Oracle® Fusion Middleware

Repository Creation Utility User's Guide

11*g* Release 1 (11.1.1)

E14259-01

May 2009

ORACLE®

Oracle Fusion Middleware Repository Creation Utility User's Guide 11*g* Release 1 (11.1.1)

E14259-01

Primary Author: Kevin Hwang

Contents

Preface

The *Oracle Fusion Middleware Repository Creation Utility User's Guide* 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 11*g* Release 1 (11.1.1) products for the first time 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 11*g* Release 1 (11.1.1) products will be installed. Users on UNIX systems need `root` access to run some scripts.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible to all users, including users that are disabled. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at `http://www.oracle.com/accessibility/`.

Accessibility of Code Examples in Documentation

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

Accessibility of Links to External Web Sites in Documentation

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

Deaf/Hard of Hearing Access to Oracle Support Services

To reach Oracle Support Services, use a telecommunications relay service (TRS) to call Oracle Support at 1.800.223.1711. An Oracle Support Services engineer will handle technical issues and provide customer support according to the Oracle service request process. Information about TRS is available at

`http://www.fcc.gov/cgb/consumerfacts/trs.html`, and a list of phone numbers is available at `http://www.fcc.gov/cgb/dro/trsphonebk.html`.

Related Documents

For additional information, see the following manuals:

- *Oracle Fusion Middleware Installation Planning Guide*
- *Oracle Fusion Middleware Administrator's Guide*
- *Oracle Fusion Middleware High Availability Guide*

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.

1

Repository Creation Utility Overview

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, "What is RCU?"
- Section 1.2, "Where Can I Get RCU?"
- Section 1.3, "RCU System Requirements"

1.1 What is RCU?

Repository Creation Utility is a graphical and CLI-based tool used to create and manage Oracle Fusion Middleware database schemas.

Note: For 11*g*R1 (11.1.1) only Oracle databases are supported. For specific Oracle database version information, refer to the system requirements document, available on Oracle Technology Network (OTN):

`http://www.oracle.com/technology/software/products/ias/files/fusion_requirements.htm`

Some key features of RCU are listed below:

- Integrate Components Using Declarative XML
- Single Stand-Alone Tool
- Custom Schemas and Tablespaces
- Global and Component Level Prerequisites
- Sharing the Database

1.1.1 Integrate 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.

1.1.2 Single Stand-Alone Tool

RCU can be run locally from the CD or remotely. In either case, both a graphical interface and command line (CLI) options are available.

1.1.2.1 Launch RCU from the CD

In situations where the application administrator is not allowed to install components in the database server, RCU can be started directly from the CD. The CD contains the extracted Oracle Client software and RCU uses SQLPLUS and other scripts and libraries from the CD to perform its operations.

When RCU is launched from the CD, log files are written to the user's `TEMP` directory.

1.1.2.2 Launch RCU Remotely

In situations where a database is not accessible locally for application administrators, RCU can be lunched against a remote database. The SQLNET client is packaged with RCU to support this operation.

1.1.2.3 Launch RCU in Silent Mode (Using the CLI)

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 Section 2.5, "Using the Repository Creation Utility CLI".

1.1.3 Custom Schemas and Tablespaces

RCU provides the flexibility to create custom schemas and tablespaces.

> **Note:** Oracle Internet Directory schema names cannot be customized. Other Identity Management schemas names, like OIF (Oracle Identity Federation), can be customized.

Users can choose to rename schemas, or change the tablespace allocation so that components can share a single or multiple tablespaces.

In addition, auxiliary schemas can be mapped to additional tablespaces.

1.1.4 Global and Component Level Prerequisites

At runtime, RCU performs checks against both global and component level prerequisites. If a prerequisite is not met, RCU may issue a warning and allow the procedure to continue (soft stop), or will notify the user that a prerequisite must be met before the operation can continue (hard stop).

1.1.5 Sharing the Database

RCU supports multiple repositories (collection of related schemas) within a single physical database. For example, you could have one repository with only one schema named `MDS` and a second repository with two schemas named `MDS` and `PORTAL`. Both of these repositories can reside on the same database.

RCU also supports the concept of prefixes, which are used to group related schemas together. For example, you could have two versions of the MDS schema in your database: a test version and a production version. You could create a "Test" prefix for your test MDS schema (`Test_MDS`) and then a "Prod" prefix for your production MDS schema (`Prod_MDS`).

The mapping between the prefixes and schemas are maintained in `schema_version_registry`.

1.2 Where Can I Get RCU?

RCU is available either on its own installation CD-ROM in the `bin` directory, or in a ZIP file on Oracle Technology Network (`http://www.oracle.com/technology/`). If you download the ZIP file, you should extract the contents to a folder on your local system, then run RCU from the `bin` directory in that folder.

Note: On Windows systems, make sure that you do not unzip the RCU `.zip` file to a directory name containing spaces.

1.3 RCU System Requirements

System and database requirement information can be found in the Oracle Fusion Middleware System Requirements document, available on Oracle Technology Network:

`http://www.oracle.com/technology/software/products/ias/files/fusion_requirements.htm`

2

Running Repository Creation Utility (RCU)

This chapter describes how to use RCU to create and drop schemas using both the graphical interface and command line interface.

The following topics are covered:

- Section 2.1, "Starting RCU"
- Section 2.2, "Using RCU with Java Access Bridge (Windows Only)"
- Section 2.3, "Creating Schemas"
- Section 2.4, "Dropping Schemas"
- Section 2.5, "Using the Repository Creation Utility CLI"

2.1 Starting RCU

To start RCU, insert the RCU CD-ROM and start RCU from the `rcuHome/bin` (on UNIX) or `rcuHome\bin` (on Windows) directory:

On UNIX:

```
./rcu
```

On Windows:

```
rcu.bat
```

You can also download a `.zip` file containing RCU from Oracle Technology Network (OTN):

```
http://www.oracle.com/technology/
```

After downloading the `.zip` file, extract the contents to a directory of your choice, and run RCU from the *`RCU_HOME`*`/bin` (on UNIX) or *`RCU_HOME`*`\bin` (on Windows) directory with the commands shown above, where *`RCU_HOME`* is the folder where RCU was unzipped, or the drive or mount point of the CD-ROM.

Note: On Windows systems, make sure that you do not extract the RCU `.zip` file to a directory name containing spaces.

Note: RCU is available only on Linux and Windows platforms. Either the Linux RCU or Windows RCU may be used to create schemas on any supported database platform.

2.2 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.

2.2.1 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.

2.2.2 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**.

2.3 Creating Schemas

RCU is used to create the various component schemas in an existing database.

2.3.1 Do all Schemas Have to Reside in the Same Database?

You can choose to create all the schemas in a single database or distribute them throughout multiple databases.

2.3.2 Are Multiple Versions of the Same Schema Supported?

You can use RCU to create multiple versions of each schema using custom prefixes (for example, you could have a test version of the Metadata Services schema called `TEST_MDS` and also a production or live version of the same schema called `PROD_MDS` on the same database).

2.3.3 What Happens When a Schema is Created?

The following sequence takes place when a schema is created with RCU:

1. Prior to the schema being created, RCU performs global and component level prerequisite checks to ensure that certain minimum requirements are met.
2. The schemas are created; the required tablespaces and data files are created.
3. The `schema_version_registry` table is updated so that the schema type is mapped to the actual schema name (for example, `Test_MDS` might be mapped to the MDS Schema type).
4. The scripts provided by the various component owners are invoked; these scripts perform the following:
   a. Create the user and grant the required roles.
   b. Run `ALTER SESSION SET CURRENT SCHEMA` to switch the schema to user context.
   c. Create the schema objects.

2.3.4 Creating Schemas

After successfully starting RCU (see Section 2.1, "Starting RCU"), 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 How to Create Schemas

No.	RCU Screen	Instructions and Action Required
1	Welcome Screen	None.
2	Create Repository Screen	Select **Create**.
3	Database Connection Details Screen	Specify the connection details for your Oracle database.
4	Select Components Screen (for Create Operation)	Specify a schema prefix and select the components for which you want to create schemas in the database. You must remember the prefix and schema names for the components you are installing; you will need this information during the configuration phase of Fusion Middleware product installation. It is recommended that you write these values down.
5	Schema Passwords Screen	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 Fusion Middleware product installation. It is recommended that you write these values down.

Table 2–1 (Cont.) How to Create Schemas

No.	RCU Screen	Instructions and Action Required
6	Map Tablespaces Screen	Configure the desired tablespace mapping for the schemas you want to create.
7	Summary Screen (for Create Operation)	Review the information on this screen, then click **Create** to begin schema creation.
8	Completion Summary Screen (for Create Operation)	Note the location of the log files, then click **Close** to dismiss the screen.

2.4 Dropping Schemas

To drop schemas from the database, start RCU (see Section 2.1, "Starting RCU"), then follow the instructions in Table 2–2.

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 2–2 How to Drop Schemas

No.	Screen	Instructions and Action Required
1	Welcome Screen	None.
2	Create Repository Screen	Select **Drop**.
3	Database Connection Details Screen	Specify the connection details for your database.
4	Select Components Screen (for Drop Operation)	Select the prefix and the schemas you want to drop.
5	Summary Screen (for Drop Operation)	Review the information on this screen, then click **Drop** to drop the schemas.
6	Completion Summary Screen (for Drop Operation)	Note the location of the log files, then click **Close** to dismiss the screen.

2.5 Using the Repository Creation Utility CLI

This section describes how to use the Repository Creation Utility's (RCU) command line interface (CLI). The 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 section contains the following topics:

- Section 2.5.1, "CLI Syntax and Parameters"
- Section 2.5.2, "RCU and Password Handling"
- Section 2.5.3, "Creating a Repository"
- Section 2.5.4, "Dropping a Repository"
- Section 2.5.5, "RCU Environment Variables"

2.5.1 CLI Syntax and Parameters

The syntax for the RCU command line interface is:

```
rcu [-silent | -interactive] {<command> <options>}
```

Table 2–3 describes the various command line options.

Table 2–3 RCU Command Line Interface Options and Descriptions

Option	Description
`-silent` or `-interactive`	Specify `-silent` if you want to run RCU with minimal interaction once you have entered the commend. You must specify all necessary command line parameters. Specify `-interactive` if you want to run RCU and be prompted for each parameter.
command	One of the following: ■ `-createRepository` Use this command to create a repository. For more information, see Section 2.5.3, "Creating a Repository". ■ `-dropRepository` Use this command to drop a repository. For more information, see Section 2.5.4, "Dropping a Repository"

2.5.2 RCU and Password Handling

If you use the `-silent` flag, RCU will prompt you for the database and schema passwords. If you use the `-randomizePasswords` parameter and "false" is passed as the value, then RCU will prompt for the schema passwords. If they are not entered, RCU will generate random passwords and use those.

2.5.3 Creating a Repository

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

```
rcu [-silent | -interactive] -createRepository
   [-compInfoXMLLocation <location of ComponentInfo.xml file>]
   [-storageXMLLocation <location of Storage.xml file>]
   [-databaseType ORACLE]
   -connectString <database connect string (for example: host:port:service_id)>
   -dbUser <database username>
   [-dbRole <database role>]
   [-variables <comma separated variables in the format: variablename=value>]
   [-lockSchemas <true|false>]
   [-randomizePasswords <true|false>]
   [-schemaPrefix <schema prefix (optional for non-prefixable components)>]
   -component <component name>
]
```

> **Note:** When loading 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, RCU will stop before any loading takes place.

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.

2.5.4 Dropping a Repository

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

```
rcu [-silent | -interactive] -dropRepository
   [-compInfoXMLLocation <location of ComponentInfo.xml file>]
   [-storageXMLLocation <location of Storage.xml file>]
   [-databaseType ORACLE]
   -connectString <database connect string (for example: host:port:service_id)>
   -dbUser <database username>
   [-dbRole <database role>]
   [-variables <comma separated variables in the format: variablename=value>]
   [-schemaPrefix <schema prefix (optional for non-prefixable components)>]
   -component <component 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.

2.5.5 RCU Environment Variables

Table 2–4 shows the variables picked up by RCU from the environment. If the environment variable is not set, then RCU uses the default value.

Table 2–4 RCU Environment Variables

Variable	Default	Description
`RCU_LOG_LOCATION`	*`ORACLE_HOME`*`/rcu/log` (UNIX) *`ORACLE_HOME`*`\rcu\log` (Windows)	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`.

3

Extending RCU to Configure Custom Application Repositories

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 chapter:

- Section 3.1, "RCU Integration Options"
- Section 3.2, "RCU Configuration Files"
- Section 3.3, "RCU Script Writing Guidelines"

3.1 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.

3.1.1 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, it is recommended that the RCU JDBCEngine tool is also used in your development environment to ensure that these scripts run properly when integrated with RCU.

3.1.2 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>
```

3.1.3 SQL*Plus Scripts

This option is mainly for the consumption of legacy components that need to be loaded from RCU. This option is available only when running RCU from the CD or standalone shiphome. RCU will use Oracle client on the CD or database server. Any 11*g* component that is expected to be loaded by launching RCU from the Oracle Home should not use this option.

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>
```

3.1.4 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>
```

3.1.5 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>
```

3.2 RCU Configuration Files

RCU provides the following configuration files types for component integration:

- Section 3.2.1, "XML DTDs Defined by RCU"
- Section 3.2.2, "Component Repository Configuration File"
- Section 3.2.3, "Component List Configuration File"
- Section 3.2.4, "Soft-Prerequisite Support"
- Section 3.2.5, "Default Tablespaces Configuration File"

3.2.1 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

3.2.1.1 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 *`RCU_HOME`*`/rcu/config` (on UNIX) or *`RCU_HOME`*`\rcu\config` (on Windows) 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?)>
<!ATTLIST Component
        ID CDATA #REQUIRED
```

```
        PROGRESS_UNITS CDATA #IMPLIED
        IS_GROUPING_COMPONENT  (TRUE|FALSE) "FALSE"
        DEFAULT_SELECTED (TRUE|FALSE) "FALSE"
        CHILD_OF CDATA #IMPLIED >
<!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) 'ORACLE'>
<!ELEMENT SchemaVersion (#PCDATA)>
<!ELEMENT SchemaUser (#PCDATA)>
<!ATTLIST SchemaUser
        USER_EDITABLE (TRUE|FALSE) "TRUE"
        PREFIXABLE (TRUE|FALSE) "TRUE"
        IS_CREATED  (TRUE|FALSE) "TRUE">
<!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) "DEFAULT_TABLESPACE">
<!ELEMENT Tablespace (Prompt, TablespaceName)>
<!ELEMENT Prompt (#PCDATA)>
<!ATTLIST Prompt NLS_ID CDATA #IMPLIED>
<!ELEMENT TablespaceName (#PCDATA)>
```

3.2.1.2 Repository Configuration File

The Repository configuration file is called `RepositoryConfig.dtd` and is located in the *`RCU_HOME`*`/rcu/config` (on UNIX) or *`RCU_HOME`*`\rcu\config` (on Windows) 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*)>
```

3.2.1.3 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 *`RCU_HOME`*`/rcu/config` (on UNIX) or *`RCU_HOME`*`\rcu\config` (on Windows) directory:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!ELEMENT PrerequisiteDescriptor (DBPrerequisiteSet*, DBPrerequisite*)>
<!ATTLIST PrerequisiteDescriptor
        TYPE (CREATE|DROP|REGISTER|DEREGISTER) '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) "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) "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"
        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
          OPERATOR (OR|AND) "OR">
```

```
<!ELEMENT ValidIf (CustomQueryFilter?)>
<!ATTLIST ValidIf
        DBTYPE CDATA #IMPLIED
        DBVERSION CDATA #IMPLIED
        OSNAME CDATA #IMPLIED >
<!ELEMENT CustomQueryFilter (#PCDATA)>
<!ATTLIST CustomQueryFilter
        DATA_TYPE (STRING|NUMBER) "STRING"
        COMPARE_OPERATOR (EQ|GT|LT|NE|GE|LE|COMPARE_VERSION) "EQ"
        VALUE CDATA #REQUIRED >
```

3.2.1.4 Storage Attributes Configuration File

RCU would maintain 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 *RCU_HOME*`/rcu/config` (on UNIX) or *RCU_HOME*`\rcu\config` (on Windows) directory:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!ELEMENT StorageAttributes (TablespaceAttributes*)>
<!ELEMENT TablespaceAttributes (Type?,DefaultTemp?, BlockSize?, Bigfile?,
AutoSegmentSpaceManagement?, DatafilesList)>
<!ATTLIST TablespaceAttributes
        NAME CDATA #REQUIRED>
<!ELEMENT Type (#PCDATA)>
<!ELEMENT DefaultTemp (#PCDATA)>
<!ELEMENT BlockSize (#PCDATA)>
<!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'>
```

3.2.2 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 Component List Configuration File (`ComponentInfo.xml`).

Each *<component>*`.xml` file can be found in the *RCU_HOME*`/rcu/integration`*component*`/`*component*`.xml` (on UNIX) or *RCU_HOME*`\rcu\integration`*component*`\`*component*`.xml` (on Windows) 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 3–1 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 *RCU_HOME*.
%SCHEMA_USER%	Database schema name (owner) entered by the user in RCU.
%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 Screen.
%DBADMIN_PASSWORD%	Database admin user password that is provided on the Database Connection Details Screen.
%DBADMIN_ROLE%	Database admin user role that is provided on the Database Connection Details Screen.
%DB_HOSTNAME%	Database hostname that is provided on the Database Connection Details Screen.
%DB_SERVICE%	Database service name.
%DB_PORTNUMBER%	Database port number that is provided on the Database Connection Details Screen.
%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 MDS (`mds.xml`), which lists the series of prerequisites and actions:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!-- DOCTYPE RepositoryConfig SYSTEM
```

```
"file:////home/mmehta/development/XML/latest/RepositoryConfig.dtd" -->
<!DOCTYPE RepositoryConfig SYSTEM "RepositoryConfig.dtd">
<RepositoryConfig COMP_ID="MDS">
   <PrerequisiteDescriptor>
         <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>20</PrereqValue>
         </DBPrerequisite>
   </PrerequisiteDescriptor>

   <PrerequisiteDescriptor TYPE="DROP">
       <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>
   </PrerequisiteDescriptor>

   <ExecutionDescriptor>
     <Action TYPE="Java">
         <Command
TYPE="METHOD">oracle.ias.version.SchemaVersionUtil:utilCreateRegistryEntry</Comman
d>
         <Parameters>
            <Parameter TYPE="Connection"></Parameter>
            <Parameter TYPE="String">MDS</Parameter>
            <Parameter TYPE="String">Metadata Services</Parameter>
            <Parameter TYPE="String">%PREFIX_NAME%</Parameter>
            <Parameter TYPE="String">MDS</Parameter>
            <Parameter TYPE="String">MDS</Parameter>
            <Parameter TYPE="String">%SCHEMA_USER%</Parameter>
            <Parameter TYPE="String">11.1.1.1.0</Parameter>
            <Parameter TYPE="String">LOADING</Parameter>
         </Parameters>
      </Action>
      <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>
      <Action TYPE="JDBCSqlScript" PERCENT_PROGRESS="20">
         <ValidIf DBTYPE="SQLSERVER" />
```

```
         <Command TYPE="SCRIPT">%SCRIPT_
HOME%/mds/MSSQL/cremduser-rcu.sql</Command>
         <Parameters>
            <Parameter TYPE="CmdLine">%DATABASE_NAME%</Parameter>
            <Parameter TYPE="CmdLine">%SCHEMA_USER%</Parameter>
            <Parameter TYPE="CmdLine">%SCHEMA_PASSWORD%</Parameter>
         </Parameters>
      </Action>
      <Action TYPE="JDBCSqlScript" PERCENT_PROGRESS="20">
 <ValidIf DBTYPE="SQLSERVER" />
         <Command TYPE="SCRIPT">%SCRIPT_
HOME%/mds/MSSQL/cremduser-rcu.sql</Command>
         <Parameters>
            <Parameter TYPE="CmdLine">%DATABASE_NAME%</Parameter>
            <Parameter TYPE="CmdLine">%SCHEMA_USER%</Parameter>
            <Parameter TYPE="CmdLine">%SCHEMA_PASSWORD%</Parameter>
         </Parameters>
      </Action>
      <Action TYPE="JDBCSqlScript" PERCENT_PROGRESS="20">
         <ValidIf DBTYPE="ORACLE" />
         <Command TYPE="SCRIPT">%SCRIPT_HOME%/mds/sql/cremds-rcu.sql</Command>
         <Parameters>
            <Parameter TYPE="CmdLine">%SCHEMA_USER%</Parameter>
         </Parameters>
      </Action>
      <Action TYPE="JDBCSqlScript" PERCENT_PROGRESS="20" CONNECT_AS_OWNER="TRUE">
         <ValidIf DBTYPE="SQLSERVER" />
         <Command TYPE="SCRIPT">%SCRIPT_HOME%/mds/MSSQL/cremds-rcu.sql</Command>
         <Parameters>
            <Parameter TYPE="CmdLine">%DATABASE_NAME%</Parameter>
            <Parameter TYPE="CmdLine">%MDS_VARCHAR%</Parameter>
         </Parameters>
      </Action>
      <Action TYPE="Java">
         <Command
TYPE="METHOD">oracle.ias.version.SchemaVersionUtil:utilSetComponentValid</Command>
         <Parameters>
            <Parameter TYPE="String">MDS</Parameter>
         </Parameters>
      </Action>
   </ExecutionDescriptor>
   <DeleteDescriptor>
      <Action TYPE="JDBC" PERCENT_PROGRESS="20">
         <ValidIf DBTYPE="ORACLE" />
         <Command TYPE="INLINE">DROP USER %SCHEMA_USER% CASCADE</Command>
      </Action>
      <Action TYPE="JDBCSqlScript" PERCENT_PROGRESS="20">
         <ValidIf DBTYPE="SQLSERVER" />
         <Command TYPE="SCRIPT">%SCRIPT_
HOME%/mds/MSSQL/dropmduser-rcu.sql</Command>
         <Parameters>
            <Parameter TYPE="CmdLine">%DATABASE_NAME%</Parameter>
            <Parameter TYPE="CmdLine">%SCHEMA_USER%</Parameter>
         </Parameters>
      </Action>
      <Action TYPE="Java">
         <Command
TYPE="METHOD">oracle.ias.version.SchemaVersionUtil:utilDropRegistryEntry</Command>
         <Parameters>
            <Parameter TYPE="Connection"></Parameter>
```

```
            <Parameter TYPE="String">MDS</Parameter>
            <Parameter TYPE="String">%PREFIX_NAME%</Parameter>
            <Parameter TYPE="String">MDS</Parameter>
          </Parameters>
       </Action>
    </DeleteDescriptor>
</RepositoryConfig>
```

3.2.3 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 *RCU_HOME*`/rcu/config` (on UNIX) or *RCU_HOME*`\rcu\config` (on Windows) 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" [
<!ENTITY portlet SYSTEM "../integration/portlet/portlet_ComponentInfo.xml">
<!ENTITY mds SYSTEM "../integration/mds/mds_ComponentInfo.xml">
<!ENTITY oid SYSTEM "../integration/oid/oid_ComponentInfo.xml">
<!ENTITY soainfra SYSTEM "../integration/soainfra/soainfra_ComponentInfo.xml">
<!ENTITY bam SYSTEM "../integration/bam/bam_ComponentInfo.xml">
<!ENTITY webcenter SYSTEM "../integration/webcenter/webcenter_ComponentInfo.xml">
<!ENTITY jive SYSTEM "../integration/jive/jive_ComponentInfo.xml">
<!ENTITY wiki SYSTEM "../integration/wiki/wiki_ComponentInfo.xml">
<!ENTITY iau SYSTEM "../integration/iau/iau_ComponentInfo.xml">
<!ENTITY discoverer SYSTEM "../integration/dc/discoverer_ComponentInfo.xml">
<!ENTITY sdpm SYSTEM "../integration/sdpm/sdpm_ComponentInfo.xml">
<!ENTITY portal SYSTEM "../integration/portal/portal_ComponentInfo.xml">
<!ENTITY contentserver SYSTEM "../integration/contentserver/contentserver_
ComponentInfo.xml">
<!ENTITY oif SYSTEM "../integration/oif/oif_ComponentInfo.xml">
<!ENTITY ess SYSTEM "../integration/ess/ess_ComponentInfo.xml">
<!ENTITY commspresence SYSTEM "../integration/commspresence/commspresence_
ComponentInfo.xml">
<!ENTITY commssds SYSTEM "../integration/commssds/commssds_ComponentInfo.xml">
<!ENTITY commsls SYSTEM "../integration/commsls/commsls_ComponentInfo.xml">
]>
<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>
  </PrefixSettings>


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

```
&ess;
<!-- AS Common GROUP END -->


<!-- OID GROUP START -->
<Component ID="IDM" IS_GROUPING_COMPONENT="TRUE">
        <ValidIf DBTYPE="ORACLE" />
      <Display NLS_ID="IDM_ID">Identity Management</Display>
    </Component>
&oid;
&oif;

<!-- OID GROUP END -->


<!-- OWLCS START -->
   <Component ID="OWLCS" IS_GROUPING_COMPONENT="TRUE">
       <Display NLS_ID="OWLCS_ID">WebLogic Communication Services</Display>
   </Component>
&commspresence;
&commssds;
&commsls;
<!-- OWLCS END -->


<!-- SOA INFRA GROUP START -->
        <Component ID="SOA" IS_GROUPING_COMPONENT="TRUE">
      <Display NLS_ID="SOA_ID">SOA Infrastructure</Display>
    </Component>
&soainfra;
&bam;
&sdpm;
<!-- SOA INFRA GROUP END -->


<!-- WEBCENTER_SUITE START -->
        <Component ID="WEBCENTER_SUITE" IS_GROUPING_COMPONENT="TRUE">
      <Display NLS_ID="WEBCENTER_SUITE_ID">Webcenter Suite</Display>
    </Component>
&webcenter;
&portlet;
&contentserver;
&jive;
&wiki;
<!-- WEBCENTER_SUITE END -->


<!-- PORTAL_BI START -->
        <Component ID="PORTAL_BI" IS_GROUPING_COMPONENT="TRUE">
        <ValidIf DBTYPE="ORACLE" />
      <Display NLS_ID="PORTAL_BI_ID">Portal and BI</Display>
    </Component>

&portal;
&discoverer;
<!-- PORTAL_BI END -->

 <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>

         <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>

    <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. The database version should be either 10.2.0.4.0 or 11.1.0.7.0 or
greater.
                  </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>
   CU_HOME/rcu/config (on UNIX)

   <ExecutionDescriptor TYPE="PreLoad">
         <Action TYPE="Java" PERCENT_PROGRESS="60">
         <ValidIf DBTYPE="ORACLE">
             <CustomQueryFilter DATA_TYPE="NUMBER" COMPARE_OPERATOR="EQ"
VALUE="1">
               select count(*) from  dba_views where VIEW_NAME = 'APP_REGISTRY'
and not exists (select view_name from  dba_views where VIEW_NAME= 'SCHEMA_VERSION_
REGISTRY')
             </CustomQueryFilter>
         </ValidIf>
         <Command
TYPE="METHOD">oracle.ias.version.SchemaVersionUtil:utilCreateRegistryAndCopyData</
Command>
         <Parameters>
            <Parameter TYPE="Connection"></Parameter>
         </Parameters>
     </Action>
         <Action TYPE="Java" PERCENT_PROGRESS="60">
         <ValidIf DBTYPE="ORACLE">
             <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>
   </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>
```

3.2.4 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>` leve, 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>
```

3.2.5 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 *`RCU_HOME`*`/rcu/config` (on UNIX) or *`RCU_HOME`*`\rcu\config` (on Windows) directory.

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

Below is a sample `soainfra_Storage.xml` file:

```
<?xml version="1.0" encoding="UTF-8"?>
 <!-- SOAINFRA -->
<TablespaceAttributes NAME="SOAINFRA" >
   <DatafilesList>
      <DatafileAttributes ID="%DATAFILE_LOCATION%/soainfra.dbf">
         <Size UNIT="MB">200</Size>
         <Reuse>True</Reuse>
         <AutoExtend>True</AutoExtend>
         <Increment UNIT="MB">50</Increment>
         <Maxsize UNIT="GB">2</Maxsize>
      </DatafileAttributes>
   </DatafilesList>
</TablespaceAttributes>

<!-- End 0f SOAINFRA -->
```

3.3 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 3.1.5, "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.

3.3.1 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 workaround is to have a separate Action with "ValidIf" tag to specify the condition.

3.3.2 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()`.

3.3.3 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.

3.3.4 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.

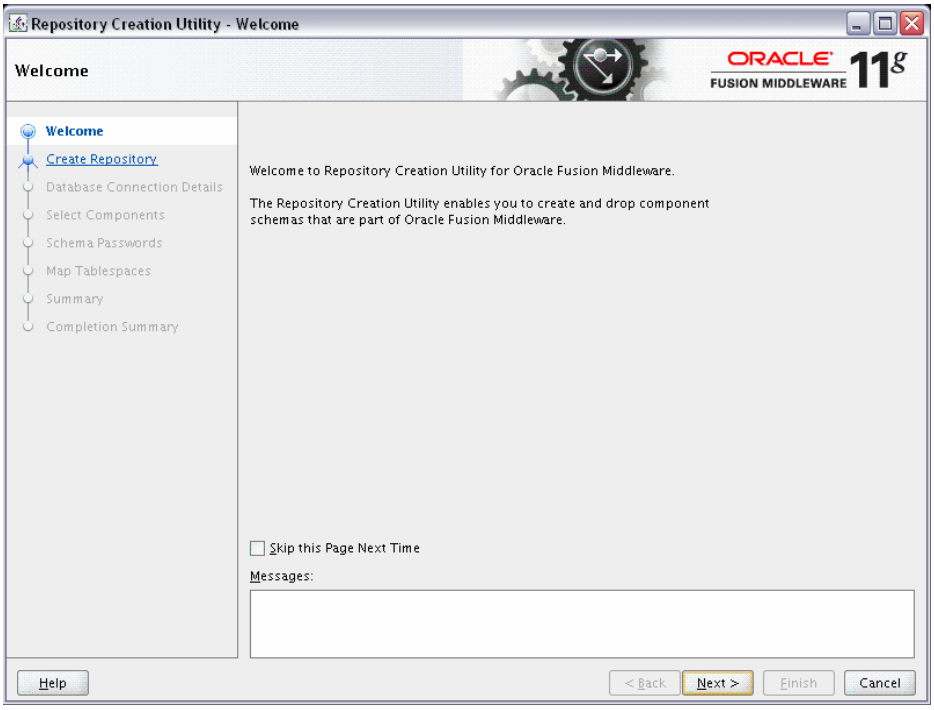
A

Repository Creation Utility Screens

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

- Welcome Screen
- Create Repository Screen
- Drop Repository Screen
- Database Connection Details Screen
- Select Components Screen (for Create Operation)
- Select Components Screen (for Drop Operation)
- Schema Passwords Screen
- Map Tablespaces Screen
- Summary Screen (for Create Operation)
- Summary Screen (for Drop Operation)
- Completion Summary Screen (for Create Operation)
- Completion Summary Screen (for Drop Operation)

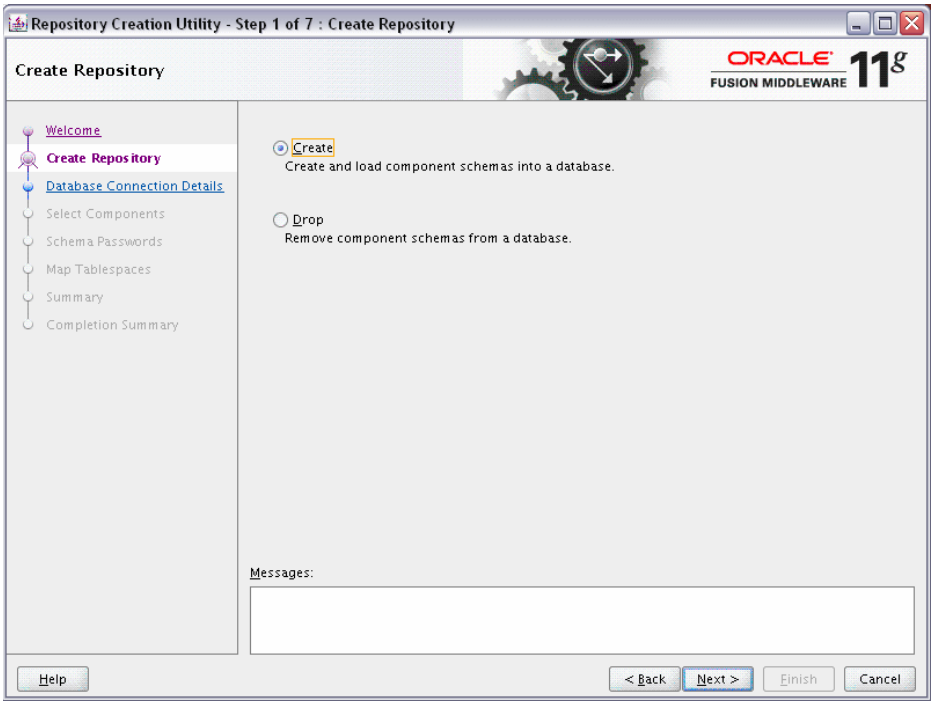
A.1 Welcome Screen



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

Click **Next** to continue.

A.2 Create Repository Screen

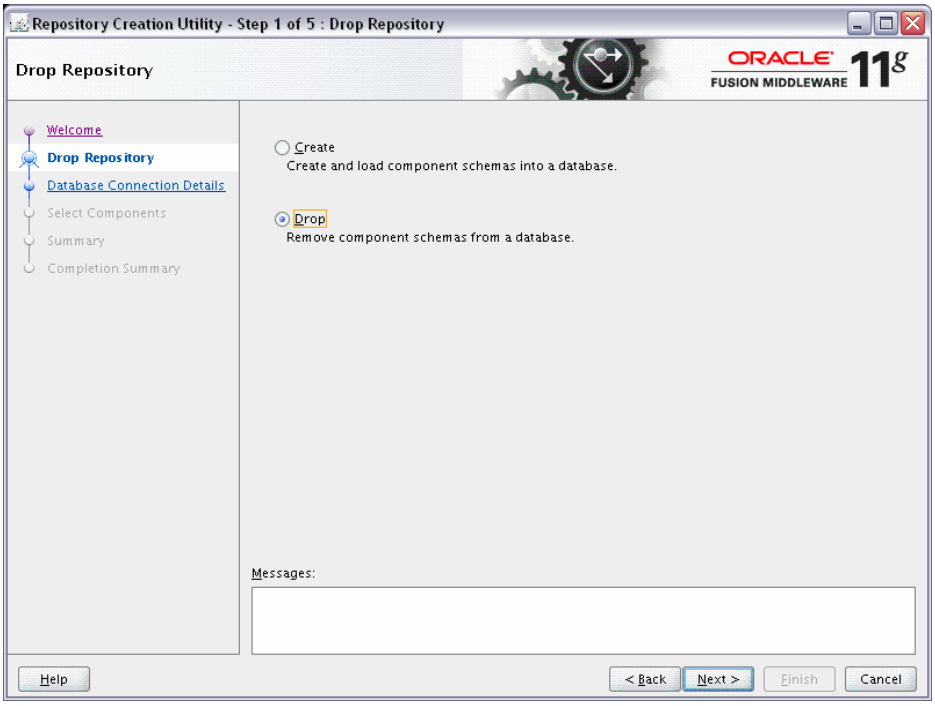


Select **Create** to create component schemas in the database.

Select **Drop** to remove component schemas from the database.

Click **Next** to continue.

A.3 Drop Repository Screen

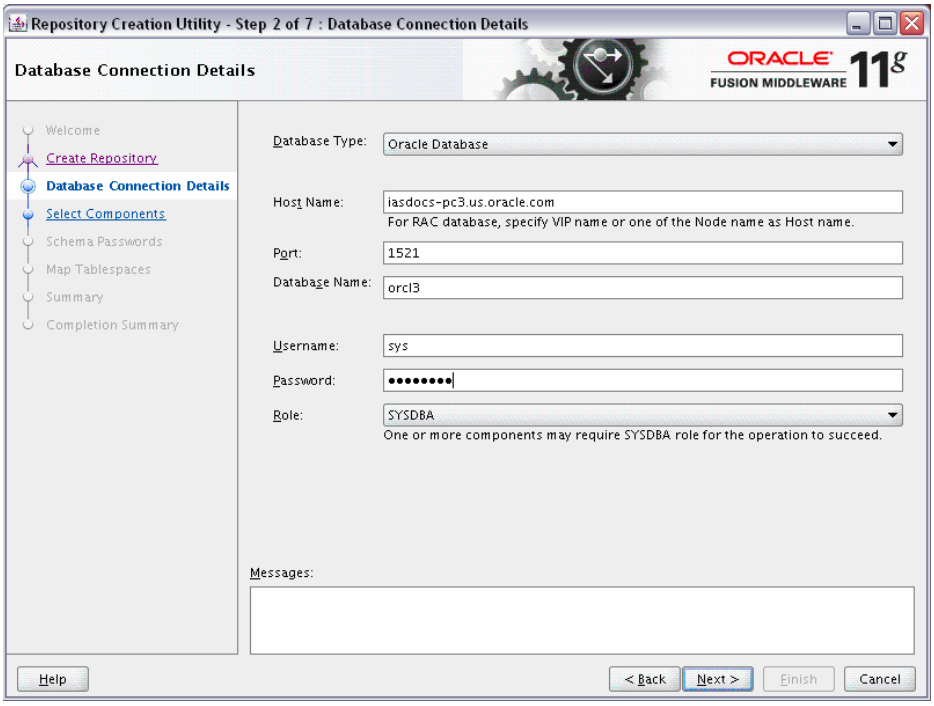


Select **Create** to create component schemas in the database.

Select **Drop** to remove component schemas from the database.

Click **Next** to continue.

A.4 Database Connection Details Screen



Specify the connection details for your Oracle database.

- Host Name

  Enter the name of the server where your database is running. Use the following format:

  ```
  myhost.mydomain.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.

- Database 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:

  ```
  sales.mydomain.com
  ```

- Username

  Enter the user name for your database. The default user name is `SYS`.

- Password

  Enter the password for your database user.

- Role

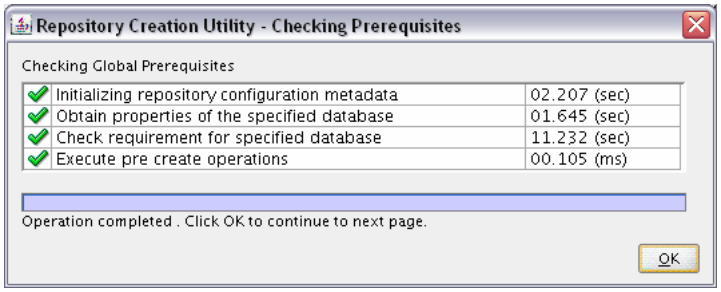
  Select the database user's role from the drop-down list:

  - Normal
  - SYSDBA

  The `SYS` user reqiures the SYSDBA role. All other users would use the Normal role.

Click **Next** when you are finished.

The following screen appears, indicating the progress of the installer establishing the connection with the specified database.

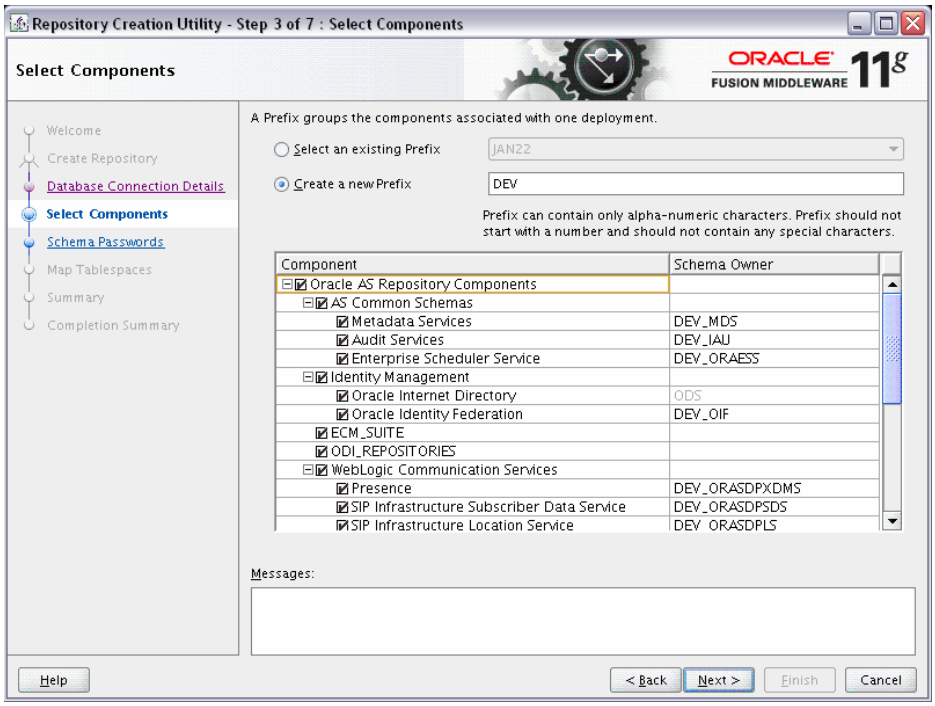


If an error occurs while the connection is being established, the error message(s) appear in the Messages field on the Database Connection Details screen.

Click **OK** to dismiss this screen.

A.5 Select Components Screen (for Create Operation)

Below is the Select Components screen if you selected **Create** on the Create Repository Screen.



The following topics are covered in this section:

- Section A.5.1, "Create Prefixes"
- Section A.5.2, "Select Components and Dependencies"
- Section A.5.3, "Specify Custom Schema Names"
- Section A.5.4, "Check Schema Prerequisites"

> **Note:** You must remember the prefix and schema names for the components you are installing; you will need this information during the configuration phase of Fusion Middleware product installation. It is recommended that you write these values down.

A.5.1 Create Prefixes

Prefixes are used to create logical groupings of multiple repositories in a database. For example, if you want to create two repositories for MDS in the database, you can use different prefixes to uniquely identify each one (for example, `TEST_MDS` and `PROD_MDS`).

> **Note:** Some components (for example, Identity Management) cannot be prepended with a custom prefix; there can only be one repository for that component per database.

If you want to create a new prefix for your schemas, select **Create a New Prefix** and specify a new prefix name in the field. The prefix name must be a minimum of 3 alphanumeric characters (0-9, a-z, or A-Z) in length and cannot exceed 12 characters in length. No whitespace or special characters are allowed.

The default new prefix is `DEV`. If `DEV` already exists as a prefix, then `DEV1` is used; if `DEV1` exists, then `DEV2` is the default, and so on.

Use existing prefixes to add additional components to existing schemas in the database. To use an existing prefix, select **Select an Existing Prefix** and choose a prefix from the drop-down list.

A.5.2 Select Components and Dependencies

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 **SOA Infrastructure**, then all schemas in this category are selected along with the **Metadata Services** schema. The **Metadata Services** schema is required by each component in the **SOA Infrastructure** 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 (+).

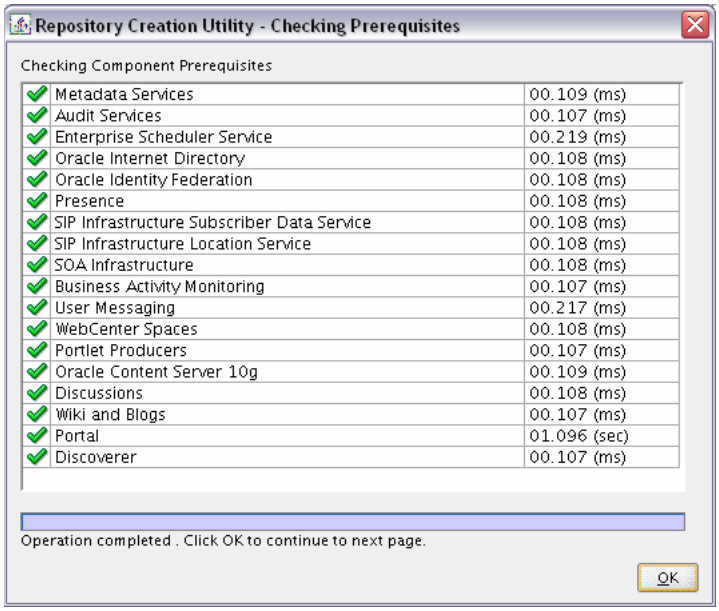
A.5.3 Specify Custom Schema Names

Click on the name of any schema in the "Schema Owner" column to change the name of the schema. Schema names can only contain alphanumeric characters (0-9, a-z, or A-Z) and are case-sensitive.

Note: The schema names for Identity Management can not be changed.

A.5.4 Check Schema Prerequisites

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.

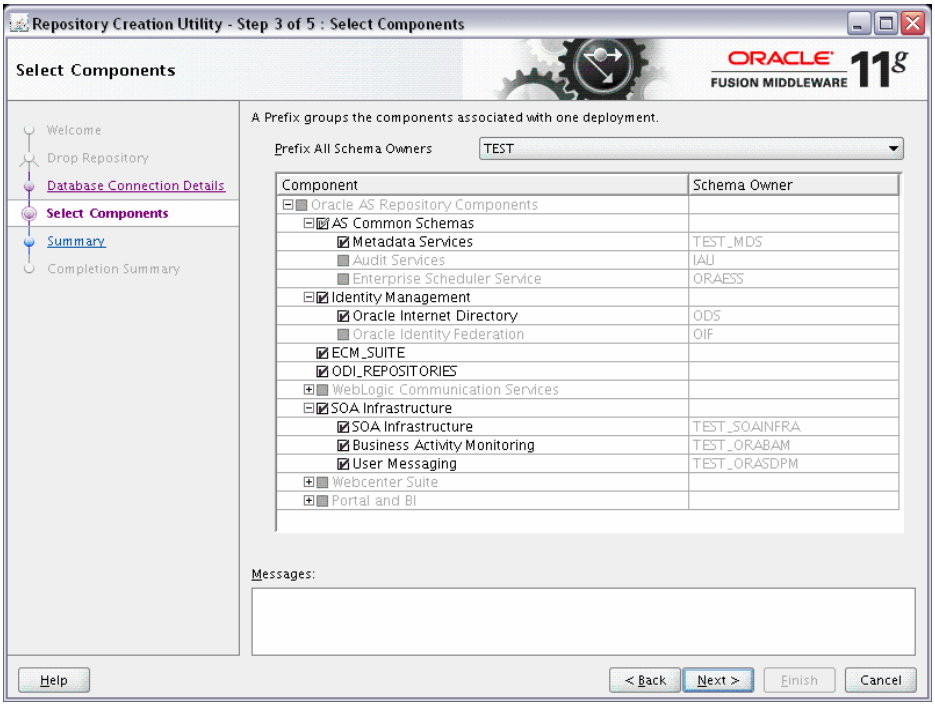


If an error occurs during the prerequisite checking, the error message(s) appear in the Messages field on the Select Components screen.

Click **OK** to dismiss this screen.

A.6 Select Components Screen (for Drop Operation)

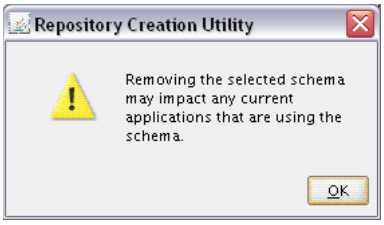
Below is the Select Components screen if you selected **Drop** on the Create Repository Screen.



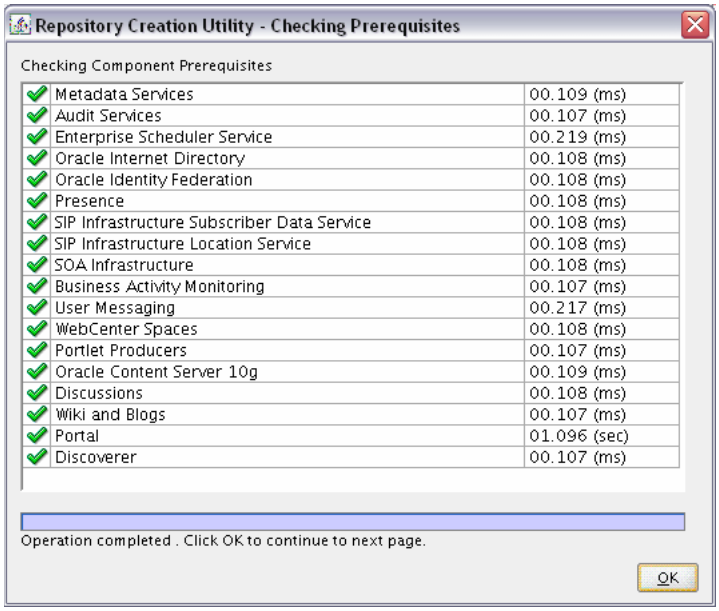
First, select the prefix associated with the schema(s) you want to drop.

Then, select the component(s) whose schemas you want to drop.

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



Click **OK** to continue. The following screen appears:

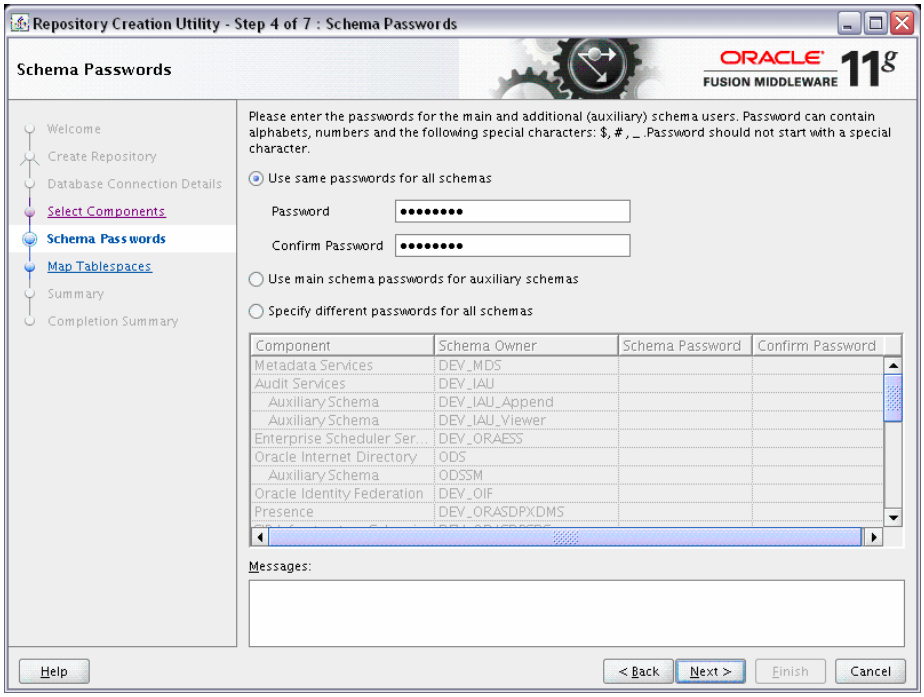


If an error occurs during the prerequisite checking, the error message(s) appear in the Messages field on the Select Components screen.

Click **OK** to dismiss this screen.

A.7 Schema Passwords Screen

Below is the Schema Passwords screen.



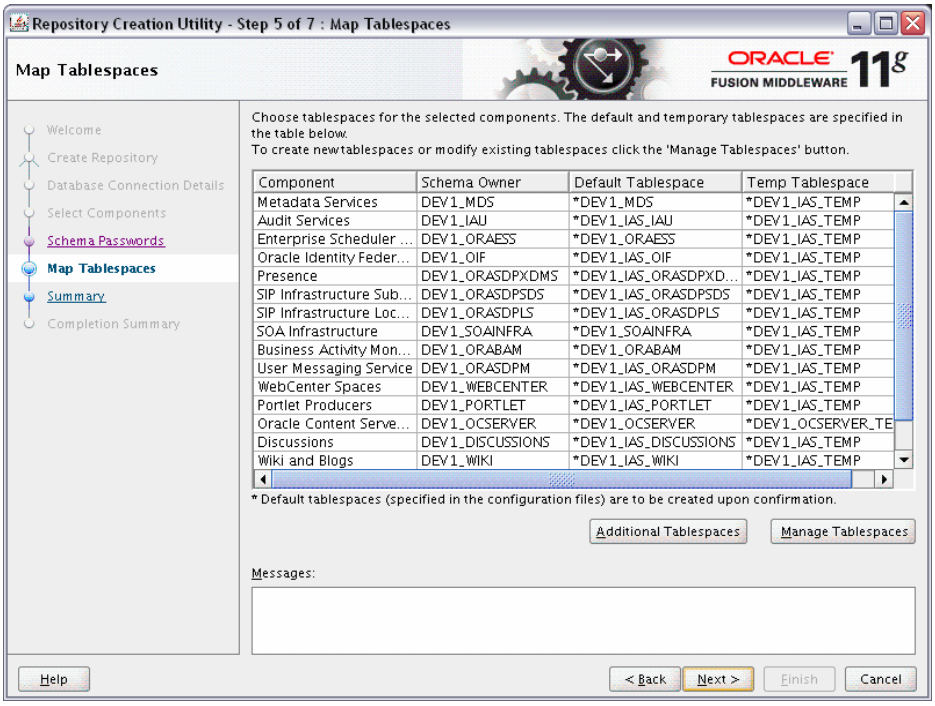
There are three ways to specify schema passwords on this screen:

- Select **Use same password for all schemas** 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.
- Select **Use main schema passwords for auxiliary schemas** 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 Password column in the table, and enter the same password in the Confirm Password column.
- Select **Specify different passwords for all schemas** 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 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 Fusion Middleware product installation. It is recommended that you write these values down.

Click **Next** when you are finished.

A.8 Map Tablespaces Screen



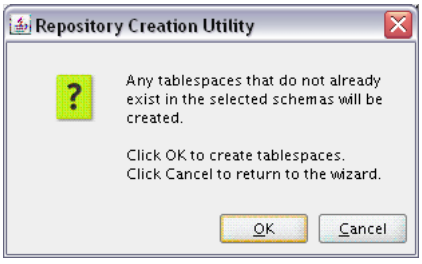
RCU Map Tablespaces screen. This screen is described in the surrounding text.

****

This screen only appears if you selected the **Create** option on the Create Repository Screen. The following topics are covered:

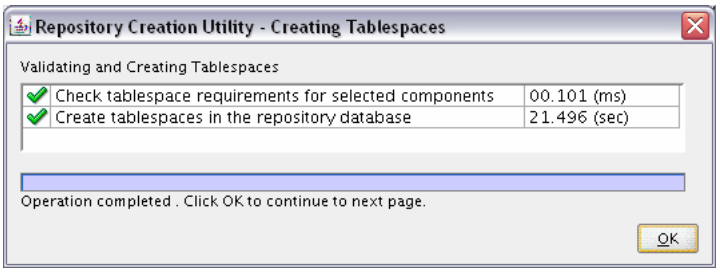
- Section A.8.1, "Default Tablespace Mappings"
- Section A.8.2, "Changing Default and Temporary Tablespaces"
- Section A.8.3, "Viewing and Changing Additional Tablespaces"
- Section A.8.4, "Managing Tablespaces and Datafiles"

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 is shown in Table A–1.

Table A–1 Default Tablespace Mapping

Component	Schema Owner	Default tablespace	Temp Tablespace
AS Common Schemas			
Metadata Services	MDS	MDS	IAS_TEMP
Audit Services	IAU	IAS_IAU	IAS_TEMP
Enterprise Scheduler Service	ORAESS	ORAESS	IAS_TEMP
Identity Management Schemas			
Oracle Internet Directory	ODS	OLTS_DEFAULT	IAS_TEMP
Oracle Identity Federation	OIF	IAS_OIF	IAS_TEMP
WebLogic Communication Services			
Presence	ORASDPXDMS	IAS_ORASDPXDMS	IAS_TEMP
SIP Infrastructure Subscriber Data Service	ORASDPSDS	IAS_ORASDPSDS	IAS_TEMP
SIP Infrastructure Location Service	ORASDPLS	IAS_ORASDPLS	IAS_TEMP
SOA Infrastructure Schemas			
SOA Infrastructure	SOAINFRA	SOAINFRA	IAS_TEMP
Business Activity Monitoring	ORABAM	ORABAM	IAS_TEMP
User Messaging	ORASDPM	IAS_ORASDPM	IAS_TEMP

Table A–1 (Cont.) Default Tablespace Mapping

Component	Schema Owner	Default tablespace	Temp Tablespace
WebCenter Suite Schemas			
Portlet Producers	PORTLET	IAS_PORTLET	IAS_TEMP
WebCenter Spaces	WEBCENTER	IAS_WEBCENTER	IAS_TEMP
Oracle Content Server 10*g*	OCSERVER	OCSERVER	OCSERVER_TEMP
Discussions	DISCUSSIONS	IAS_DISCUSSIONS	IAS_TEMP
Wiki and Blog	WIKI	IAS_WIKI	IAS_TEMP
Portal and Business Intelligence Schemas			
Discoverer	DISCOVERER	DISCO_PTM5_META	IAS_TEMP
Portal	PORTAL	PORTAL	IAS_TEMP

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.

Note: OID tablespace names cannot be user specified.

A.8.2 Changing Default and Temporary Tablespaces

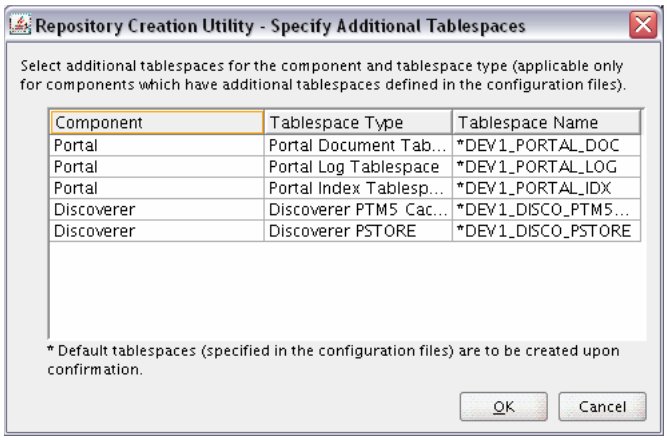
To change the default tablespace for a component, select the tablespace name in the "Default Tablespace" column, then select the tablespace name you want to use from the drop-down list. You can have your components use as many or as few tablespaces as desired to suit your configuration.

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. You will see a screen similar to the following:



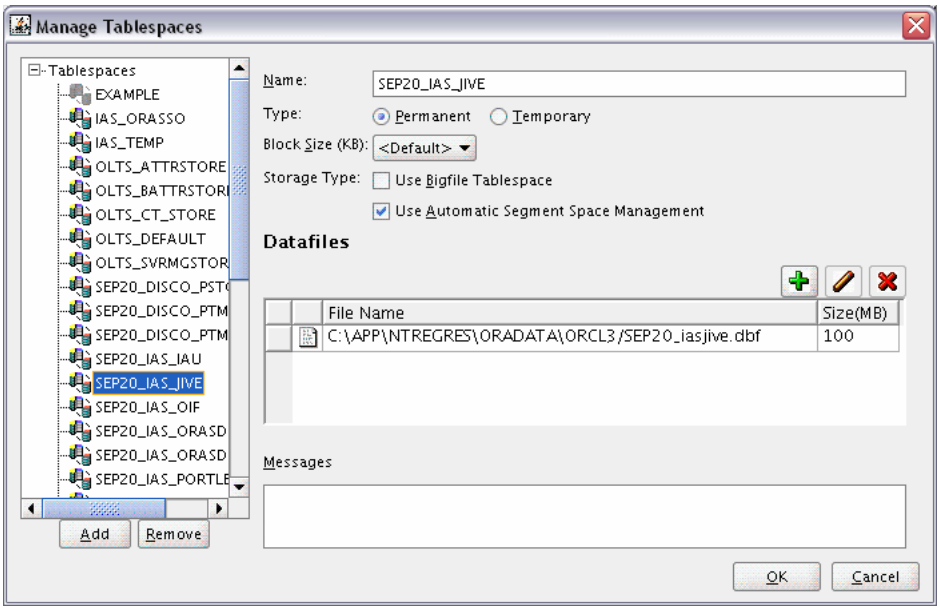
Only those components with additional tablespaces as defined in the configuration files will appear on this screen.

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 following fields as necessary:

- 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.

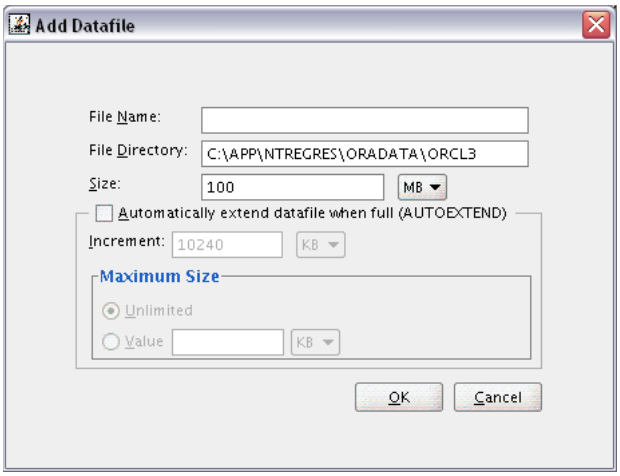
A.8.4.2 Adding, Modifying, and Removing Datafiles

In the Datafiles section, specify the datafiles that make up the selected tablespace. Select one of the following for more information:

- Section A.8.4.2.1, "Adding a Datafile"
- Section A.8.4.2.2, "Modifying a Datafile"
- Section A.8.4.2.3, "Deleting a Datafile"

A.8.4.2.1 Adding a Datafile To add a datafile, click the icon with the plus sign (+):

The Add Datafile screen appears:



Provide the following information:

- File Name

  Specify the name of the datafile.

- File Directory

  Specify the location where this datafile will reside.

- Size

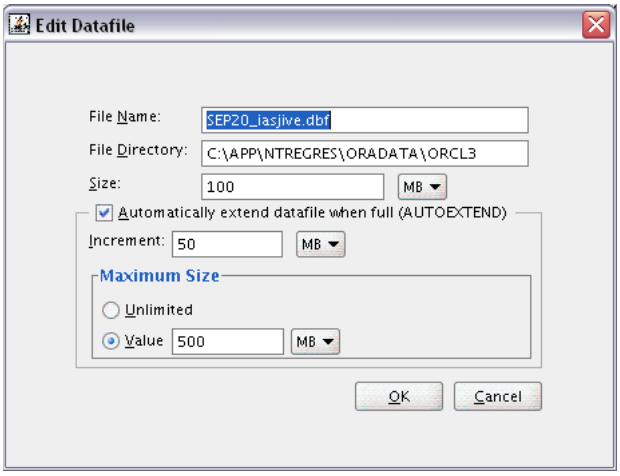
  Specify the initial size of the datafile. Use the drop-down list to specify the size in kilobytes (KB), megabytes (MB), or gigabytes (GB).

- 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.

A.8.4.2.2 Modifying a Datafile 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:

The Edit Datafile screen appears:



Provide the following information:

- File Name

  Specify the name of the datafile.

- 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).

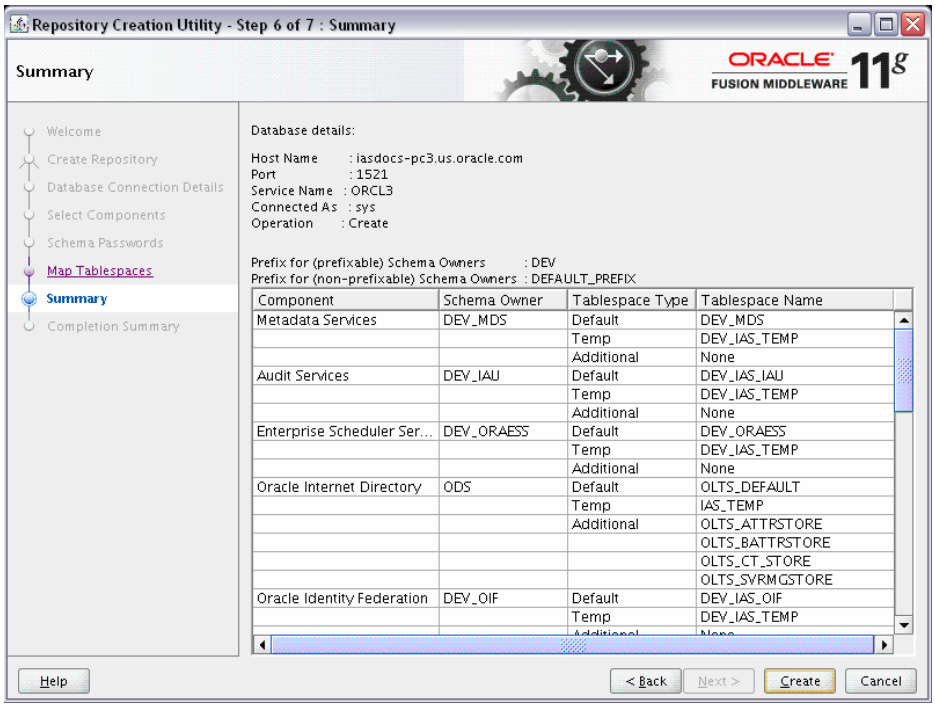
- Select **Automatically extend datafile when full (AUTOEXTEND)** if you want to automatically extend the size of your datafile when it becomes full. In the "Increment" field, specify the size by which your datafile should be increased each time it becomes full. Use the drop-down list to specify the 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.

A.8.4.2.3 Deleting a Datafile To delete a datafile, select the icon next to the datafile name you want to delete, then click the icon with the "X":

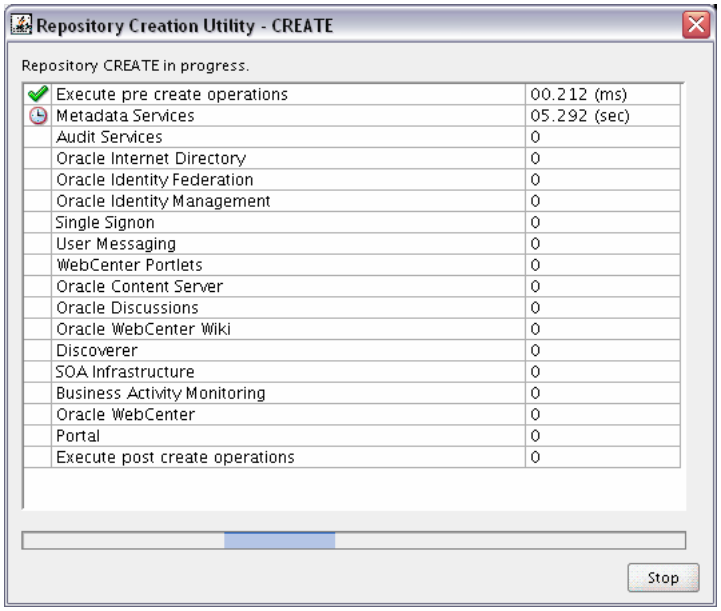
A.9 Summary Screen (for Create Operation)

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**.

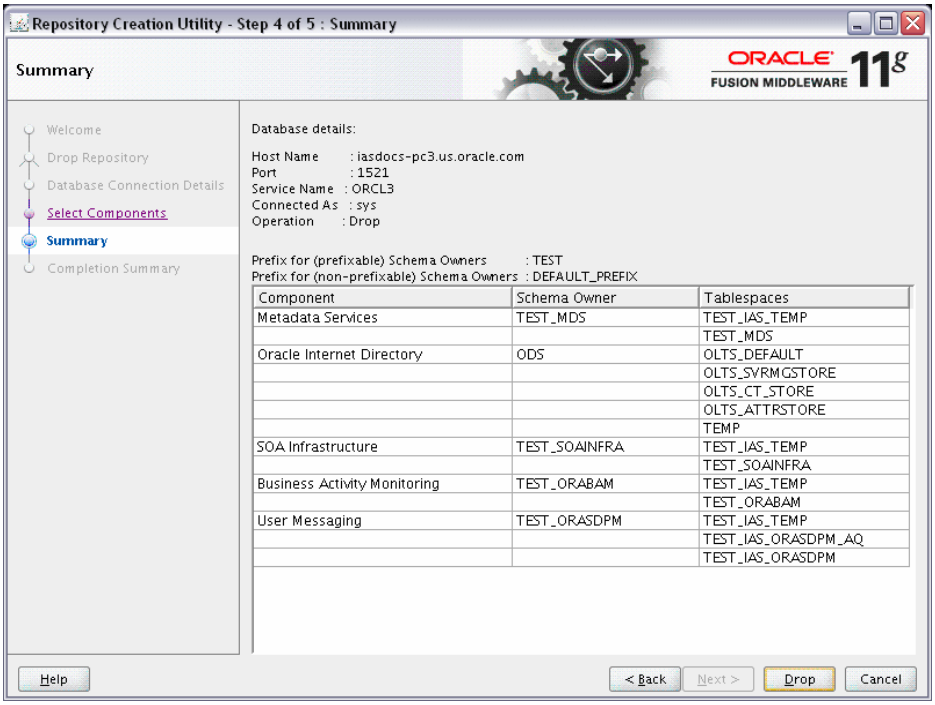
While the schemas are being created, you will see the following progress screen:



Click **Stop** if you want to stop creating the schemas.

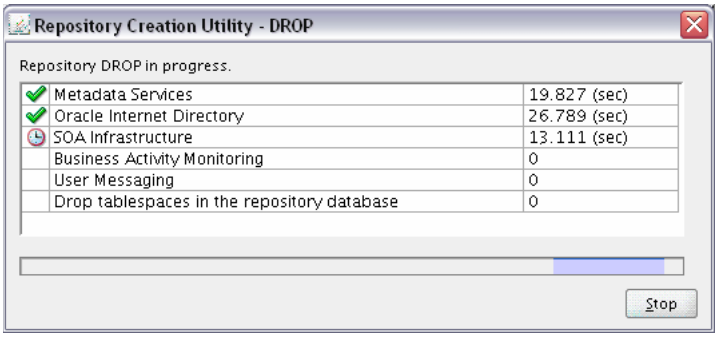
A.10 Summary Screen (for Drop Operation)

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.

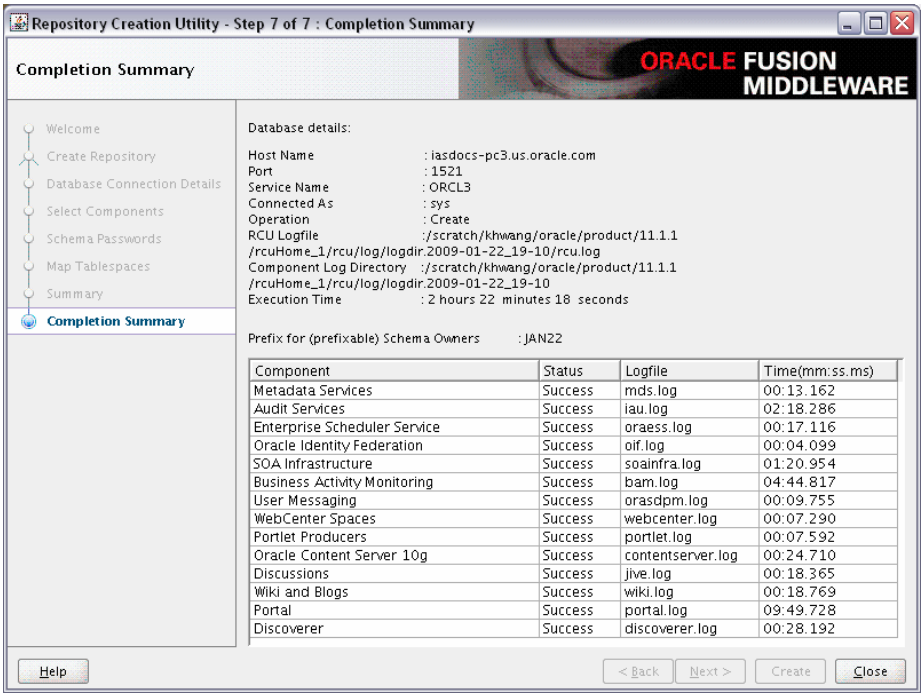
While the schema(s) are being dropped, you will see the following progress screen:



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

A.11 Completion Summary Screen (for Create Operation)

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



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

The main RCU log and component log files are written to the following directory on UNIX systems:

RCU_HOME/rcu/log/logdir.*date_timestamp*

On Windows:

RCU_HOME\rcu\log\logdir.*date_timestamp*

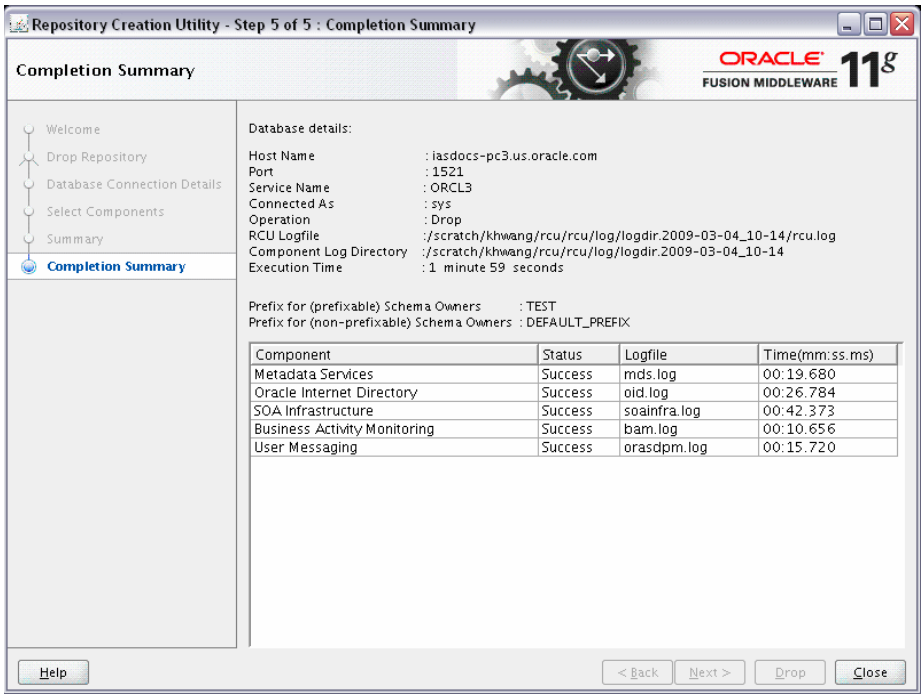
If there were any problems encoutered during schema creation, you can troubleshoot the issue using the log files. For more information, see Appendix B, "Troubleshooting Repository Creation Utility".

If errors are encountered during a Create operation, or if a Create operation fails forany 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 Screen (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 and component log files are written to the following directory on UNIX systems:

RCU_HOME`/rcu/log/logdir.`*date_timestamp*

On Windows:

RCU_HOME`\rcu\log\logdir.`*date_timestamp*

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

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

B

Troubleshooting Repository Creation Utility

This appendix describes solutions to common problems that you might encounter when running Repository Creation Utility (RCU). It contains the following sections:

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

B.1 General Troubleshooting Tips

If you encounter an error during installation:

- Read the *Oracle Fusion Middleware Release Notes* for the latest updates. The release notes are available with the platform-specific documentation. The most current version of the release notes is available on Oracle Technology Network (OTN):

  `http://www.oracle.com/technology/documentation`

- Verify that your computer meets the requirements as specified in the system requirements and prerequisites document, which can be found on OTN:

  `http://www.oracle.com/technology/`

- Verify that your environment meets the certification requirements as specified in the Oracle Fusion Middleware Certification document, which can be found at the following location:

  `http://www.oracle.com/technology/software/products/ias/files/fusion_certification.html`

- 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 B.2, "RCU Log Files").
  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.

B.2 RCU Log Files

The main RCU log file is written to the *RCU_HOME*/rcu/log/logdir.*date_timestamp*/rcu.log (on UNIX) or *RCU_HOME*\rcu\log\logdir.*date_timestamp*\rcu.log (on Windows) file. For example, in a UNIX system:

```
ORACLE_HOME/rcu/log/logdir.2008-04-11_11-00/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 *RCU_HOME*/rcu/log/logdir.*date_timestamp* (on UNIX) or *RCU_HOME*\rcu\log\logdir.*date_timestamp* (on Windows) directory.

Table B–1 lists the component log file names.

Table B–1 RCU Component Log File Names

Component	Log File Name
Metadata Services	mds.log
Audit Services	iau.log
Oracle Internet Directory	oid.log
Single Signon	sso.log
SOA Infrastructure	soainfra.log
Business Activity Monitoring	bam.log
User Messaging	orasdpm.log
Oracle WebCenter	webcenter.log
WebCenter Portlets	portlet.log
Oracle Content Server	contentserver.log
Portal	portal.log
Discoverer	discoverer.log

B.3 Need More Help?

If this appendix does not solve the problem you encountered, try these other sources:

- *Oracle Fusion Middleware Release Notes*, available on the Oracle Technology Network (OTN):

  http://www.oracle.com/technology/documentation

- My Oracle Support (formerly Oracle*MetaLink*):

  http://metalink.oracle.com/

If you do not find a solution for your problem, open a service request.

Index

S

T