNIT (KB|MB|NoUnit) 'NoUnit'>
<!ELEMENT ExecutionDescriptor (Action*)>
<!ATTLIST ExecutionDescriptor
        TYPE (Load|PreLoad|PostLoad|DataLoad|PostDataLoad) "Load">
<!ELEMENT Action (ValidIfSet?, ValidIf?, Command, Parameters?, FatalErrors?,
IgnorableErrors?)>
<!ATTLIST Action
        TYPE (JDBCSqlScript|JDBC|SQLPlus|HostCmd|Java) "JDBCSqlScript"
        DB_VERSION CDATA #IMPLIED
        PERCENT_PROGRESS CDATA #IMPLIED
        CONNECT_AS_OWNER (TRUE|FALSE) "FALSE"
        CONNECT_AS_REGISTRYOWNER (TRUE|FALSE) "FALSE"
        RESET_SESSION (TRUE|FALSE) "FALSE">
```

```
<!ELEMENT Command (#PCDATA)>
<!ATTLIST Command
        TYPE (SCRIPT|INLINE|METHOD) "SCRIPT">
<!ELEMENT Parameters (Parameter*)>
<!ELEMENT Parameter (#PCDATA)>
<!ATTLIST Parameter
        TYPE
(BindVar|CmdLine|ProcessInput|EnvVar|Connection|int|String|StringArray|boolean)
"CmdLine">
<!ELEMENT FatalErrors (Error*)>
<!ELEMENT IgnorableErrors (Error*)>
<!ELEMENT Error (#PCDATA)>
<!ATTLIST Error
        Type CDATA #REQUIRED>
<!ELEMENT ValidIfSet (ValidIf*)>
<!ATTLIST ValidIfSet
        DBTYPE CDATA #IMPLIED
        DBVERSION CDATA #IMPLIED
        OSNAME CDATA #IMPLIED
        OPERATION_TYPE CDATA "ALL"
        OPERATOR (OR|AND) "OR">
<!ELEMENT ValidIf (CustomQueryFilter?)>
<!ATTLIST ValidIf
 DBTYPE CDATA #IMPLIED
        DBVERSION CDATA #IMPLIED
        OSNAME CDATA #IMPLIED
        OPERATION_TYPE CDATA "ALL">
<!ELEMENT CustomQueryFilter (#PCDATA)>
<!ATTLIST CustomQueryFilter
        DATA_TYPE (STRING|NUMBER) "STRING"
        COMPARE_OPERATOR (EQ|GT|LT|NE|GE|LE|COMPARE_VERSION) "EQ"
        VALUE CDATA #REQUIRED >
```

C.2.1.4 Storage Attributes Configuration File

RCU maintains the list of tablespaces/datafiles and their attributes to be created. This way the tablespaces and datafiles attributes can be modified externally.

The Storage Attributes configuration file is called Storage.dtd and is located in the *ORACLE_HOME*/oracle_common/rcu/config (on UNIX operating systems) or *ORACLE_HOME*\oracle_common\rcu\config (on Windows operating systems) directory:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!ELEMENT StorageAttributes ( ComponentStorageFiles?, TablespaceAttributes*)>
<!ELEMENT ComponentStorageFiles (StorageFile*)>
<!ATTLIST StorageFile
        ID CDATA #REQUIRED>
<!ELEMENT StorageFile (#PCDATA)>
<!ELEMENT TablespaceAttributes (ValidIfSet?,
ValidIf?,Type?,DefaultTemp?,BlockSize?,ExtentSize?,PageSize?,AutoResize?,IncreaseS
ize?,MaxSize?,Bigfile?,AutoSegmentSpaceManagement?, DatafilesList)>
<!ATTLIST TablespaceAttributes
        NAME CDATA #REQUIRED
    OMF (TRUE|FALSE) "FALSE">
<!ELEMENT ValidIfSet (ValidIf*)>
<!ATTLIST ValidIfSet
    DBTYPE CDATA #IMPLIED
    DBVERSION CDATA #IMPLIED
    OSNAME CDATA #IMPLIED
    OPERATOR (OR|AND) "OR">
<!ELEMENT ValidIf (CustomQueryFilter?)>
```

```
            <!ATTLIST ValidIf
                DBTYPE CDATA #IMPLIED
                DBVERSION CDATA #IMPLIED
                OSNAME CDATA #IMPLIED >
        <!ELEMENT Type (#PCDATA)>
        <!ELEMENT DefaultTemp (#PCDATA)>
        <!ELEMENT BlockSize (#PCDATA)>
        <!ELEMENT ExtentSize (#PCDATA)>
        <!ELEMENT PageSize (#PCDATA)>
        <!ATTLIST PageSize
                UNIT (KB|NoUnit) 'KB'>
        <!ELEMENT AutoResize (#PCDATA)>
        <!ELEMENT IncreaseSize (#PCDATA)>
        <!ATTLIST IncreaseSize
                UNIT (KB|MB|GB) 'MB'>
        <!ELEMENT MaxSize (#PCDATA)>
        <!ATTLIST MaxSize
                UNIT (KB|MB|GB) 'MB'>
        <!ELEMENT Bigfile (#PCDATA)>
        <!ELEMENT AutoSegmentSpaceManagement (#PCDATA)>
        <!ELEMENT DatafilesList (DatafileAttributes+)>
        <!ELEMENT DatafileAttributes (Size, Reuse?, AutoExtend?, Increment?, Maxsize?)>
        <!ATTLIST DatafileAttributes
         ID CDATA #REQUIRED>
        <!ELEMENT Size (#PCDATA)>
        <!ATTLIST Size
                UNIT (KB|MB|GB) 'MB'>
        <!ELEMENT Reuse (#PCDATA)>
        <!ELEMENT AutoExtend (#PCDATA)>
        <!ELEMENT Increment (#PCDATA)>
        <!ATTLIST Increment
                UNIT (KB|MB|GB) 'KB'>
        <!ELEMENT Maxsize (#PCDATA)>
        <!ATTLIST Maxsize
                UNIT (KB|MB|GB) 'MB'>
```

C.2.2 Component Repository Configuration File

A Component Repository Configuration File (`component.xml`) lists the pre-requisites and the list of scripts or actions that need to be performed to load or drop a schema. This file is provided and maintained by component owners. This configuration file is referenced from the Component List Configuration File (`ComponentInfo.xml`).

Each `component.xml` file can be found in the `ORACLE_HOME`\oracle_ common/rcu/integration*component*/*component*.xml (on UNIX operating systems) or `ORACLE_HOME`\oracle_common\rcu\integration*component**component*.xml (on Windows operating systems) file.

Component owners can use a set of predefined RCU parameters which will be substituted at runtime by RCU based on user input. Here is the list of predefined parameters:

Table C–1   Predefined RCU Parameters

RCU Parameter	Description
%ORACLE_HOME%	Location of the Oracle home directory.
%SCRIPT_HOME%	Location where scripts are located. It may be same as *ORACLE_HOME*.
%SCHEMA_USER%	Database schema name (owner) entered by the user in RCU.

Table C–1   (Cont.) Predefined RCU Parameters

RCU Parameter	Description
%SCHEMA_PASSWORD%	Database schema password entered by the user in RCU.
%ADDITIONAL_SCHEMA_USER%	Additional schema users as defined in the ComponentInfo.xml file
%ADDITIONAL_SCHEMA_PASSWORD<n>%	Password for the additional schema users.
%DEFAULT_TABLESPACE%	Default tablespace assigned to the component by the user.
%TEMPORARY_TABLESPACE%	Temporary tablespace assigned to the component by the user.
%ADDITIONAL_TABLESPACE<n>%	Additional tablespace assigned to the component by the user. Up to three additional tablespaces are supported.
%DEFAULT_PERMANENT_TABLESPACE%	Default permanent tablespace in the database (for example, USERS or SYSTEM) is none is set.
%DEFAULT_TEMP_TABLESPACE%	Default temporary tablespace in the database (for example, TEMP in Oracle shipped databases or SYSTEM) if none is set.
%DATAFILE_LOCATION%	Default location where the tablespace/datafile will be created.
%JDBC_CONNECT_STRING%	JDBC connect string.
%PREFIX_NAME%	User-specified prefix for schema and tablespace names.
%CONNECTION%	Already-connected java.sql.Connection object to be passed into JavaAction.
%DBADMIN_USER%	Database admin user that is provided on the Database Connection Details.
%DBADMIN_PASSWORD%	Database admin user password that is provided on the Database Connection Details.
%DBADMIN_ROLE%	Database admin user role that is provided on the Database Connection Details.
%DB_HOSTNAME%	Database hostname that is provided on the Database Connection Details.
%DB_SERVICE%	Database service name.
%DB_PORTNUMBER%	Database port number that is provided on the Database Connection Details.
%RCU_HOME%	Directory where RCU is installed.
%SQLPLUS_HOME%	*ORACLE_HOME* where SQL*Plus is located.
%RCU_LOG_LOCATION%	Location of the directory where RCU log files are created.
%DATABASE_NMAE%	Database name (for SQLServer database).

Below is a sample Component Repository Configuration file for OPSS (opss.xml), which lists the series of prerequisites and actions:

```
<?xml version="1.0" encoding="UTF-8" ?>

<!DOCTYPE RepositoryConfig SYSTEM "RepositoryConfig.dtd">
<RepositoryConfig COMP_ID="OPSS">

    <!-- Prerequisites for OPSS -->
    <PrerequisiteDescriptor>
         <!-- ORACLE Prerequisites -->
         <DBPrerequisite PREREQ_TYPE="TablespaceFreeMB" DATA_TYPE="NUMBER"
                      COMPARE_OPERATOR="GT">
            <ValidIf DBTYPE="ORACLE" />
```

```
                    <PrereqIdentifier>%DEFAULT_TABLESPACE%</PrereqIdentifier>
                    <PrereqValue>50</PrereqValue>
            </DBPrerequisite>
            <DBPrerequisite PREREQ_TYPE="TablespaceFreeMB" DATA_TYPE="NUMBER"
                            COMPARE_OPERATOR="GT">
                    <ValidIf DBTYPE="ORACLE" />
                    <PrereqIdentifier>%TEMPORARY_TABLESPACE%</PrereqIdentifier>
                    <PrereqValue>50</PrereqValue>
            </DBPrerequisite>

            <!-- ORACLE EBR Prerequisites -->
            <DBPrerequisite PREREQ_TYPE="TablespaceFreeMB" DATA_TYPE="NUMBER"
                            COMPARE_OPERATOR="GT">
                    <ValidIf DBTYPE="EBR" />
                    <PrereqIdentifier>%DEFAULT_TABLESPACE%</PrereqIdentifier>
                    <PrereqValue>50</PrereqValue>
            </DBPrerequisite>
            <DBPrerequisite PREREQ_TYPE="TablespaceFreeMB" DATA_TYPE="NUMBER"
                            COMPARE_OPERATOR="GT">
                    <ValidIf DBTYPE="EBR" />
                    <PrereqIdentifier>%TEMPORARY_TABLESPACE%</PrereqIdentifier>
                    <PrereqValue>50</PrereqValue>
            </DBPrerequisite>

        <!-- SQLServer Prerequisites -->

        <!-- DB2 Prerequisites -->

    </PrerequisiteDescriptor>

 <PrerequisiteDescriptor TYPE="DROP">
        <!-- ORACLE Prerequisites -->
        <DBPrerequisite PREREQ_TYPE="CustomSQL" DATA_TYPE="NUMBER"
                COMPARE_OPERATOR="EQ">
                <ValidIf DBTYPE="ORACLE" />
                <PrereqIdentifier>
                        select count(*) from v$session where username='%SCHEMA_
USER%'
                </PrereqIdentifier>
                <PrereqValue>0</PrereqValue>
                <PrereqErrorMsg>
                        The schema owner %SCHEMA_USER% is connected to the
                        database. Please disconnect and try again.
                </PrereqErrorMsg>
        </DBPrerequisite>

        <!-- ORACLE EBR Prerequisites -->
        <DBPrerequisite PREREQ_TYPE="CustomSQL" DATA_TYPE="NUMBER" COMPARE_
OPERATOR="EQ">
                <ValidIf DBTYPE="EBR" />
                <PrereqIdentifier>select count(*) from v$session where
username='%SCHEMA_USER%'</PrereqIdentifier>
                <PrereqValue>0</PrereqValue>
                <PrereqErrorMsg NLS_ID="WC_USER_CONNECTED">The schema owner
%SCHEMA_USER% is connected to the database. Please disconnect and try
again.</PrereqErrorMsg>
        </DBPrerequisite>

        <!-- SQLServer Prerequisites -->
```

```
        <!-- DB2 Prerequisites -->

    </PrerequisiteDescriptor>


    <!-- Creating the OPSS Schema and Setting it to valid in the Registry -->
    <ExecutionDescriptor>
        <Action TYPE="Java" PERCENT_PROGRESS="20">
            <Command
TYPE="METHOD">oracle.ias.version.SchemaVersionUtil:utilCreateRegistryEntry</Comman
d>
            <Parameters>
                <Parameter TYPE="Connection"></Parameter>
                <Parameter TYPE="String">OPSS</Parameter>
                <Parameter TYPE="String">Oracle Platform Security Services</Parameter>
                <Parameter TYPE="String">%PREFIX_NAME%</Parameter>
                <Parameter TYPE="String">OPSS</Parameter>
                <Parameter TYPE="String">OPSS</Parameter>
                <Parameter TYPE="String">%SCHEMA_USER%</Parameter>
                <Parameter TYPE="String">12.1.2.0.0</Parameter>
                <Parameter TYPE="String">LOADING</Parameter>
            </Parameters>
        </Action>
 <Action TYPE="JDBCSqlScript" PERCENT_PROGRESS="40">
            <ValidIf DBTYPE="ORACLE" />
            <Command TYPE="SCRIPT">%SCRIPT_HOME%/opss/scripts/opss_user.sql</Command>
            <Parameters>
                <Parameter TYPE="CmdLine">%SCHEMA_USER%</Parameter>
                <Parameter TYPE="CmdLine">%SCHEMA_PASSWORD%</Parameter>
                <Parameter TYPE="CmdLine">%DEFAULT_TABLESPACE%</Parameter>
                <Parameter TYPE="CmdLine">%TEMPORARY_TABLESPACE%</Parameter>
            </Parameters>
            <IgnorableErrors>
                <Error Type="ORA-01918">user name does not exist</Error>
            </IgnorableErrors>
        </Action>
        <Action TYPE="JDBCSqlScript" PERCENT_PROGRESS="20">
            <ValidIf DBTYPE="EBR" />
            <Command TYPE="SCRIPT">%SCRIPT_HOME%/opss/scripts/oracleEbr/opss_
user.sql</Command>
            <Parameters>
                <Parameter TYPE="CmdLine">%SCHEMA_USER%</Parameter>
                <Parameter TYPE="CmdLine">%SCHEMA_PASSWORD%</Parameter>
                <Parameter TYPE="CmdLine">%DEFAULT_TABLESPACE%</Parameter>
                <Parameter TYPE="CmdLine">%TEMPORARY_TABLESPACE%</Parameter>
                <Parameter TYPE="CmdLine">%EDITION_NAME%</Parameter>
            </Parameters>
            <IgnorableErrors>
                <Error Type="ORA-01918">user name does not exist</Error>
            </IgnorableErrors>
        </Action>

        <Action TYPE="JDBCSqlScript" PERCENT_PROGRESS="40">
            <ValidIf DBTYPE="SQLSERVER" />
            <Command TYPE="SCRIPT">%SCRIPT_HOME%/opss/scripts/sqlserver/opss_
user.sql</Command>
            <Parameters>
                <Parameter TYPE="CmdLine">%SCHEMA_USER%</Parameter>
                <Parameter TYPE="CmdLine">%SCHEMA_PASSWORD%</Parameter>
```

```
                <Parameter TYPE="CmdLine">%DATABASE_NAME%</Parameter>
            </Parameters>
        </Action>
        <Action TYPE="JDBCSqlScript" PERCENT_PROGRESS="40">
            <ValidIf DBTYPE="IBMDB2"/>
            <Command TYPE="SCRIPT">%SCRIPT_HOME%/opss/scripts/db2/opss_
user.sql</Command>
            <Parameters>
                <Parameter TYPE="CmdLine">%SCHEMA_USER%</Parameter>
                <Parameter TYPE="CmdLine">%DEFAULT_TABLESPACE%</Parameter>
                <Parameter TYPE="CmdLine">%SCHEMA_USER%</Parameter>
            </Parameters>
        </Action>

 <Action TYPE="JDBCSqlScript" PERCENT_PROGRESS="60">
            <ValidIf DBTYPE="ORACLE" />
            <Command TYPE="SCRIPT">%SCRIPT_HOME%/opss/scripts/opss_
tables.sql</Command>
            <Parameters>
                <Parameter TYPE="CmdLine">%SCHEMA_USER%</Parameter>
            </Parameters>
            <IgnorableErrors>
                <Error Type="ORA-00942">table or view does not exist</Error>
            </IgnorableErrors>
        </Action>
        <Action TYPE="JDBCSqlScript" PERCENT_PROGRESS="60">
            <ValidIf DBTYPE="EBR" />
            <Command TYPE="SCRIPT">%SCRIPT_HOME%/opss/scripts/oracleEbr/opss_
tables.sql</Command>
            <Parameters>
                <Parameter TYPE="CmdLine">%SCHEMA_USER%</Parameter>
                <Parameter TYPE="CmdLine">%EDITION_NAME%</Parameter>
            </Parameters>
            <IgnorableErrors>
                <Error Type="ORA-00942">table or view does not exist</Error>
            </IgnorableErrors>
        </Action>

        <Action TYPE="JDBCSqlScript" PERCENT_PROGRESS="60">
            <ValidIf DBTYPE="SQLSERVER" />
            <Command TYPE="SCRIPT">%SCRIPT_HOME%/opss/scripts/sqlserver/opss_
tables.sql</Command>
            <Parameters>
                <Parameter TYPE="CmdLine">%DATABASE_NAME%</Parameter>
            </Parameters>
        </Action>
        <Action TYPE="JDBCSqlScript" PERCENT_PROGRESS="60">
            <ValidIf DBTYPE="IBMDB2" />
            <Command TYPE="SCRIPT">%SCRIPT_HOME%/opss/scripts/db2/opss_
tables.sql</Command>
            <Parameters>
                <Parameter TYPE="CmdLine">%SCHEMA_USER%</Parameter>
                <Parameter TYPE="CmdLine">%DEFAULT_TABLESPACE%</Parameter>
                <Parameter TYPE="CmdLine">%TEMPORARY_TABLESPACE%</Parameter>
            </Parameters>
        </Action>

        <Action TYPE="JDBCSqlScript" PERCENT_PROGRESS="70">
            <ValidIf DBTYPE="ORACLE" />
            <Command TYPE="SCRIPT">%SCRIPT_HOME%/opss/scripts/opss_
```

```
version.sql</Command>
        <Parameters>
            <Parameter TYPE="CmdLine">%SCHEMA_USER%</Parameter>
        </Parameters>
    </Action>
    <Action TYPE="JDBCSqlScript" PERCENT_PROGRESS="70">
        <ValidIf DBTYPE="EBR" />
        <Command TYPE="SCRIPT">%SCRIPT_HOME%/opss/scripts/oracleEbr/opss_
version.sql</Command>
        <Parameters>
            <Parameter TYPE="CmdLine">%SCHEMA_USER%</Parameter>
            <Parameter TYPE="CmdLine">%EDITION_NAME%</Parameter>
        </Parameters>
    </Action>

 <Action TYPE="JDBCSqlScript" PERCENT_PROGRESS="70">
        <ValidIf DBTYPE="SQLSERVER" />
        <Command TYPE="SCRIPT">%SCRIPT_HOME%/opss/scripts/sqlserver/opss_
version.sql</Command>
        <Parameters>
            <Parameter TYPE="CmdLine">%DATABASE_NAME%</Parameter>
        </Parameters>
    </Action>
    <Action TYPE="JDBCSqlScript" PERCENT_PROGRESS="70">
        <ValidIf DBTYPE="IBMDB2" />
        <Command TYPE="SCRIPT">%SCRIPT_HOME%/opss/scripts/db2/opss_
version.sql</Command>
        <Parameters>
            <Parameter TYPE="CmdLine">%SCHEMA_USER%</Parameter>
        </Parameters>
    </Action>

    <Action TYPE="JDBCSqlScript" PERCENT_PROGRESS="80">
        <ValidIf DBTYPE="ORACLE" />
        <Command TYPE="SCRIPT">%SCRIPT_HOME%/opss/scripts/opss_
gencatalog.sql</Command>
        <Parameters>
            <Parameter TYPE="CmdLine">%SCHEMA_USER%</Parameter>
        </Parameters>
    </Action>

    <Action TYPE="JDBCSqlScript" PERCENT_PROGRESS="80">
        <ValidIf DBTYPE="SQLSERVER" />
        <Command TYPE="SCRIPT">%SCRIPT_HOME%/opss/scripts/sqlserver/opss_
gencatalog.sql</Command>
        <Parameters>
            <Parameter TYPE="CmdLine">%DATABASE_NAME%</Parameter>
        </Parameters>
    </Action>

    <Action TYPE="JDBCSqlScript" PERCENT_PROGRESS="80">
        <ValidIf DBTYPE="IBMDB2" />
        <Command TYPE="SCRIPT">%SCRIPT_HOME%/opss/scripts/db2/opss_
gencatalog.sql</Command>
        <Parameters>
            <Parameter TYPE="CmdLine">%SCHEMA_USER%</Parameter>
        </Parameters>
    </Action>

    <Action TYPE="JDBCSqlScript" PERCENT_PROGRESS="80">
```

```
            <ValidIf DBTYPE="EBR" />
            <Command TYPE="SCRIPT">%SCRIPT_HOME%/opss/scripts/oracleEbr/opss_
gencatalog.sql</Command>
            <Parameters>
                <Parameter TYPE="CmdLine">%SCHEMA_USER%</Parameter>
                <Parameter TYPE="CmdLine">%EDITION_NAME%</Parameter>
            </Parameters>
        </Action>
        <Action TYPE="JDBCSqlScript"  CONNECT_AS_OWNER="TRUE" PERCENT_
PROGRESS="100">
            <ValidIf DBTYPE="ORACLE" />
            <Command TYPE="SCRIPT">%SCRIPT_HOME%/opss/scripts/upgd_mgmt_
schema.sql</Command>
        </Action>

        <Action TYPE="Java">
            <Command
TYPE="METHOD">oracle.ias.version.SchemaVersionUtil:utilSetComponentValid</Command>
            <Parameters>
                <Parameter TYPE="String">OPSS</Parameter>
            </Parameters>
        </Action>
 </ExecutionDescriptor>

 <!-- Deleting the OPSS Schema and removing it from the Registry -->
   <DeleteDescriptor>
        <Action TYPE="JDBC" PERCENT_PROGRESS="50">
            <ValidIf DBTYPE="ORACLE" />
            <Command TYPE="INLINE">DROP USER %SCHEMA_USER% CASCADE</Command>
        </Action>
        <Action TYPE="JDBC" PERCENT_PROGRESS="50">
            <ValidIf DBTYPE="EBR" />
            <Command TYPE="INLINE">DROP USER %SCHEMA_USER% CASCADE</Command>
        </Action>

        <Action TYPE="JDBCSqlScript" PERCENT_PROGRESS="20">
            <ValidIf DBTYPE="SQLSERVER" />
            <Command TYPE="SCRIPT">%SCRIPT_HOME%/opss/scripts/sqlserver/opss_drop_
table.sql</Command>
            <Parameters>
                <Parameter TYPE="CmdLine">%DATABASE_NAME%</Parameter>
            </Parameters>
        </Action>
        <Action TYPE="JDBCSqlScript" PERCENT_PROGRESS="50">
            <ValidIf DBTYPE="SQLSERVER" />
            <Command TYPE="SCRIPT">%SCRIPT_HOME%/opss/scripts/sqlserver/opss_drop_
user.sql</Command>
            <Parameters>
                <Parameter TYPE="CmdLine">%DATABASE_NAME%</Parameter>
                <Parameter TYPE="CmdLine">%SCHEMA_USER%</Parameter>
            </Parameters>
        </Action>

      <Action TYPE="Java" PERCENT_PROGRESS="50">
        <ValidIf DBTYPE="IBMDB2" />
        <Command
TYPE="METHOD">oracle.sysman.assistants.common.dbutil.jdbc.DB2DropUtil:dropCompRepo
s</Command>
            <Parameters>
            <Parameter TYPE="Connection"></Parameter>
```

```
            <Parameter TYPE="String">%SCHEMA_USER%</Parameter>
          </Parameters>
        </Action>

        <Action TYPE="Java" PERCENT_PROGRESS="100">
            <Command
TYPE="METHOD">oracle.ias.version.SchemaVersionUtil:utilDropRegistryEntry</Command>
            <Parameters>
                <Parameter TYPE="Connection"></Parameter>
                <Parameter TYPE="String">OPSS</Parameter>
                <Parameter TYPE="String">%PREFIX_NAME%</Parameter>
                <Parameter TYPE="String">OPSS</Parameter>
            </Parameters>
        </Action>
    </DeleteDescriptor>
</RepositoryConfig>
```

C.2.3 Component List Configuration File

The Component List configuration file (`ComponentInfo.xml`) lists all the components, their respective configuration files and their default user and tablespace mappings. This file also lists the high-level pre-requisite checks and high level actions (like creating `schema_version_registry` table) to be done globally for all the components. Also, a list of global Ignorable or Fatal errors can be specified.

This file can be found in the *ORACLE_HOME*/oracle_common/rcu/config (on UNIX operating systems) or *ORACLE_HOME*\oracle_common\rcu\config (on Windows operating systems) directory.

Below is a sample `ComponentInfo.xml` file:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!-- DOCTYPE ComponentInfo SYSTEM "dtds/ComponentInfo.dtd" -->
<!DOCTYPE ComponentInfo SYSTEM "ComponentInfo.dtd" []>
<ComponentInfo VERSION="11.0.0.0" TYPE="AS_REPOSITORY" RESOURCE_BUNDLE_
PACKAGE="oracle.sysman.rcu.as.ASBundle">
  <Display NLS_ID="ASREP_ID">Oracle AS Repository Components</Display>
  <PrefixSettings USE_SCHEMA_PREFIX="TRUE" USE_TABLESPACE_PREFIX="TRUE">
      <DetectQuery>
          Select distinct mrc_name from schema_version_registry
      </DetectQuery>
      <DetectQuery TYPE="IBMDB2">
          Select distinct mrc_name from NULLID.schema_version_registry
      </DetectQuery>
          <DetectQuery TYPE="JAVADB">
          Select distinct mrc_name from ORACLEFMW.schema_version_registry
      </DetectQuery>
  </PrefixSettings>

  <!-- AS Common GROUP START -->
  <Component ID="AS_COMMON" IS_GROUPING_COMPONENT="TRUE">
      <Display NLS_ID="AS_COMMON_ID">AS Common Schemas</Display>
  </Component>

  <Component ID="SOA" IS_GROUPING_COMPONENT="TRUE">
      <Display NLS_ID="SOA">SOA Suite</Display>
```

```
                        </Component>


     <Component ID="ODI_REPOSITORIES" IS_GROUPING_COMPONENT="TRUE">
          <Display NLS_ID="ODI_REPOSITORIES">Oracle Data Integrator</Display>
     </Component>



 <ComponentConfigFiles>
     <ConfigFile ID="MDS">%RCU_HOME%/../oracle_common/common/sql/mds/mds_
ComponentInfo.xml</ConfigFile>
     <ConfigFile ID="IAU">%RCU_HOME%/../oracle_common/common/sql/iau/iau_
ComponentInfo.xml</ConfigFile>
     <ConfigFile ID="OPSS">%RCU_HOME%/../oracle_common/common/sql/opss/opss_
ComponentInfo.xml</ConfigFile>
     <ConfigFile ID="UCSMESSAGING">%RCU_HOME%/../oracle_
common/common/sql/ucs.messaging/ucs.messaging_ComponentInfo.xml</ConfigFile>
     <ConfigFile ID="WLS">%RCU_HOME%/../oracle_
common/common/sql/wlsservices/wlsservices_ComponentInfo.xml</ConfigFile>
     <ConfigFile ID="ESS">%RCU_HOME%/../oracle_common/common/sql/ess/ess_
ComponentInfo.xml</ConfigFile>
     <ConfigFile ID="SOAINFRA">%RCU_HOME%/../soa/common/sql/soainfra/soainfra_
ComponentInfo.xml</ConfigFile>
     <ConfigFile ID="ODI">%RCU_HOME%/../odi/common/sql/odi/odi_
ComponentInfo.xml</ConfigFile>
<ConfigFile ID="OER">%RCU_HOME%/../oer/common/sql/oer/oer_
ComponentInfo.xml</ConfigFile>
<ConfigFile ID="OWLCS">%RCU_HOME%/../oracle_
common/common/sql/ucs.callcontrol/ucs.callcontrol_ComponentInfo.xml</ConfigFile>
<ConfigFile ID="SVCTBL">%RCU_HOME%/../oracle_common/common/sql/svctbl/svctbl_
ComponentInfo.xml</ConfigFile>
<ConfigFile ID="MFT">%RCU_HOME%/../mft/common/sql/mft/mft_
ComponentInfo.xml</ConfigFile>
 </ComponentConfigFiles>

<PrerequisiteDescriptor>
      <DBPrerequisiteSet OPERATOR="OR">
      <ValidIf DBTYPE="ORACLE" />
       <DBPrerequisite PREREQ_TYPE="InitParameter" DATA_TYPE="NUMBER" COMPARE_
OPERATOR="GE">
            <PrereqIdentifier>SHARED_POOL_SIZE</PrereqIdentifier>
            <PrereqValue UNIT="KB">147456</PrereqValue>
         </DBPrerequisite>
        <DBPrerequisite PREREQ_TYPE="InitParameter" DATA_TYPE="NUMBER" COMPARE_
OPERATOR="GE">
            <PrereqIdentifier>SGA_MAX_SIZE</PrereqIdentifier>
            <PrereqValue UNIT="KB">147456</PrereqValue>
        </DBPrerequisite>
     </DBPrerequisiteSet>

         <DBPrerequisiteSet OPERATOR="AND">
            <ValidIf DBTYPE="ORACLE" />
     <DBPrerequisite PREREQ_TYPE="CustomSQL" DATA_TYPE="NUMBER" COMPARE_
OPERATOR="EQ"  SOFT="TRUE">
     <ValidIf DBTYPE="ORACLE">
     <CustomQueryFilter DATA_TYPE="NUMBER" COMPARE_OPERATOR="EQ" VALUE="1">
        select 1 from dual where exists (select column_name from dba_tab_columns
where table_name(+) like 'V_$INSTANCE' and column_name(+) = 'EDITION') union
select 0 from dual where not exists (select column_name from dba_tab_columns where
```

```
table_name(+) like 'V_$INSTANCE' and column_name(+) = 'EDITION')
   </CustomQueryFilter>
     </ValidIf>
           <PrereqIdentifier>select count (*) from V_$INSTANCE where EDITION =
'XE'</PrereqIdentifier>
           <PrereqValue>0</PrereqValue>
           <PrereqErrorMsg>
                                   The database you are connecting is not a
supported version. Refer to the certification matrix for supported DB versions.
           </PrereqErrorMsg>
   </DBPrerequisite>

   <DBPrerequisite PREREQ_TYPE="CustomSQL" DATA_TYPE="NUMBER" COMPARE_
OPERATOR="EQ">
     <ValidIf DBTYPE="ORACLE">
     <CustomQueryFilter DATA_TYPE="NUMBER" COMPARE_OPERATOR="EQ" VALUE="1">
       select 1 from dual where exists (select column_name from dba_tab_columns
where table_name(+) like 'V_$INSTANCE' and column_name(+) = 'EDITION') union
select 0 from dual where not exists (select column_name from dba_tab_columns where
table_name(+) like 'V_$INSTANCE' and column_name(+) = 'EDITION')
        </CustomQueryFilter>
     </ValidIf>
             <PrereqIdentifier>select count(*) from product_component_version
where product like 'Oracle%Database%' AND version >= '12.0.0.0.0' AND  version
&lt;  '12.1.0.1.0'</PrereqIdentifier>
           <PrereqValue>0</PrereqValue>
           <PrereqErrorMsg>
                                 The database you are connecting is not a supported
version. Enter Database with version equal to or higher than 12.1.0.1.0 in 12c.
Refer to the certification matrix for supported DB versions.
             </PrereqErrorMsg>
   </DBPrerequisite>

 <DBPrerequisite PREREQ_TYPE="CustomSQL" DATA_TYPE="NUMBER" COMPARE_OPERATOR="EQ"
SOFT="TRUE">
     <ValidIf DBTYPE="ORACLE">
     <CustomQueryFilter DATA_TYPE="NUMBER" COMPARE_OPERATOR="EQ" VALUE="1">
       select 1 from dual where exists (select column_name from dba_tab_columns
where table_name(+) like 'V_$INSTANCE' and column_name(+) = 'EDITION') union
select 0 from dual where not exists (select column_name from dba_tab_columns where
table_name(+) like 'V_$INSTANCE' and column_name(+) = 'EDITION')
        </CustomQueryFilter>
     </ValidIf>
      <PrereqIdentifier>select count(*) from product_component_version where
product like 'Oracle%Database%' AND version > '12.1.0.1.9'</PrereqIdentifier>
           <PrereqValue>0</PrereqValue>
           <PrereqErrorMsg>
       The database you are connecting to, is a more recent one than the
supported version. Refer to the certification matrix for supported DB versions.
             </PrereqErrorMsg>
   </DBPrerequisite>

  </DBPrerequisiteSet>



     <DBPrerequisite PREREQ_TYPE="InitParameter" DATA_TYPE="NUMBER" COMPARE_
OPERATOR="GE">
           <ValidIf DBTYPE="ORACLE" />
           <PrereqIdentifier>DB_BLOCK_SIZE</PrereqIdentifier>
```

```
                                 <PrereqValue UNIT="KB">8</PrereqValue>
                    </DBPrerequisite>
                    <!--DBPrerequisite PREREQ_TYPE="CustomSQL" DATA_TYPE="STRING" COMPARE_
OPERATOR="NE">
                    <ValidIf  DBTYPE="ORACLE" >
                    <CustomQueryFilter DATA_TYPE="NUMBER" COMPARE_OPERATOR="EQ" VALUE="0">
                        select 1 from dual where exists (select column_name from dba_tab_
columns where table_name(+) like 'V_$INSTANCE' and column_name(+) = 'EDITION')
union select 0 from dual where not exists (select column_name from dba_tab_columns
where table_name(+) like 'V_$INSTANCE' and column_name(+) = 'EDITION')
                        </CustomQueryFilter>
                    </ValidIf>
                    <PrereqIdentifier>version</PrereqIdentifier>
                    <PrereqValue>11.1.0.6.0</PrereqValue>
                            <PrereqErrorMsg>
                                        The database you are connecting is 11.1.0.6.0
version. 11.1.0.6.0 is not a supported version. The database version should be
11.1.0.7.0 or greater.
                            </PrereqErrorMsg>
                    </DBPrerequisite-->
                    <DBPrerequisite PREREQ_TYPE="DBVersion" DATA_TYPE="STRING" COMPARE_
OPERATOR="GE">
                    <ValidIf  DBTYPE="ORACLE" >
                    <CustomQueryFilter DATA_TYPE="NUMBER" COMPARE_OPERATOR="EQ" VALUE="0">
                        select 1 from dual where exists (select column_name from dba_tab_
columns where table_name(+) like 'V_$INSTANCE' and column_name(+) = 'EDITION')
union select 0 from dual where not exists (select column_name from dba_tab_columns
where table_name(+) like 'V_$INSTANCE' and column_name(+) = 'EDITION')
                        </CustomQueryFilter>
                    </ValidIf>
                    <PrereqIdentifier>version</PrereqIdentifier>
                    <PrereqValue>10.2.0.4.0</PrereqValue>
                            <PrereqErrorMsg>
 The database you are connecting is not a supported version.  Enter Database with
version equal to or higher than 10.2.0.4.0 in 10g or version equal to or higher
than 11.1.0.7.0 in 11g. Refer to the certification matrix for supported DB
versions
                            </PrereqErrorMsg>
                    </DBPrerequisite>


<DBPrerequisite PREREQ_TYPE="CustomSQL" DATA_TYPE="NUMBER" COMPARE_OPERATOR="EQ">
                    <ValidIf  DBTYPE="ORACLE" >
                    <CustomQueryFilter DATA_TYPE="NUMBER" COMPARE_OPERATOR="EQ" VALUE="0">
                        select 1 from dual where exists (select column_name from dba_tab_
columns where table_name(+) like 'V_$INSTANCE' and column_name(+) = 'EDITION')
union select 0 from dual where not exists (select column_name from dba_tab_columns
where table_name(+) like 'V_$INSTANCE' and column_name(+) = 'EDITION')
                        </CustomQueryFilter>
                    </ValidIf>
                    <PrereqIdentifier>select count(*) from product_component_version  where
product like 'Oracle%Database%'  AND version BETWEEN '11' AND '11.1.0.6.0'
</PrereqIdentifier>
                    <PrereqValue>0</PrereqValue>
                            <PrereqErrorMsg>
                                        The database you are connecting is not a supported
version.  Enter Database with version equal to or higher than 10.2.0.4.0 in 10g or
version equal to or higher than 11.1.0.7.0 in 11g. Refer to the certification
matrix for supported DB versions
                            </PrereqErrorMsg>
```

```
            </DBPrerequisite>


<DBPrerequisite PREREQ_TYPE="CustomSQL" DATA_TYPE="NUMBER" COMPARE_OPERATOR="EQ">
            <ValidIf DBTYPE="IBMDB2"/>
            <PrereqIdentifier>select count(*) from syscat.tablespaces where
tbspace = 'SYSCATSPACE' and pagesize >= 32768
            </PrereqIdentifier>
            <PrereqValue>1</PrereqValue>
            <PrereqErrorMsg>
                        Component     : RCU
                        Error         : Database prerequisite check failed.
                        Cause         : Database: '%DATABASE_NAME%' needs to be
configured with
                        default pagesize 32768 or 32K.
                        Action        : Modify the default of the current database
or create a new
                        database with the required default pagesize.
            </PrereqErrorMsg>
    </DBPrerequisite>


<DBPrerequisite PREREQ_TYPE="CustomSQL" DATA_TYPE="STRING" COMPARE_OPERATOR="EQ">
                <ValidIf DBTYPE="SQLSERVER" />
                 <PrereqIdentifier>SELECT count(*) where
CAST(SERVERPROPERTY('productversion') as VARCHAR(20)) like '1.%' or
CAST(SERVERPROPERTY('productversion') as VARCHAR(20)) like '4.%' or
CAST(SERVERPROPERTY('productversion') as VARCHAR(20)) like '6.%' or
CAST(SERVERPROPERTY('productversion') as VARCHAR(20)) like '7.%' or
CAST(SERVERPROPERTY('productversion') as VARCHAR(20)) like
'8.%'</PrereqIdentifier>
                <PrereqValue>0</PrereqValue>
                 <PrereqErrorMsg>
                The database you are connecting is not a supported version.
Enter Database with version equal to or higher than 2005. Refer to the
certification matrix for supported DB versions.
                </PrereqErrorMsg>
</DBPrerequisite>

 <DBPrerequisite PREREQ_TYPE="CustomSQL" DATA_TYPE="STRING" COMPARE_OPERATOR="EQ">
                <ValidIf DBTYPE="IBMDB2" />
                <PrereqIdentifier>select count(*) FROM TABLE (sysproc.env_get_
inst_info()) where INT(substr(service_level, POSSTR(service_level, 'v')+1,
LOCATE('.', service_level,POSSTR(service_level, 'v') +1) - POSSTR(service_level,
'v') -1 ) ) = 9  AND INT(substr(service_level, POSSTR(service_level,  '.')+1,
LOCATE('.',service_level,POSSTR(service_level, '.') +1) - POSSTR(service_level,
'.') -1 )) &lt; 7 OR INT(substr(service_level, POSSTR(service_level, 'v')+1,
LOCATE('.', service_level,POSSTR(service_level, 'v') +1) - POSSTR(service_level,
'v') -1 ) ) &lt; 9</PrereqIdentifier>
                <PrereqValue>0</PrereqValue>
                <PrereqErrorMsg>
                        The database you are connecting is not a supported
version. Enter Database with version equal to or higher than 9.7. Refer to the
certification matrix for supported DB versions.
                </PrereqErrorMsg>
</DBPrerequisite>


        <DBPrerequisite PREREQ_TYPE="CustomSQL" DATA_TYPE="STRING" COMPARE_
OPERATOR="EQ">
                <ValidIf DBTYPE="ORACLE" />
```

```
                    <PrereqIdentifier>select GRANTED_ROLE from DBA_ROLE_PRIVS
where((GRANTED_ROLE='DBA' and GRANTEE=(select user from dual) and lower(SYS_
CONTEXT ('USERENV', 'SESSION_USER'))='sys') OR(GRANTED_ROLE='DBA' and
GRANTEE=(select user from dual)))</PrereqIdentifier>
                    <PrereqValue>DBA</PrereqValue>
                    <PrereqErrorMsg>
                            User should have sysdba or dba privilages.
                    </PrereqErrorMsg>
                </DBPrerequisite>

<DBPrerequisite PREREQ_TYPE="CustomSQL" DATA_TYPE="STRING" COMPARE_OPERATOR="EQ">
                    <ValidIf DBTYPE="SQLSERVER" />
                    <PrereqIdentifier>select Is_Member('db_
owner')</PrereqIdentifier>
                    <PrereqValue>1</PrereqValue>
                    <PrereqErrorMsg>
                            User should have sysdba or dba privilages.
                    </PrereqErrorMsg>
    </DBPrerequisite>

        <DBPrerequisite PREREQ_TYPE="CustomSQL" DATA_TYPE="STRING" COMPARE_
OPERATOR="EQ" SOFT="TRUE">
                    <ValidIf DBTYPE="ORACLE" />
                    <PrereqIdentifier>select value from nls_database_parameters
where parameter = 'NLS_CHARACTERSET'</PrereqIdentifier>
                    <PrereqValue>AL32UTF8</PrereqValue>
                    <PrereqErrorMsg>
                                 The database you are connecting is with
non-AL32UTF8 character set. Oracle strongly recommends using AL32UTF8 as the
database character set.
                    </PrereqErrorMsg>
                </DBPrerequisite>

 <DBPrerequisite PREREQ_TYPE="CustomSQL" DATA_TYPE="NUMBER" COMPARE_OPERATOR="EQ"
SOFT="TRUE">
        <ValidIf  DBTYPE="ORACLE" >
        <CustomQueryFilter DATA_TYPE="NUMBER" COMPARE_OPERATOR="EQ" VALUE="0">
           select 1 from dual where exists (select column_name from dba_tab_
columns where table_name(+) like 'V_$INSTANCE' and column_name(+) = 'EDITION')
union select 0 from dual where not exists (select column_name from dba_tab_columns
where table_name(+) like 'V_$INSTANCE' and column_name(+) = 'EDITION')
          </CustomQueryFilter>
        </ValidIf>
        <PrereqIdentifier>select count(*) from product_component_version  where
product like 'Oracle%Database%'  AND version BETWEEN '10.2.0.6.0' AND '10.9.9.9.9'
</PrereqIdentifier>
        <PrereqValue>0</PrereqValue>
                <PrereqErrorMsg>
                The database you are connecting to, is a more recent than the
supported version. Refer to the certification matrix for supported DB versions.
                </PrereqErrorMsg>
        </DBPrerequisite>


        <DBPrerequisite PREREQ_TYPE="CustomSQL" DATA_TYPE="NUMBER" COMPARE_
OPERATOR="EQ" SOFT="TRUE">
        <ValidIf  DBTYPE="ORACLE" >
        <CustomQueryFilter DATA_TYPE="NUMBER" COMPARE_OPERATOR="EQ" VALUE="0">
           select 1 from dual where exists (select column_name from dba_tab_
columns where table_name(+) like 'V_$INSTANCE' and column_name(+) = 'EDITION')
```

```
union select 0 from dual where not exists (select column_name from dba_tab_columns
where table_name(+) like 'V_$INSTANCE' and column_name(+) = 'EDITION')
        </CustomQueryFilter>
      </ValidIf>
      <PrereqIdentifier>select count(*) from product_component_version  where
product like 'Oracle%Database%'  AND version &gt; '11.1.0.7.0' AND version &lt;=
'11.1.9.9.9'</PrereqIdentifier>
      <PrereqValue>0</PrereqValue>
            <PrereqErrorMsg>
         The database you are connecting to, is a more recent one than the
supported version. Refer to the certification matrix for supported DB versions.
            </PrereqErrorMsg>
      </DBPrerequisite>


      <DBPrerequisite PREREQ_TYPE="CustomSQL" DATA_TYPE="NUMBER" COMPARE_
OPERATOR="EQ" SOFT="TRUE">
      <ValidIf  DBTYPE="ORACLE" >
      <CustomQueryFilter DATA_TYPE="NUMBER" COMPARE_OPERATOR="EQ" VALUE="0">
         select 1 from dual where exists (select column_name from dba_tab_
columns where table_name(+) like 'V_$INSTANCE' and column_name(+) = 'EDITION')
union select 0 from dual where not exists (select column_name from dba_tab_columns
where table_name(+) like 'V_$INSTANCE' and column_name(+) = 'EDITION')
        </CustomQueryFilter>
      </ValidIf>
      <PrereqIdentifier>select count(*) from product_component_version  where
product like 'Oracle%Database%'  AND version between '11.2.0.4.0' AND '11.2.9.9.9'
</PrereqIdentifier>
      <PrereqValue>0</PrereqValue>
            <PrereqErrorMsg>
         The database you are connecting to, is a more recent one than the
supported version. Refer to the certification matrix for supported DB versions.
                              </PrereqErrorMsg>
      </DBPrerequisite>

<DBPrerequisite PREREQ_TYPE="CustomSQL" DATA_TYPE="NUMBER" COMPARE_OPERATOR="EQ">
      <ValidIf  DBTYPE="ORACLE" >
      <CustomQueryFilter DATA_TYPE="NUMBER" COMPARE_OPERATOR="EQ" VALUE="0">
         select 1 from dual where exists (select column_name from dba_tab_
columns where table_name(+) like 'V_$INSTANCE' and column_name(+) = 'EDITION')
union select 0 from dual where not exists (select column_name from dba_tab_columns
where table_name(+) like 'V_$INSTANCE' and column_name(+) = 'EDITION')
        </CustomQueryFilter>
      </ValidIf>
                <PrereqIdentifier>select count(*) from product_component_version
where product like 'Oracle%Database%'  AND version between '11.2.0.0.0' AND
'11.2.0.2.9' </PrereqIdentifier>
                <PrereqValue>0</PrereqValue>
                <PrereqErrorMsg>
                              The database you are connecting is not a supported
version.  Enter Database with version equal to or higher than 11.2.0.3.0 in 11g.
Refer to the certification matrix for supported DB versions.
                </PrereqErrorMsg>
      </DBPrerequisite>
<!--
<DBPrerequisite PREREQ_TYPE="CustomSQL" DATA_TYPE="NUMBER" COMPARE_OPERATOR="EQ"
SOFT="TRUE">
      <ValidIf  DBTYPE="ORACLE" >
      <CustomQueryFilter DATA_TYPE="NUMBER" COMPARE_OPERATOR="EQ" VALUE="0">
         select 1 from dual where exists (select column_name from dba_tab_
columns where table_name(+) like 'V_$INSTANCE' and column_name(+) = 'EDITION')
```

```
union select 0 from dual where not exists (select column_name from dba_tab_columns
where table_name(+) like 'V_$INSTANCE' and column_name(+) = 'EDITION')
        </CustomQueryFilter>
      </ValidIf>
              <PrereqIdentifier>select count(*) from product_component_version
where product like 'Oracle%Database%'  AND version >= '12.2._._._'
</PrereqIdentifier>
              <PrereqValue>0</PrereqValue>
              <PrereqErrorMsg>
        The database you are connecting to, is a more recent one than the
supported version. Refer to the certification matrix for supported DB versions.
            </PrereqErrorMsg>
      </DBPrerequisite>

<DBPrerequisite PREREQ_TYPE="CustomSQL" DATA_TYPE="NUMBER" COMPARE_OPERATOR="EQ">
      <ValidIf  DBTYPE="ORACLE" >
      <CustomQueryFilter DATA_TYPE="NUMBER" COMPARE_OPERATOR="EQ" VALUE="0">
          select 1 from dual where exists (select column_name from dba_tab_
columns where table_name(+) like 'V_$INSTANCE' and column_name(+) = 'EDITION')
union select 0 from dual where not exists (select column_name from dba_tab_columns
where table_name(+) like 'V_$INSTANCE' and column_name(+) = 'EDITION')
        </CustomQueryFilter>
      </ValidIf>
  <PrereqIdentifier>select count(*) from product_component_version  where
product like 'Oracle%Database%'  AND version like '12.0._._._'
</PrereqIdentifier>
    <PrereqValue>0</PrereqValue>
     <PrereqErrorMsg>
                            The database you are connecting is not a supported
version.  Enter Database with version equal to or higher than 12.1.0.0.0 in 12c.
Refer to the certification matrix for supported DB versions.
        </PrereqErrorMsg>
 </DBPrerequisite>
-->

 <DBPrerequisite PREREQ_TYPE="CustomSQL" DATA_TYPE="STRING" COMPARE_OPERATOR="EQ"
SOFT="TRUE">
              <ValidIf DBTYPE="SQLSERVER" />
               <PrereqIdentifier>SELECT count(*) where
CAST(SERVERPROPERTY('productversion') as VARCHAR(20)) not like '1.%' and
CAST(SERVERPROPERTY('productversion') as VARCHAR(20)) not like '4.%' and
CAST(SERVERPROPERTY('productversion') as VARCHAR(20)) not like '6.%' and
CAST(SERVERPROPERTY('productversion') as VARCHAR(20)) not like '7.%' and
CAST(SERVERPROPERTY('productversion') as VARCHAR(20)) not like '8.%' and
CAST(SERVERPROPERTY('productversion') as VARCHAR(20)) not like '9.%' and
CAST(SERVERPROPERTY('productversion') as VARCHAR(20)) not like '10.%'
</PrereqIdentifier>
              <PrereqValue>0</PrereqValue>
              <PrereqErrorMsg>
               The database you are connecting to, is a more recent than the
supported version.  Refer to the certification matrix for supported DB versions.
            </PrereqErrorMsg>
      </DBPrerequisite>

      <DBPrerequisite PREREQ_TYPE="CustomSQL" DATA_TYPE="STRING" COMPARE_
OPERATOR="EQ" SOFT="TRUE">
              <ValidIf DBTYPE="IBMDB2" />
  <PrereqIdentifier>select count(*) FROM TABLE (sysproc.env_get_inst_info())
where INT(substr(service_level, POSSTR(service_level, 'v')+1, LOCATE('.', service_
level, POSSTR(service_level, 'v') +1) - POSSTR(service_level, 'v') -1 ) ) = 9 and
```

```
INT(substr(service_level, POSSTR(service_level, '.')+1, LOCATE('.', service_level,
POSSTR(service_level, '.') +1) - POSSTR(service_level, '.') -1 )) &gt; 7 or
INT(substr(service_level, POSSTR(service_level, 'v')+1, LOCATE('.', service_level,
POSSTR(service_level, 'v') +1) - POSSTR(service_level, 'v') -1 ) ) &gt;
9</PrereqIdentifier>
     <PrereqValue>0</PrereqValue>
      <PrereqErrorMsg>
                          The database you are connecting to, is a more recent
than the supported version. Refer to the certification matrix for supported DB
versions.
          </PrereqErrorMsg>
 </DBPrerequisite>


<DBPrerequisite PREREQ_TYPE="Java" DATA_TYPE="STRING" COMPARE_OPERATOR="NE"
SOFT="TRUE">
   <ValidIf DBTYPE="JAVADB" />
    <PrereqIdentifier>%RCU_HOME%/../oracle_
common/rcu/config/JavaDB.jar:javadbproject.JavaDB:checkJavaDBVersion</PrereqIdenti
fier>
       <PrereqValue>SOFTPASS</PrereqValue>
       <PrereqErrorMsg>
           The database you are connecting to, is a more recent than the
supported version. Refer to the certification matrix for supported DB versions.
       </PrereqErrorMsg>
</DBPrerequisite>


<DBPrerequisite PREREQ_TYPE="Java" DATA_TYPE="STRING" COMPARE_OPERATOR="NE">
   <ValidIf DBTYPE="JAVADB" />
                 <PrereqIdentifier>%RCU_HOME%/../oracle_
common/rcu/config/JavaDB.jar:javadbproject.JavaDB:checkJavaDBVersion</PrereqIdenti
fier>
                 <PrereqValue>FAIL</PrereqValue>
                 <PrereqErrorMsg>
           The database you are connecting is not a supported version. Enter
Database with version equal to or higher than 10.5.3.0. Refer to the certification
matrix for supported DB versions.
                 </PrereqErrorMsg>
      </DBPrerequisite>

 <DBPrerequisite PREREQ_TYPE="CustomSQL" DATA_TYPE="NUMBER" COMPARE_OPERATOR="EQ">
                 <ValidIf DBTYPE="MYSQL" />
                 <PrereqIdentifier>select version() &lt;
'5.5.14'</PrereqIdentifier>
                 <PrereqValue>0</PrereqValue>
                 <PrereqErrorMsg>
                          The database you are connecting is not a supported
version. Enter Database with version equal to or higher than 5.5.14. Refer to the
certification matrix for supported DB versions.
                 </PrereqErrorMsg>
      </DBPrerequisite>


     <DBPrerequisite PREREQ_TYPE="CustomSQL" DATA_TYPE="NUMBER" COMPARE_
OPERATOR="EQ" SOFT="TRUE">
                 <ValidIf DBTYPE="MYSQL" />
                 <PrereqIdentifier>select version() &gt; '5.6' or version() =
'5.6'</PrereqIdentifier>
                 <PrereqValue>0</PrereqValue>
                 <PrereqErrorMsg>
```

```
                                 The database you are connecting to, is a more recent
than the supported version. Refer to the certification matrix for supported DB
versions.
                </PrereqErrorMsg>
        </DBPrerequisite>


<DBPrerequisite PREREQ_TYPE="CustomSQL" DATA_TYPE="STRING" COMPARE_OPERATOR="EQ">
                <ValidIf DBTYPE="MYSQL" />
                  <PrereqIdentifier>select VARIABLE_VALUE from INFORMATION_
SCHEMA.GLOBAL_VARIABLES where VARIABLE_NAME = 'INNODB_FILE_PER_
TABLE'</PrereqIdentifier>
                    <PrereqValue>ON</PrereqValue>
                    <PrereqErrorMsg>
                      DB Init Param Prerequisite failure for INNODB_FILE_PER_TABLE.
Its value should be 'ON'.
                </PrereqErrorMsg>
        </DBPrerequisite>

<DBPrerequisite PREREQ_TYPE="CustomSQL" DATA_TYPE="STRING" COMPARE_OPERATOR="EQ">
                <ValidIf DBTYPE="MYSQL" />
                  <PrereqIdentifier>select VARIABLE_VALUE from INFORMATION_
SCHEMA.GLOBAL_VARIABLES where VARIABLE_NAME = 'INNODB_FILE_
FORMAT'</PrereqIdentifier>
                    <PrereqValue>Barracuda</PrereqValue>
                    <PrereqErrorMsg>
                      DB Init Param Prerequisite failure for INNODB_FILE_FORMAT. Its
value should be 'Barracuda'.
                </PrereqErrorMsg>
        </DBPrerequisite>

<DBPrerequisite PREREQ_TYPE="CustomSQL" DATA_TYPE="STRING" COMPARE_OPERATOR="EQ">
                <ValidIf DBTYPE="MYSQL" />
                  <PrereqIdentifier>select VARIABLE_VALUE from INFORMATION_
SCHEMA.GLOBAL_VARIABLES where VARIABLE_NAME = 'INNODB_LARGE_
PREFIX'</PrereqIdentifier>
                    <PrereqValue>ON</PrereqValue>
                    <PrereqErrorMsg>
                        DB Init Param Prerequisite failure for INNODB_LARGE_PREFIX.
Its value should be 'ON'.
                </PrereqErrorMsg>
 </DBPrerequisite>

<DBPrerequisite PREREQ_TYPE="CustomSQL" DATA_TYPE="STRING" COMPARE_OPERATOR="EQ">
                <ValidIf DBTYPE="MYSQL" />
                  <PrereqIdentifier>select VARIABLE_VALUE from INFORMATION_
SCHEMA.GLOBAL_VARIABLES where VARIABLE_NAME ='log_bin_trust_function_
creators'</PrereqIdentifier>
                    <PrereqValue>ON</PrereqValue>
                    <PrereqErrorMsg>
                        DB Init Param Prerequisite failure for log_bin_trust_
function_creators. Its value should be 'ON'.
                </PrereqErrorMsg>
 </DBPrerequisite>

 </PrerequisiteDescriptor>

   <ExecutionDescriptor TYPE="PreLoad">
         <Action TYPE="Java" PERCENT_PROGRESS="60">
         <ValidIf DBTYPE="ORACLE,EBR">
```

```
             <CustomQueryFilter DATA_TYPE="NUMBER" COMPARE_OPERATOR="EQ"
VALUE="0">
             select count(*) from  dba_views where VIEW_NAME= 'SCHEMA_VERSION_
REGISTRY'
             </CustomQueryFilter>
         </ValidIf>
         <Command
TYPE="METHOD">oracle.ias.version.SchemaVersionUtil:utilCreateRegistryTable</Comman
d>
         <Parameters>
            <Parameter TYPE="Connection"></Parameter>
         </Parameters>
     </Action>
         <Action TYPE="Java" PERCENT_PROGRESS="60">
         <ValidIf DBTYPE="SQLSERVER">
             <CustomQueryFilter DATA_TYPE="NUMBER" COMPARE_OPERATOR="EQ"
VALUE="0">
             select count(*) from  INFORMATION_SCHEMA.TABLES where TABLE_
NAME='SCHEMA_VERSION_REGISTRY'
             </CustomQueryFilter>
         </ValidIf>
         <Command
TYPE="METHOD">oracle.ias.version.SchemaVersionUtil:utilCreateRegistryTable</Comman
d>
         <Parameters>
            <Parameter TYPE="Connection"></Parameter>
         </Parameters>
     </Action>
         <Action TYPE="Java" PERCENT_PROGRESS="60">
                <ValidIf DBTYPE="IBMDB2">
             <CustomQueryFilter DATA_TYPE="NUMBER" COMPARE_OPERATOR="EQ" VALUE="0">
                          select count(*) from syscat.tables where TABNAME='SCHEMA_
VERSION_REGISTRY'
             </CustomQueryFilter>
         </ValidIf>
         <Command
TYPE="METHOD">oracle.ias.version.SchemaVersionUtil:utilCreateRegistryTable</Comman
d>
         <Parameters>
            <Parameter TYPE="Connection"></Parameter>
         </Parameters>
     </Action>
         <Action TYPE="Java" PERCENT_PROGRESS="60">
         <ValidIf DBTYPE="JAVADB">
                         <CustomQueryFilter DATA_TYPE="NUMBER" COMPARE_
OPERATOR="EQ" VALUE="0">
                SELECT  COUNT(*) FROM SYS.SYSTABLES INNER JOIN SYS.SYSSCHEMAS ON
SYS.SYSTABLES.SCHEMAID = SYS.SYSSCHEMAS.SCHEMAID WHERE SCHEMANAME='ORACLEFMW' AND
TABLENAME='SCHEMA_VERSION_REGISTRY_T'
                         </CustomQueryFilter>
                 </ValidIf>
                 <Command
TYPE="METHOD">oracle.ias.version.SchemaVersionUtil:utilCreateRegistryTable</Comman
d>
       <Parameters>
          <Parameter TYPE="Connection"></Parameter>
       </Parameters>
     </Action>

 <Action TYPE="Java" PERCENT_PROGRESS="60">
```

```
         <ValidIf DBTYPE="MYSQL">
               <CustomQueryFilter DATA_TYPE="NUMBER" COMPARE_OPERATOR="EQ" VALUE="0">
                   select count(*) from  INFORMATION_SCHEMA.TABLES where TABLE_
NAME='SCHEMA_VERSION_REGISTRY'
               </CustomQueryFilter>
          </ValidIf>
        <Command
TYPE="METHOD">oracle.ias.version.SchemaVersionUtil:utilCreateRegistryTable</Comman
d>
        <Parameters>
            <Parameter TYPE="Connection"></Parameter>
        </Parameters>
   </Action>

    </ExecutionDescriptor>

   <ExecutionDescriptor TYPE="PostLoad">
           <Action TYPE="Java" PERCENT_PROGRESS="60">
            <ValidIf DBTYPE="EBR"/>
            <Command
TYPE="METHOD">oracle.ias.version.SchemaVersionUtil:utilSetEditionName</Command>
                 <Parameters>
               <Parameter TYPE="String">%EDITION_NAME%</Parameter>
            </Parameters>
           </Action>

       <Action TYPE="Java">
           <ValidIf DBTYPE="ORACLE,EBR"/>
           <Command
TYPE="METHOD">com.oracle.cie.rcu.ComponentSchemaUtil:loadCompSchemaInfo</Command>
           <Parameters>
               <Parameter TYPE="Connection"></Parameter>
               <Parameter TYPE="String">%SERVICE_TABLE_STRING%</Parameter>
               <Parameter TYPE="String">%PREFIX_NAME%</Parameter>
               <Parameter TYPE="String">%DB_HOSTNAME%</Parameter>
               <Parameter TYPE="String">%DB_SERVICE%</Parameter>
               <Parameter TYPE="String">%DB_PORTNUMBER%</Parameter>
           </Parameters>
       </Action>
       <Action TYPE="Java">
         <ValidIf DBTYPE="SQLSERVER,IBMDB2,MYSQL,JAVADB"/>
         <Command
TYPE="METHOD">com.oracle.cie.rcu.ComponentSchemaUtil:loadNonOracleCompSchemaInfo</
Command>
         <Parameters>
           <Parameter TYPE="Connection"></Parameter>
           <Parameter TYPE="String">%SERVICE_TABLE_STRING%</Parameter>
           <Parameter TYPE="String">%PREFIX_NAME%</Parameter>
           <Parameter TYPE="String">%DB_HOSTNAME%</Parameter>
           <Parameter TYPE="String">%DB_PORTNUMBER%</Parameter>
           <Parameter TYPE="String">%DATABASE_NAME%</Parameter>
           </Parameters>
       </Action>
    </ExecutionDescriptor>


   <FatalErrors>
     <Error Type="ORA-17439">Invalid SQL type</Error>
         <Error Type="ORA-01435">user does not exist</Error>
         <Error Type="ORA-01435">user does not exist</Error>
```

```
            <Error Type="ORA-00955">name is already used by an existing object</Error>
                <Error Type="ORA-01031">name is already used by an existing
object</Error>
        </FatalErrors>

    <IgnorableErrors>
            <Error Type="ORA-02289">sequence does not exist</Error>
            <Error Type="ORA-00904">invalid identifier</Error>
            <Error Type="ORA-01920">user name conflicts with another user or role
name</Error>
            <Error Type="ORA-01418">specified index does not exist</Error>
            <Error Type="ORA-00942">table or view does not exist</Error>
            <Error Type="ORA-06512">Not found</Error>
            <Error Type="ORA-01403">no data found</Error>
                <Error Type="ORA-04043">does not exist</Error>
                <Error Type="ORA-04080">Trigger does not exist</Error>
                <Error Type="ORA-00959">Tablespace  does not exist</Error>
                <Error Type="ORA-24035">AQ agent not subscriber</Error>
                <Error Type="ORA-24185">Transformation not found</Error>
                <Error Type="ORA-24042">Does not exist</Error>
                <Error Type="ORA-24088">Does not exist</Error>
        </IgnorableErrors>
</ComponentInfo>
```

C.2.4 Soft-Prerequisite Support

In the `ComponentInfo.xml` file, If a particular `<DBPrerequisiteSet>` or `<DBPrerequisite>` is not mandatory, then you can use the soft-prerequisite option by setting the `SOFT` attribute to `TRUE`. When a soft-prerequisite is not met, a pop-up dialog window with an error or warning message will appear; the user will have the option to ignore the message or abort the operation. You can define a soft-prerequisite at the `<DBPrerequisiteSet>` level, the `<DBPrerequisite>` level, or both; if both are defined, then `<DBPrerequisiteSet>` will take higher precedence.

Below is an example of setting a soft-prerequisite at the `<DBPrerequisite>` level:

```
<DBPrerequisiteSet>
...
    <DBPrerequisite PREREQ_TYPE="CustomSQL" DATA_TYPE="STRING" COMPARE_
OPERATOR="EQ" SOFT="TRUE">
        <PrereqIdentifier>select value from nls_database_parameters where parameter
= 'NLS_CHARACTERSET'</PrereqIdentifier>
        <PrereqValue>AL32UTF8</PrereqValue>
        <PrereqErrorMsg>
        The database you are connecting is with non-AL32UTF8 character set.
        Oracle strongly recommends using AL32UTF8 as the database character
        set.
        </PrereqErrorMsg>
    </DBPrerequisite>
...
<DBPrerequisiteSet>
```

C.2.5 Default Tablespaces Configuration File

The default tablespaces configuration file (`Storage.xml`) lists the components for which tablespaces are created out-of-the-box. This file is located in the *ORACLE_ HOME*/oracle_common/rcu/config (on UNIX operating systems) or *ORACLE_ HOME*\oracle_common\rcu\config (on Windows operating systems) directory.

The actual tablespace configuration file for each component is located in the *ORACLE_HOME*/oracle_common/rcu/integration*component*/*component*_Storage.xml (on UNIX operating systems) or *ORACLE_HOME*\oracle_common\rcu\integration*component**component*_Storage.xml (on Windows operating systems) file. Each component has its own tablespaces configuration file.

Below is a sample opss_Storage.xml file:

```
<?xml version="1.0" encoding="UTF-8" ?>

<!-- OPSS START -->
<TablespaceAttributes NAME="IAS_OPSS">
  <Type>Permanent</Type>
  <DefaultTemp>False</DefaultTemp>
  <Bigfile>False</Bigfile>
  <DatafilesList>
    <DatafileAttributes ID="%DATAFILE_LOCATION%/ias_opss.dbf">
      <Size UNIT="MB">60</Size>
      <Reuse>True</Reuse>
      <AutoExtend>True</AutoExtend>
    </DatafileAttributes>
  </DatafilesList>
</TablespaceAttributes>
<!-- OPSS END -->
```

C.3 RCU Script Writing Guidelines

Below are some common RCU script writing guidelines:

Schema user names and passwords should not be hard coded. They should be coded as substitutable variables.

- If schema user needs to be created, it should be created first using the parameters passed in by RCU.

- Tablespace and temporary tablespace references should not be hard coded; they should be coded as variables.

- Do not use CONNECT; instead, use "ALTER SESSION SET CURRENT_SCHEMA = <SCHEMA_OWNER>" after creating the schema user.

- The set of ignorable and fatal ORA errors (if any) should be listed in the RCU XML component configuration file.

- Avoid any "shutdown" or "startup" that would bounce the database instance.

- SCHEMA_VERSION_REGISTRY should be updated before and after loading schema. This can be done using JavaAction as shown in Section C.1.5, "Java Code Using JavaAction" or with in the component scripts using SCHEMA_VERSION PL/SQL package.

- Block comments that contain line comments (/* -- comment */) are not supported.

C.3.1 Guidelines for RCU JDBC Engine Compliant SQL*Plus Scripts

Below are some guidelines for writing RCU JDBC Engine SQL*Plus scripts:

- All statements must be terminated with appropriate terminating chars. CREATE PACKAGE, TYPE needs to be terminated with ";" with "/" on the next line. All other statements (Create TABLE, VIEW, etc.) need to be terminated by ";" or "/" (only one of them, not both).

- EXECUTE calls should be replaced with "BEGIN/END blocks".

- DEFINE statements should be in one line, no comments in the same line and no ";" at the end.

- SET, SHOW, SPOOL, WHENEVER, BREAK, EXIT statements are simply ignored.

- HOST command is not supported yet.

- VARIABLE and COL(UMN) are not supported.

  Dynamically calling another SQL Script within a PL/SQL block using the following technique is not supported:

```
VARIABLE initfile VARCHAR2(32)
COLUMN :initfile NEW_VALUE init_file NOPRINT;
BEGIN
  IF (some condition) THEN
     :initfile := 'initcdc.sql';
  ELSE
     :initfile := 'nothing.sql';
  END IF;
END;
/
SELECT :initfile FROM DUAL;
@@&init_file
```

  The work around is to have a separate Action with "ValidIf" tag to specify the condition.

C.3.2 Guidelines for Pure JDBC Scripts

Below are some guidelines for writing Pure JDBC scripts for RCU:

- Should not contain any SQL*Plus directives (like SET, WHENEVER, etc.).

- All DEFINES should be changed to PL/SQL variable declarations.

- All SQL statements should be wrapped in EXECUTE IMMEDIATE.

- PL/SQL style comments are allowed, But SQL*Plus style (REM) comments are not allowed.

- DROP statements preceding CREATE statements do not work. DROP should only be done after checking for the existence of the object. Ideally, all DROP statements should put into different PL/SQL script and RCU can call this script before calling a CREATE script, if that is desired.

- Contents of the script file should be a valid PL/SQL block, which can be called within `Connection.prepareCall()`.

C.3.3 Guidelines for SQL*Plus Scripts

Below are some guidelines for writing SQL*Plus scripts for RCU:

- Should not have any "exit" statements or "WHENEVER ERROR EXIT" directives. This would cause RCU SQL*Plus session to exit unexpectedly and may impact other component scripts to be executed later.

- Scripts should not have any spool commands. RCU would generate a spool log for each component.

C.3.4 Guidelines for SQL Server-Based Scripts

Below are some guidelines for writing SQL Server-based scripts for RCU:

- Support is a subset of what is supported in t-sql scripts that can be executed by `sqlcmd`.

- "ValidIf" tags should be added around all database-specific Actions and Prerequisites. For example:

```
<DBPrerequisite PREREQ_TYPE="TablespaceFreeMB" DATA_TYPE="NUMBER" COMPARE_
OPERATOR="GT">
    <ValidIf DBTYPE="ORACLE" />
    <PrereqIdentifier>%DEFAULT_TABLESPACE%</PrereqIdentifier>
    <PrereqValue>50</PrereqValue>
</DBPrerequisite>
```

- RCU supports recursive variable definitions such as:

```
setvar var1 value1
setvar var2 $(var1)
```

- There should be a "go" statement to end blocks of statements. All statements preceding the "go" statement will be executed as a single statement over JDBC.

- The JDBC connection is created in the auto-commit "on" mode.

- Currently, begin transaction and commit transaction statements are not supported.

- Variables passed to scripts via the XML file will be passed as follows:

```
Script.sql –v v1=value1 v2=value2
```

This is only for scripts called using the XML files. If a script calls another script, you can use any other variable name.

D

Troubleshooting Repository Creation Utility

This appendix contains information to help you troubleshoot any issues you might encounter while running Repository Creation Utility (RCU).

This chapter contains the following sections:

- Section D.1, "General Troubleshooting Tips"
- Section D.2, "RCU Log Files"
- Section D.3, "Need More Help?"

D.1 General Troubleshooting Tips

If you encounter an error during installation:

- Read the *Oracle Fusion Middleware Release Notes* for the latest updates. The most current version of the release notes is available on Oracle Technology Network in the Oracle Fusion Middleware Documentation page.

  Select the documentation library for your specific product release to view the release notes.

- Verify that your computer meets the requirements specified in the *Oracle Fusion Middleware System Requirements and Specifications* document.

  Select the document that is applicable for your release.

- Verify that your environment meets the certification requirements for your release and platform, as specified on the Oracle Fusion Middleware Supported System Configurations page.

- Make sure that your database is up and running.

- If you entered incorrect information on one of the screens, use the navigation pane on the left hand side of the graphical interface to return to that screen.

- If an error occurred while running RCU:

  1. Note the error and review the installation log files (see Section D.2).

  2. Correct the issue that caused the error. Depending on the type of error, you may either continue with your RCU operation, or be forced to restart RCU.

  3. Continue or restart RCU to complete your desired operation.

D.2 RCU Log Files

The main RCU log file is written to the *ORACLE_HOME*/oracle_
common/rcu/log/logdir.*date_timestamp*/rcu.log (on UNIX operating
systems) or *ORACLE_HOME*\oracle_common\rcu\log\logdir.*date_
timestamp*\rcu.log (on Windows operating systems) file. For example, on a UNIX
operating system:

```
ORACLE_HOME/oracle_common/rcu/log/logdir.2014-01-02_03-00/rcu.log
```

In addition to this general log file, each component writes a log file of its own. All
component log files are also written to the same directory as the rcu.log file.

Table D–1 lists the component log file names in alphabetical order by log file name.

Table D–1   RCU Component Log File Names

Component	Log File Name
Oracle Enterprise Scheduler	ess.log
Audit Services	iau.log
Audit Services Append	iau_append.log
Audit Services Viewer	iau_viewer.log
Metadata Services	mds.log
Managed File Transfer	mft.log
Oracle Platform Security Services	opss.log
SOA Infrastructure	soainfra.log
Common Infrastructure Services	stb.log
Call Control	ucscc.log
User Messaging Service	ucsums.log
WebLogic Services	wls.log

D.3 Need More Help?

If this appendix does not solve the problem you encountered, try looking for a solution
on My Oracle Support (formerly Oracle*MetaLink*):

```
https://support.oracle.com/
```

If you are unable to find a solution for your problem, open a service request.