Oracle Utilities Work and Asset Management

Database Administrator's Guide

Release 2.2.0.6.0

F21792-01

October 2019

ORACLE®

Oracle Utilities Work and Asset Management Database Administrator's Guide for Release 2.2.0.6.0

Contents

Appendix A

Appendix B

Appendix C

Preface

Welcome to the Oracle Utilities Work and Asset Management Database Administrator's Guide.

This guide provides instructions for installing and maintaining the Oracle Utilities Work and Asset Management database and is intended for database administrators who will be installing and maintaining the database.

Related Documents

For more information on this release, refer to the following documents:

Installation Guides

- Oracle Utilities Work and Asset Management Release Notes
- Oracle Utilities Work and Asset Management Quick Install Guide
- Oracle Utilities Work and Asset Management Installation Guide
- Oracle Utilities Work and Asset Management Database Administrator's Guide
- Oracle Utilities Work and Asset Management Licensing Information User Manual

User Guides

- Oracle Utilities Work and Asset Management and Oracle Utilities Operational Device Management Business User's Guide
- Oracle Utilities Work and Asset Management and Oracle Utilities Operational Device Management Administrative User's Guide
- Oracle Utilities Work and Asset Management Mobile Application User's Guide

Supplemental Documents

- Oracle Utilities Work and Asset Management Server Administration Guide
- Oracle Utilities Work and Asset Management Security Guide

Updates to this Documentation

Additional and updated information about the product is available from the **Knowledge Base** section of **My Oracle Support** (http://support.oracle.com). Please refer to **My Oracle Support** for more information. Documentation updates are also posted on the Oracle Technology Network documentation page as they become available (http://docs.oracle.com/cd/E72219_01/documentation.html).

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 on the screen, or text that you enter.

Chapter 1

Oracle Utilities Work and Asset Management Database Overview

This chapter provides an overview of the Oracle Utilities Work and Asset Management database, including:

- Supported Database Platforms
- Database Maintenance Rules

Supported Database Platforms

Oracle Utilities Work and Asset Management is certified on the following platforms:

Platform	Database Version
AIX 7.1 TL01 AIX 7.2 TL0 (POWER 64-bit)	Oracle Database Server 12.1.0.1+ Oracle Database Server 12.2.0.1.0
Oracle Enterprise Linux 6.x or 7.x (64-bit) x86_64 (64-bit)/ Red Hat Enterprise Linux 6.x or 7.x (64-bit) x86_64 (64-bit)	Oracle Database Server 12.1.0.1+ Oracle Database Server 12.2.0.1.0
Oracle Solaris 11 (SPARC 64-bit)	Oracle Database Server 12.1.0.1+ Oracle Database Server 12.2.0.1.0

> **Note**: Windows Server is **not** supported for Production environments. Wherever Windows Server is referenced within this guide, it is supported for Test or Development environments **only**.

The following Oracle Database Server Editions are supported:

- Oracle Database Enterprise Edition
- Oracle Database Standard Edition

> **Note:** Oracle Database Enterprise Edition and the Partitioning and Advanced Compression options are not mandatory but recommended. Standard Edition should only be considered suitable for environments where scalability, performance, and database size-on-disk are not important considerations. Oracle Database Enterprise Edition, including the Advanced Compression and Partitioning options, is strongly recommended in all other situations.

Refer to My Oracle Support for additional details.

Support for Software Patches and Upgrades

Due to the ongoing nature of software improvement, vendors will periodically issue patches and service packs for the operating systems, application servers and database servers on top of specific versions that Oracle products have already been tested against.

If it is necessary to apply an upgrade, please do so in a test environment that is running on the same platform as your production environment prior to updating the production environment itself. The exception to this is Hibernate software 4.1.0 which should not be upgraded.

Always contact Oracle Support prior to applying vendor updates that do not guarantee backward compatibility.

Database Maintenance Rules

The database supplied with the product consists of the following elements:

- A set of users to administrate, execute and read the database schema provided.

- A set of database roles to implement security for each of the users provided.
- A tablespace and a schema containing the base database objects used by the product.

The installation instructions are outlined in the installation section of this document.

Permitted Database Changes

During and after installation of the product the following changes may be performed by the database administrator personnel on site:

- Users supplied by product may be changed according to the site standards.
- Database objects may be added to the schema according to database naming standards outlined later in this document.
- Database views and indexes may be created against base database objects. Please make sure to prefix new items with "CM" (for customer modification).
- Database storage attributes for base indexes and base tables may be changed according to site standards and hardware used.
- Tablespace names, attributes and locations may be changed according to site standards.
- Database topology (base table/index to tablespace, tablespace to data file, data file to location) may be altered according to tuning and/or site standards.
- Database triggers may be created against base database objects unless they attempt to contravene base data integrity rules.
- Database initialization and parameter settings may be altered according to site standards unless otherwise advised by Oracle Support or outlined.

Non-Permitted Database Changes

In order to maintain operability and upgradeability of the product, during and after the installation of the product, the following changes may *not* be performed by the database administration personnel on site.

Base objects must not be removed or altered in the following ways:

- Columns in base tables must not be altered, removed or added in anyway.
- Columns in Indexes must not be altered or removed.
- Tables must not be renamed or removed.
- Base views must not be renamed or removed.
- Base Triggers and Sequences must not be renamed or removed.
- Base indexes must not be altered or removed.

Chapter 2

Oracle Utilities Work and Asset Management Database Installation Overview

This chapter provides an overview to the installation of the database for the Oracle Utilities Work and Asset Management, including:

- Installation Overview
- Installation Types

Installation Overview

Refer to Supported Database Platforms for information about the supported platforms on which Oracle Utilities Work and Asset Management is verified to operate.

The following types of installation are available for Oracle Utilities Work and Asset Management:

- Initial Install - a database with no demo data.

- Upgrade Install - a database upgrade to v2.2.0.6.0

- Demo Install - a database populated with demo data

Prerequisites

The database installation requires:

- Java Development Kit v8.0

    This must be installed on the Windows or Linux machine where the install package will be staged and run.

Creating the Database

For an initial install or demo install you will create an empty database on the Unix or Windows database server on which you operate the production instance of Oracle Utilities Work and Asset Management.

1. Create the database using the Database Configuration Assistant (DBCA). Make sure to set character set for database as AL32UTF8.

2. Enable Mandatory Software Options.

    - Oracle Spatial OR Oracle Locator

    - Oracle Text

3. Run the SQL below to make sure it is successful.

   ```
   SELECT COMP_NAME,STATUS FROM DBA_REGISTRY WHERE COMP_NAME IN
   ('Spatial','Oracle Text');
   ```

4. Create default tablespace CISTS_01 and required users and roles.

   ```
   CREATE TABLESPACE CISTS_01 LOGGING DATAFILE '/<db_file_location>/
   cists01.dbf' SIZE 1024M REUSE AUTOEXTEND ON NEXT 8192K MAXSIZE
   UNLIMITED EXTENT MANAGEMENT LOCAL UNIFORM SIZE 1M;
   ```

5. Create required roles.

   ```
   CREATE ROLE CIS_USER;
   CREATE ROLE CIS_READ;
   ```

6. Create users.

   ```
   CREATE USER CISADM IDENTIFIED BY CISADM DEFAULT TABLESPACE CISTS_01
   TEMPORARY TABLESPACE TEMP PROFILE DEFAULT;
   GRANT UNLIMITED TABLESPACE TO CISADM WITH ADMIN OPTION;
   GRANT SELECT ANY TABLE TO CISADM;
   GRANT CREATE DATABASE LINK TO CISADM;
   GRANT CONNECT TO CISADM;
   ```

```
GRANT RESOURCE TO CISADM;
GRANT DBA TO CISADM WITH ADMIN OPTION;
GRANT CREATE ANY SYNONYM TO CISADM;
GRANT SELECT ANY DICTIONARY TO CISADM;

CREATE USER CISUSER PROFILE DEFAULT IDENTIFIED BY CISUSER DEFAULT
TABLESPACE CISTS_01 TEMPORARY TABLESPACE TEMP;
GRANT SELECT ANY TABLE TO CISUSER;
GRANT CIS_USER TO CISUSER;
GRANT CIS_READ TO CISUSER;
GRANT CONNECT TO CISUSER;

CREATE USER CISOPR PROFILE DEFAULT IDENTIFIED BY OPRPLUS DEFAULT
TABLESPACE CISTS_01 TEMPORARY TABLESPACE TEMP;
GRANT CONNECT,RESOURCE,EXP_FULL_DATABASE TO CISOPR;

CREATE USER CISREAD IDENTIFIED BY CISREAD DEFAULT TABLESPACE
CISTS_01 TEMPORARY TABLESPACE TEMP;
GRANT SELECT ANY TABLE TO CISREAD;
GRANT CIS_READ TO CISREAD;
GRANT CONNECT TO CISREAD;
```

7. Review the Storage.xml file under the FW44020\Install-Upgrade folder prior to an initial install or upgrade install. This file allocates all base tables and indexes to the default tablespace CISTS_01 and the required users and roles. Information in this file is used by ORADBI.jar while installing the Oracle Utilities Work and Asset Management database objects. Refer to Updating Storage.xml for more details on updating this file.

> **Note:** You will need to review the Storage.xml file, prior to an initial install, to update the default values to custom values (for e.g. TableSpace Name). OraDBI.jar can be executed by a non-schema owner in order to upgrade the database. The initial install still needs to be done by the schema owner.

If you decide to allocate some tables or indexes outside of the default tablespace, change the tablespace name from the default value to a custom value in the Storage.xml file.

For instance, if you decide to allocate table CI_ACCT in a tablespace MyTablespace, change Storage.xml as shown:

```
<CI_ACCT>
<TABLESPACE>MyTablespace</TABLESPACE>
</CI_ACCT>
```

For optimum storage allocation, database administrators should create multiple tablespaces with extents sized to store different types of tables/indexes. They can then edit the storage.xml file before install process, to spread tables and indexes across these tablespaces. Tables and indexes can be created in parallel by editing degree of parallelism. Tablespace, storage options, secure file options, Advanced Compression, and parallel information are used only for new objects. Therefore, for initial installs, information for each object should be reviewed. Be careful while editing this file. Make sure that tablespace names being used exist in the database. Do not change the basic format of this file.

> **Note:** Prior to the installation of the database schema for the product, please verify that the Database Management System software is installed according to your site standards and the installation guide provided by

the database vendor. Also verify that you have necessary licenses to use some of the advanced database features such as Advanced Compression.

Optimizing Storage Allocation

To optimize storage allocation, database administrators should do the following:

1. Create multiple tablespaces with extents sized to store different types of tables/ indexes.

2. Edit the Storage.xml file before the install process, to spread tables and indexes across these tablespaces.
    Be very careful when editing this file and do not change the basic format.
    Tables and indexes can be created in parallel by editing the degree of parallelism.
    Tablespace, storage options, secure file options, Advanced Compression, and parallel information are used only for new objects. Therefore, for initial installs, information for each object should be reviewed. Make sure that tablespace names being used exist in the database.

    > **Note**: Prior to the installation of the database schema for the product, please verify that the Database Management System software is installed according to your site standards and the installation guide provided by the database vendor. Also verify that you have necessary licenses to use some of the advanced database features such as Advanced Compression.

Installation Types

The first step in the installation procedure is to determine the installation type that meets your business requirements. The following are the possible installation types:

- **Initial Install** - a database with no demo data

- **Upgrade Install** - a database upgrade from a previous release

- **Demo Install** - a database populated with demo data

Refer to Supported Database Platforms for information about the supported platforms on which Oracle Utilities Work and Asset Management is verified to operate.

Initial Install

To install Oracle Utilities Work and Asset Management v2.2.0.6 for the first time refer to the Installing Oracle Utilities Work and Asset Management - Initial Installation section for complete instructions.

Upgrade Install

This is applicable for customers who already have Oracle Utilities Work and Asset Management v2.2.0.1.0, v2.2.0.2.0, v2.2.0.3.0, or v2.2.0.4.0 installed.

To upgrade to Oracle Utilities Work and Asset Management v2.2.0.6.0, follow instructions in the Upgrading the Oracle Utilities Work and Asset Management Database section.

Demo Install

For a demo installation, follow the instructions in Installing the Database - Demo Environment.

Installation Notes

When installing components using OraDBI.jar, you may receive the following message in the display output or logs:

```
-  2016-05-23 16:31:38,315 [main] ERROR
(common.cryptography.KeyStoreWrapperFactory) The keystore file
'<filename>' does not exist.…
This file is either provided by the property
com.oracle.ouaf.system.keystore.file or expected to exist at the
default file location null Attempting to use the legacy
cryptography.
 -  2016-05-23 16:31:38,566 [main] INFO  (oem.install.OraDBI)
```

These errors can be safely ignored and the process should proceed to completion.

Chapter 3

Installing the Oracle Utilities Work and Asset Management Database - Initial Installation

This chapter provides the instructions for installing the Oracle Utilities Work and Asset Management v2.2.0.6.0 database as part of initial installation. It describes how to install the database components of Oracle Utilities Work and Asset Management, including:

- Copying and Decompressing Install Media
- Database Creation
- Installing the WAM Schema
- Generating Database Statistics
- Enable USER_LOCK Package

Copying and Decompressing Install Media

To copy and decompress the database:

1. Download the Oracle Utilities Application Framework 4.4.0.2.0 - Database Installation Media, Oracle Utilities Application Framework v4.4.0.2.0 Single Fix Prerequisite Database Rollup for Oracle Utilities Work and Asset Management v2.2.0.6.0 (if there is any) and Oracle Utilities Work and Asset Management v2.2.0.6.0 Oracle Database Multiplatform from Oracle Software Delivery Cloud.

2. Unzip the Oracle Utilities Application Framework 4.4.0.2.0 - Database Installation Media file to a temporary folder. This file contains Oracle Utilities Application Framework folders with all the database components required to install the Oracle Utilities Application Framework 4.4.0.2.0 database.

3. Unzip the Oracle Utilities Application Framework v4.4.0.2.0 Single Fix Prerequisite Database Rollup for Oracle Utilities Work and Asset Management v2.2.0.6.0 to a temporary folder. This file contains Oracle Utilities Application Framework prerequisite single fix database patches for Oracle Utilities Work and Asset Management v2.2.0.6.0.

4. Unzip the Oracle Utilities Work and Asset Management v2.2.0.6.0 Oracle Database Multiplatform file to a temporary folder. This file contains Oracle Utilities Work and Asset Management folders with all the database components required to install the database.

Database Creation

Make sure you have Oracle Database Server installed on your machine in order to create the database.

Creating the Database on UNIX

Create the database using the Database Configuration Assistant (DBCA).

Make sure to set for database as follows:

- Character set for database as AL32UTF8
- National Character Set (NLS_NCHAR_CHARACTERSET - AL16UTF16)

Refer to Creating the Database for steps to create the database.

Creating the Database on Windows

You should be logged in as a user who is a member of the local ORA_DBA group on that server. The ORA_DBA group should have "administrator" privileges assigned to it.

Make sure to set for database as follows:

- Character set for database as AL32UTF8
- National Character Set (NLS_NCHAR_CHARACTERSET - AL16UTF16)

Refer to Creating the Database for steps to create the database.

Database Globalization Support Consideration Storage BYTES/ CHARACTER

The Oracle Utilities Application Framework is a multilingual capable application that supports the storage, processing and retrieval of data in multiple languages by leveraging the Oracle Database globalization support architecture. Use of the AL32UTF8 Unicode character encoding system allows the database to support multiple languages. If your application will support multiple languages with any one of which being multibyte you should consider the use of Character Length Semantics to store data in database columns in terms of CHARACTERS rather than in terms of BYTEs.

 At this time Oracle Utilities Application Framework only supports CHAR NLS_LENGTH_SEMANTICS setting at the instance level. Since this is an instance wide setting great care should be taken and a thorough evaluation should be performed if custom or third party components utilize the same database instance as the Framework application.

Note that the use of MAX_STRING_SIZE of EXTENDED is not supported at this time.

By default the database is created with BYTE length semantics. If you are going to store data using CHARACTER length semantics then use following procedure.

Initial Install
1. Execute the following statement to set nls_length_semantics=CHAR

   ```
   SQL> ALTER SYSTEM SET nls_length_semantics=CHAR SCOPE=BOTH;
   ```

2. Restart the database.

3. Verify that the nls_length_semantics is CHAR using following command:

   ```
   SQL> SHOW PARAMETER nls_length_semantics
   ```

   Note: For pluggable databases make sure to set nls_length_semantics=CHAR for both the container and pluggable database.

There are multiple ways to migrate a database from BYTE to CHAR length semantics:

1. By Script: Refer to Doc ID 313175.1 from Oracle Support.

2. Alternative procedure: The following is an alternate way to create a schema with character-length semantics, and then importing the date from a byte-based export.

Migrating from BYTE based storage to CHARACTER based storage:

1. Create a database using DBCA.

2. Execute the following statement to set nls_length_semantics=CHAR

   ```
   SQL> ALTER SYSTEM SET nls_length_semantics=CHAR SCOPE=BOTH;
   ```

3. Restart the database.

4. Make sure nls_length_semantics is CHAR using the following command:

   ```
   SQL> SHOW PARAMETER nls_length_semantics
   ```

   Note: For pluggable databases make sure to set nls_length_semantics=CHAR for both the container and pluggable database.

5. Export the schema from database which has nls_semantics_legth=BYTE using the following command:

```
expdp userid=system/<code>@<SID> directory=<DIR_NAME>
schemas=<schema_name> dumpfile=<schema_name>.dmp
logfile=<schema_name>.log
```

6. Generate DDL from the dump file using the Oracle impdp utility:

```
impdp userid=system/<code>@<SID> directory=<DIR_NAME>
DUMPFILE=<schema_name>.dmp SCHEMAS=<schema_name>
SQLFILE=<schema_name>_DDL.sql
```

7. Replace the word "Byte" with "Char" in <schema_name>DDL.sql.

   For vi editor in Linux environment use following command to replace "Byte" with "Char":

```
:%s/BYTE/CHAR/g
```

8. Replace the schema name also if it is required for the environment.

9. Execute <schema_name>DDL.sql, which was generated in step No.6, which will create objects in the schema.

   Execute following command to make sure the number of objects at source and target are equal

```
SQL>select OWNER || ' ' || OBJECT_TYPE || ' ' || COUNT(*) || ' ' ||
STATUS FROM DBA_OBJECTS WHERE OWNER in ('<SCHEMA_NAMe>') GROUP BY
OWNER, OBJECT_TYPE , STATUS ORDER BY OBJECT_TYPE;
```

10. If any object is missing for any reason create it, by fixing theDDL manually (DDL for each object is available in the file which was created in step No.6), and execute the DDL for the objects which were not created.

11. Generate s DDL to disable triggers using following command.

```
SQL> SELECT 'ALTER TABLE' || ' ' ||TABLE_NAME || ' ' || 'DISABLE ALL
TRIGGERS;' FROM USER_TABLES;
```

12. Execute the script generated from step No.11 to disable all triggers.

13. Import data only. Use following command to import data only into the schema created to support CHAR-based database storage. Use following command to import data only.

```
impdp userid=system/<code>@<SID> dumpfile=<schema_name>.dmp
CONTENT=DATA_ONLY SCHEMAS=<schema_name>
LOGFILE=<schema_name>_import.log
```

14. Enable triggers. Use following command to generate a DDL for triggers.

```
SQL>SELECT 'ALTER TABLE' || ' ' ||TABLE_NAME || ' ' || 'ENABLE ALL
TRIGGERS;' FROM USER_TABLES;
```

15. Execute the script generated from step No.14 to enable all triggers.

Exclude Tables and Indexes

If you want to exclude an index or table during the upgrade process you can do it using the following procedure.

Edit the OraSchUpg.inp file in the Install-Upgrade directory to list the tables and indexes to be excluded. Specify the tables and indexes in the following format:

```
-INDEX: 'INDEX_NAME','INDEX_NAME'
-TABLE: 'TBALE1_NAME','TABLE2_NAME'
```

For example: To exclude table F1_WEB_SVC:

```
-TABLE:'F1_WEB_SVC'
```

If there are multiple tables include them with separated coma, such as:

```
-TABLES: 'TABLE-1','TABLE_2','TABLE_3'
```

Similarly for indexes:

```
-INDEX:' F1C064S1'
```

If there are multiple indexes you can include them with separated coma, such as:

```
-INDEX:'INDEX_1',INDEX_2','INDEX_3
```

Installing the WAM Schema

You must install the Oracle Utilities Application Framework v4.4.0.2.0 and Oracle Utilities Application Framework v4.4.0.2.0 Single Fix Prerequisite Database Rollup for Oracle Utilities Work and Asset Management v2.2.0.6.0 prior to Oracle Utilities Work and Asset Management v2.2.0.6.0. The files for Oracle Utilities Application Framework installation are located in the FW44020 folder specified in the Copying and Decompressing Install Media section.

This section provides guidelines for:

- Installing the Oracle Utilities Application Framework Database Component using OraDBI.jar

- Installing Oracle Utilities Application Framework Prerequisite Database Single Fixes

- Installing Oracle Utilities Work and Asset Management Database Component using OraDBI.jar

- Tasks Performed by ORADBI

Installing the Oracle Utilities Application Framework Database Component using OraDBI.jar

Prepare the following parameters before installation:

- The name of the database server in which the database is configured - DB_SERVER

- The listener port number of the database - PORT

- The target database name in which the product is to be installed - SID

- A database user that will own the application schema (for example: CISADM) - DBUSER

- Password of the database user that will own the application schema - DBPASS

- A database user that has read-write (select/update/insert/delete) privileges to the objects in the application schema (for example: CISUSER). The application will access the database as this user - RWUSER

- A database user with read-only privileges to the objects in the application schema. (for example: CISREAD) - RUSER

- A database role that has read-write (select/update/insert/delete) privileges to the objects in the application schema. (for example: CIS_USER) - RW_USER_ROLE

- A database role with read-only privileges to the objects in the application schema (for example: CIS_READ) - R_USER_ROLE

- Location for jar files (the Jar files are bundled with the database package) - CLASSPATH

- The database user password with read-write privileges - RWUSER_PASS

- The database user password with read-only privileges - RUSER_PASS

Granting Privileges to the Database Roles

Make sure to provide the CREATE SYNONYM grants to the database role with read-write and read-only privileges.

```
GRANT CREATE SYNONYM TO CIS_USER;
GRANT CREATE SYNONYM TO CIS_READ;
```

You can execute OraDBI.jar using either of the following methods:

- Using Interactive Mode to Install the Oracle Utilities Application Framework Database Component using OraDBI.jar

- Using the Command Line to Install the Oracle Utilities Application Framework Database Component using OraDBI.jar

Using Interactive Mode to Install the Oracle Utilities Application Framework Database Component using OraDBI.jar

The following procedure lists the steps to install the schema for Oracle Utilities Application Framework v4.4.0.2.0 using OraDBI.

Run the following commands with the defined parameters on the command prompt from the Install-Upgrade folder where the Oracle Utilities Application Framework 4.4.0.2.0 - Database Installation Media was unzipped.

1. Open a command line prompt in the Windows environment.

2. Set Java Home.

   In the following example, JDK 1.8 is installed in C:\Program Files\Java\jdk1.8.0_131 directory.

   ```
   SET JAVA_HOME=C:\Program Files\Java\jdk1.8.0_131
   ```

3. Set the class path.

   In the following example, the required jarfiles, including OraDBI.jar, are available in the C:\InstallUpgrade\Jarfiles* directory.

   ```
   SET CLASSPATH=C:\InstallUpgrade\Jarfiles\*
   ```

4. Execute the following command:

   ```
   "%JAVA_HOME%"\bin\java -Xmx1500M -cp %CLASSPATH%
   com.oracle.ouaf.oem.install.OraDBI
   -p <RWUSER_PASS>,<RUSER_PASS>

      (or)

   "C:\Program Files\Java\jdk1.8.0_101"\bin\java -Xmx1500M -cp
   C:\InstallUpgrade\Jarfiles\* com.oracle.ouaf.oem.install.OraDBI -p
   <RWUSER_PASS>,<RUSER_PASS>
   ```

   The utility prompts you to enter values for the following parameters as per your environment:

   - Name of the database server: <DB SERVER>

   - Port no: <PORT>

   - Name of the target database: <SID>

   - Name of the owner of the database schema: <DBUSER>

   - Password of the user name: <DBPASS>

   - Oracle user with read-write privileges to the Database Schema: <CISUSER>

   - Password of the user with read-write privileges: <DBPASS>

   - Oracle user with read-only privileges to the Database Schema: <CISREAD>

   - Password of the user with read-only privileges: <DBPASS>

   - Oracle database role with read-write privileges to the Database Schema: <CIS_USER>

   - Oracle database role with read-only privileges to the Database Schema: <CIS_READ>

   - Enter the name of the target schema where you want to install or upgrade: <CISADM>

   This process generates log files in the directory Install-Upgrade\logs. Check the log files for any errors. Refer to Installation Notes for more information.

   Note: You can also run the OUAF install process on a Linux machine which has Java installed by exporting the JAVA HOME and CLASSPATH.

   Example:

   ```
   export JAVA_HOME=/scratch/spl/jdk1.8.0_92
   export PATH=$JAVA_HOME/bin:$PATH

   export CLASSPATH=/scratch/Candidate/FW-V4.4.0.2.0-Database/
   jarfiles/*
   ```

In interactive mode, you can run the installer using the command:

```
java -Xmx1500M com.oracle.ouaf.oem.install.OraDBI -p
<RWUSER_PASS>,<RUSER_PASS>
```

Using the Command Line to Install the Oracle Utilities Application Framework Database Component using OraDBI.jar

Run the following commands with the defined parameters on the command prompt from the Install-Upgrade folder where the Oracle Utilities Application Framework 4.4.0.2.0 - Database Installation Media was unzipped.

```
"C:\Program Files\Java\jdk1.8.0_65\bin\java" -Xmx1500M -cp /
scratch/Candidate/FW-V4.4.0.2.0-Database/jarfiles/spl/*
com.oracle.ouaf.oem.install.OraDBI -d
jdbc:oracle:thin:@<DatabaseServer_Name>:1521/<DatabaseName>,
%DBUSER%,%DBPASS%, CISUSER, CISREAD, CIS_USER, CIS_READ, %DBUSER% -
l 1,2 -j "C:\Program Files\Java\jdk1.8.0_65" -p
<RWUSER_PASS>,<RUSER_PASS>
```

OR

```
"%JAVA_HOME%"\bin\java -Xmx1500M -cp %CLASSPATH%
com.oracle.ouaf.oem.install.OraDBI -d jdbc:oracle:thin:@<Database
ServerName>:1521/
<Database_Name>,%DBUSER%,%DBPASS%,CISUSER,CISREAD,CIS_USER,CIS_REA
D,%DBUSER% -l 1,2 -j "%JAVA_HOME%" -p <RWUSER_PASS>,<RUSER_PASS>
```

This process generates log files in the directory Install-Upgrade\logs. Check the log files for any errors. Refer to Installation Notes for more information.

This process can also be run on a Linux machine which has Java installed by exporting the JAVA HOME and CLASSPATH.

> **Note**: The following error messages can be ignored during process:
>
> ```
> - 2016-05-23 16:31:38,315 [main] ERROR
> (common.cryptography.KeyStoreWrapperFactory) The keystore file
> '<filename>' does not exist.…
> …
> ```

This file is either provided by the property com.oracle.ouaf.system.keystore.file or expected to exist at the default file location null Attempting to use the legacy cryptography.

```
- 2016-05-23 16:31:38,566 [main] INFO  (oem.install.OraDBI)
```

Installing Oracle Utilities Application Framework Prerequisite Database Single Fixes

Before installing Oracle Utilities Work and Asset Management, you must install Oracle Utilities Application Framework v4.4.0.2.0 Single Fix Prerequisite Database Rollup for Oracle Utilities Work and Asset Management v2.2.0.6.0.

Applying Hot Fixes

> **Note**: Java 8 JDK should be installed on the machine to use the command. make sure to install the JDK that is supported for your platform.

1. Copy db_patch_standalone.jar from ..\WAM-V22060-FW-PREREQ-DB\ FW44020-HFix folder to any directory on your local machine under dbpatch_tools folder.

   UNIX

   ```
   cd /../dbpatch_tools
   jar xvf db_patch_standalone.jar
   ```

   Windows

   ```
   cd c:\..\dbpatch_tools
   jar xvf db_patch_standalone.jar
   ```

2. SET TOOLSBIN.

   UNIX

   ```
   export TOOLSBIN=/../dbpatch_tools/bin
   ```

   Windows

   ```
   SET TOOLSBIN=c:\..\dbpatch_tools\bin
   ```

3. Apply prerequisite Oracle Utilities Application Framework database single fixes from ..\ WAM-V22060-FW-PREREQ-DB \FW44020-HFix folder.

   UNIX

   a. Change the permission of ouafDatabasePatch.sh tool.

      ```
      chmod 755 ouafDatabasePatch.sh
      ```

   b. Run the ouafDatabasePatch.sh tool.

      ```
      sh ouafDatabasePatch.sh (or) ./ ouafDatabasePatch.sh
      ```

   Windows

   a. Run the ouafDatabasePatch.cmd tool.

      ```
      ouafDatabasePatch.cmd
      ```

The utility prompts you to enter values for the parameters listed below:

- Enter the target database type (O/M/D) [O]: <O>

- Enter the username that owns the schema: <CISADM>

- Enter the password for the cisadm user: <CISADM Password>

- Enter the name of the Oracle Database Connection String: <DB_Server:DBPORT:ORACLE_SID>

Installing Oracle Utilities Work and Asset Management Database Component using OraDBI.jar

Prepare the following parameters before installation:

- The name of the database server in which the database is configured - DB_SERVER

- The listener port number of the database - PORT

- The target database name in which the product is to be installed - SID

- A database user that will own the application schema (for example: CISADM) - DBUSER

- Password of the database user that will own the application schema - DBPASS

- A database user that has read-write (select/update/insert/delete) privileges to the objects in the application schema (for example: CISUSER). The application will access the database as this user - RWUSER

- A database user with read-only privileges to the objects in the application schema. (for example: CISREAD) - RUSER

- A database role that has read-write (select/update/insert/delete) privileges to the objects in the application schema. (for example: CIS_USER) - RW_USER_ROLE

- A database role with read-only privileges to the objects in the application schema. (for example: CIS_READ) - R_USER_ROLE

- Location for jar files. (The Jar files are bundled with the database package) - CLASSPATH

- Java Home (for example: C:\Java\jdk1.8.0) - JAVA_HOME

- The database user password with read-write privileges - RWUSER_PASS

- The database user password with read-only privileges - RUSER_PASS

You can execute OraDBI.jar using either of the following methods:

- Using Interactive Mode to Install Using ORADBI.jar

- Using the Command Line to Install Using ORADBI.jar

Using Interactive Mode to Install Using ORADBI.jar

The following procedure lists the steps to install the schema for Oracle Utilities Work and Asset Management v2.2.0.6.0 using OraDBI.

Run the following commands with the defined parameters on the command prompt from ..\WAM\Install-Upgrade directory:

1. Open a command line prompt in the Windows environment.

2. Set Java Home.

   In the following example, JDK 1.8 is installed in C:\Program Files\Java\jdk1.8.0_101 directory.

   ```
   SET JAVA_HOME=C:\Program Files\Java\jdk1.8.0_101
   ```

3. Set the class path.

   In the following example, the required jarfiles, including OraDBI.jar, are available in the C:\InstallUpgrade\Jarfiles* directory.

   ```
   SET CLASSPATH=C:\InstallUpgrade\Jarfiles\*
   ```

4. Execute the following command:

```
"%JAVA_HOME%"\bin\java -Xmx1500M -cp %CLASSPATH%
com.oracle.ouaf.oem.install.OraDBI -p <RWUSER_PASS>,<RUSER_PASS>
```

```
    (or)
```

```
"C:\Program Files\Java\jdk1.8.0_101"\bin\java -Xmx1500M -cp
C:\InstallUpgrade\Jarfiles\* com.oracle.ouaf.oem.install.OraDBI -p
<RWUSER_PASS>,<RUSER_PASS>
```

The utility prompts you to enter values for the following parameters as per your environment:

- Name of the database server: <DB SERVER>

- Port no: <PORT>

- Name of the target database: <SID>

- Name of the owner of the database schema: <DBUSER>

- Password of the user name: <DBPASS>

- Oracle user with read-write privileges to the Database Schema: <CISUSER>

- Password of the user with read-write privileges: <DBPASS>

- Oracle user with read-only privileges to the Database Schema: <CISREAD>

- Password of the user with read-only privileges: <DBPASS>

- Oracle database role with read-write privileges to the Database Schema: <CIS_USER>

- Oracle database role with read-only privileges to the Database Schema: <CIS_READ>

- Enter the name of the target schema where you want to install or upgrade: <CISADM>

  This process generates log files in the directory Install-Upgrade\logs. Check the log files for any errors. Refer to Installation Notes for more information.

You can also run the process on a Linux machine which has Java installed by exporting the JAVA HOME and CLASSPATH.

In interactive mode, you can run the installer using the command:

```
java -Xmx1500M com.oracle.ouaf.oem.install.OraDBI -p
<RWUSER_PASS>,<RUSER_PASS>
```

Using the Command Line to Install Using ORADBI.jar
Run the following commands with the defined parameters at the command prompt from the ..\WAM\Install-Upgrade directory

```
"C:\Program Files\Java\jdk1.8.0_65\bin\java" -Xmx1500M -cp
%CLASSPATH% com.oracle.ouaf.oem.install.OraDBI -d
jdbc:oracle:thin:@<DatabaseServer_Name>:1521/<DatabaseName>,
%DBUSER%,%DBPASS%, CISUSER, CISREAD, CIS_USER, CIS_READ, %DBUSER% -
l 1,2 -j "C:\Program Files\Java\jdk1.8.0_65" -p
<RWUSER_PASS>,<RUSER_PASS>
```

OR

```
"%JAVA_HOME%"\bin\java -Xmx1500M -cp %CLASSPATH%
com.oracle.ouaf.oem.install.OraDBI -d jdbc:oracle:thin:@<Database
ServerName>:1521/
<Database_Name>,%DBUSER%,%DBPASS%,CISUSER,CISREAD,CIS_USER,CIS_REA
D,%DBUSER% -l 1,2 -j "%JAVA_HOME%" -p <RWUSER_PASS>,<RUSER_PASS>
```

This process generates log files in the directory Install-Upgrade\logs. Check the log files for any errors. Refer to Installation Notes for more information.

This process can also be run on a Linux machine which has Java installed by exporting the JAVA HOME and CLASSPATH.

Tasks Performed by ORADBI

The following tasks are performed by ORADBI:

- Interacts with the user to collect information about the name of Oracle account that will own the application schema (for example, CISADM), password of this account, password of the SYSTEM account in the database, and the name of the Oracle account that the application user will use (for example, CISUSER), and the name of the Oracle account that will be assigned read-only privileges to the application schema (for example, CISREAD).

- Verifies whether tablespace names already exist in the Storage.xml file (if not, the process will abort).

- Installs the schema, installs the system data, and configures security.

- Maintains upgrade log tables in the database.

- Updates release ID when the upgrade is completed successfully.

- If an error occurs while executing a SQL script or another utility, it logs and displays the error message and allows you to re-execute the current step. Log files ORADBI###.log are created in the same folder as ORADBI and contains all the SQL commands executed against the database along with the results. The log files are incremental so that the results are never overwritten. If warning messages are generated during the upgrade, ORADBI prompts the user at the end of the process. Users should check the log files to verify the warning messages. Warning messages are only alerts and do not necessary mean a problem exists.

- Stores the Schema owner and password in the feature configuration table. The password is stored in encrypted format.

- OraDBI can be executed by a non-schema owner.

- OraDBI automatically runs the Security Oragensec utility.

Generating Database Statistics

During an install process new database objects may be added to the target database.

Before starting to use the database, generate the complete statistics for these new objects by using the DBMS_STATS package.

Enable USER_LOCK Package

For inbound web services to work, the USER_LOCK must be enabled at the database level. This is a one-time step. If this is not already enabled, enable it using the following steps:

1. Login as SYS user.

2. On SQL prompt run.

   ```
   @<ORACLE_HOME>/rdbms/admin/userlock.sql
   ```

3. Grant permission by running following SQL:

   ```
   grant execute on USER_LOCK to public;
   ```

Note that grant can also be made to the database user that the application connects to (for example, cisuser), instead of 'public'.

Chapter 4

Upgrading the Oracle Utilities Work and Asset Management Database

This chapter provides the instructions for upgrading the database to Oracle Utilities Work and Asset Management v2.2.0.6.0. It describes how to upgrade the database components of Oracle Utilities Work and Asset Management, including:

- Copying and Decompressing Install Media

- Generating Database Statistics

- Enable USER_LOCK package

Copying and Decompressing Install Media

To copy and decompress the database:

1. Download the Oracle Utilities Application Framework 4.4.0.2.0 - Database Installation Media, Oracle Utilities Application Framework v4.4.0.2.0 Single Fix Prerequisite Database Rollup for Oracle Utilities Work and Asset Management v2.2.0.6.0 (if there is any) and Oracle Utilities Work and Asset Management v2.2.0.6.0 Oracle Database Multiplatform from Oracle Software Delivery Cloud.

2. Unzip the Oracle Utilities Application Framework 4.4.0.2.0 - Database Installation Media file to a temporary folder. This file contains Oracle Utilities Application Framework folders with all the database components required to install the Oracle Utilities Application Framework 4.4.0.2.0 database.

3. Unzip the Oracle Utilities Application Framework v4.4.0.2.0 Single Fix Prerequisite Database Rollup for Oracle Utilities Work and Asset Management v2.2.0.6.0 to a temporary folder. This file contains Oracle Utilities Application Framework prerequisite single fix database patches for Oracle Utilities Work and Asset Management v2.2.0.6.0.

4. Unzip the Oracle Utilities Work and Asset Management v2.2.0.6.0 Oracle Database Multiplatform file to a temporary folder. This file contains Oracle Utilities Work and Asset Management folders with all the database components required to install the database.

Upgrading the Schema

The following section assumes that you have Oracle Utilities Work and Asset Management v2.2.0.1.0, v2.2.0.2.0, v2.2.0.3.0, v2.2.0.4.0 installed.

> **Note**: You must have Oracle Database Server 12.2.0.1+ installed on your machine to continue with the procedure. If you do not have Oracle Database Server 12.2.0.1+ installed, please upgrade to Oracle Database Server 12.2.0.1+ first before proceeding.

Upgrading the Oracle Utilities Application Framework Database Component using OraDBI.jar

Prepare the following parameters before installation:

- The name of the database server in which the database is configured - DB_SERVER

- The listener port number of the database - PORT

- The target database name in which the product is to be installed - SID

- A database user that will own the application schema (for example: CISADM) - DBUSER

- Password of the database user that will own the application schema - DBPASS

- A database user that has read-write (select/update/insert/delete) privileges to the objects in the application schema (for example: CISUSER). The application will access the database as this user - RWUSER

- A database user with read-only privileges to the objects in the application schema. (for example: CISREAD) - RUSER

- A database role that has read-write (select/update/insert/delete) privileges to the objects in the application schema. (for example: CIS_USER) - RW_USER_ROLE

- A database role with read-only privileges to the objects in the application schema (for example: CIS_READ) - R_USER_ROLE

- Location for jar files (the Jar files are bundled with the database package) - CLASSPATH

- The database user password with read-write privileges - RWUSER_PASS

- The database user password with read-only privileges - RUSER_PASS

Granting Privileges to the Database Roles

Make sure to provide the CREATE SYNONYM grants to the database role with read-write and read-only privileges.

```
GRANT CREATE SYNONYM TO CIS_USER;
GRANT CREATE SYNONYM TO CIS_READ;
```

You can execute OraDBI.jar using either of the following methods:

- Using Interactive Mode to Install the Oracle Utilities Application Framework Database Component using OraDBI.jar

- Using the Command Line to Install the Oracle Utilities Application Framework Database Component using OraDBI.jar

Using Interactive Mode to Install the Oracle Utilities Application Framework Database Component using OraDBI.jar

The following procedure lists the steps to install the schema for Oracle Utilities Application Framework v4.4.0.2.0 using OraDBI.

Run the following commands with the defined parameters on the command prompt from the Install-Upgrade folder where the Oracle Utilities Application Framework 4.4.0.2.0 - Database Installation Media was unzipped.

1. Open a command line prompt in the Windows environment.

2. Set Java Home.

   In the following example, JDK 1.8 is installed in C:\Program Files\Java\jdk1.8.0_131 directory.

   ```
   SET JAVA_HOME=C:\Program Files\Java\jdk1.8.0_131
   ```

3. Set the class path.

   In the following example, the required jarfiles, including OraDBI.jar, are available in the C:\InstallUpgrade\Jarfiles* directory.

   ```
   SET CLASSPATH=C:\InstallUpgrade\Jarfiles\*
   ```

4. Execute the following command:

   ```
   "%JAVA_HOME%"\bin\java -Xmx1500M -cp %CLASSPATH%
   com.oracle.ouaf.oem.install.OraDBI
   -p <RWUSER_PASS>,<RUSER_PASS>
   ```

```
(or)
```

```
"C:\Program Files\Java\jdk1.8.0_101"\bin\java -Xmx1500M -cp
C:\InstallUpgrade\Jarfiles\* com.oracle.ouaf.oem.install.OraDBI -p
<RWUSER_PASS>,<RUSER_PASS>
```

The utility prompts you to enter values for the following parameters as per your environment:

- Name of the database server: <DB SERVER>

- Port no: <PORT>

- Name of the target database: <SID>

- Name of the owner of the database schema: <DBUSER>

- Password of the user name: <DBPASS>

- Oracle user with read-write privileges to the Database Schema: <CISUSER>

- Password of the user with read-write privileges: <DBPASS>

- Oracle user with read-only privileges to the Database Schema: <CISREAD>

- Password of the user with read-only privileges: <DBPASS>

- Oracle database role with read-write privileges to the Database Schema: <CIS_USER>

- Oracle database role with read-only privileges to the Database Schema: <CIS_READ>

- Enter the name of the target schema where you want to install or upgrade: <CISADM>

This process generates log files in the directory Install-Upgrade\logs. Check the log files for any errors. Refer to Installation Notes for more information.

Note: You can also run the OUAF install process on a Linux machine which has Java installed by exporting the JAVA HOME and CLASSPATH.

Example:
```
export JAVA_HOME=/scratch/spl/jdk1.8.0_92
export PATH=$JAVA_HOME/bin:$PATH

export CLASSPATH=/scratch/Candidate/FW-V4.4.0.2.0-Database/
jarfiles/*
```

In interactive mode, you can run the installer using the command:

```
java -Xmx1500M com.oracle.ouaf.oem.install.OraDBI -p
<RWUSER_PASS>,<RUSER_PASS>
```

Using the Command Line to Install the Oracle Utilities Application Framework Database Component using OraDBI.jar

Run the following commands with the defined parameters on the command prompt from the Install-Upgrade folder where the Oracle Utilities Application Framework 4.4.0.2.0 - Database Installation Media was unzipped.

```
"C:\Program Files\Java\jdk1.8.0_65\bin\java" -Xmx1500M -cp /
scratch/Candidate/FW-V4.4.0.2.0-Database/jarfiles/spl/*
com.oracle.ouaf.oem.install.OraDBI -d
jdbc:oracle:thin:@<DatabaseServer_Name>:1521/<DatabaseName>,
%DBUSER%,%DBPASS%, CISUSER, CISREAD, CIS_USER, CIS_READ, %DBUSER% -
l 1,2 -j "C:\Program Files\Java\jdk1.8.0_65" -p
<RWUSER_PASS>,<RUSER_PASS>
```

OR

```
"%JAVA_HOME%"\bin\java -Xmx1500M -cp %CLASSPATH%
com.oracle.ouaf.oem.install.OraDBI -d jdbc:oracle:thin:@<Database
ServerName>:1521/
<Database_Name>,%DBUSER%,%DBPASS%,CISUSER,CISREAD,CIS_USER,CIS_REA
D,%DBUSER% -l 1,2 -j "%JAVA_HOME%" -p <RWUSER_PASS>,<RUSER_PASS>
```

This process generates log files in the directory Install-Upgrade\logs. Check the log files for any errors. Refer to Installation Notes for more information.

This process can also be run on a Linux machine which has Java installed by exporting the JAVA HOME and CLASSPATH.

> **Note**: The following error messages can be ignored during process:
>
> ```
> - 2016-05-23 16:31:38,315 [main] ERROR
> (common.cryptography.KeyStoreWrapperFactory) The keystore file
> '<filename>' does not exist.…
> …
> ```

This file is either provided by the property com.oracle.ouaf.system.keystore.file or expected to exist at the default file location null Attempting to use the legacy cryptography.

> ```
> - 2016-05-23 16:31:38,566 [main] INFO  (oem.install.OraDBI)
> ```

Installing Oracle Utilities Application Framework Prerequisite Database Single Fixes

Before installing Oracle Utilities Work and Asset Management, you must install Oracle Utilities Application Framework v4.4.0.2.0 Single Fix Prerequisite Database Rollup for Oracle Utilities Work and Asset Management v2.2.0.6.0.

Applying Hot Fixes

> **Note**: Java 8 JDK should be installed on the machine to use the command. make sure to install the JDK that is supported for your platform.

1. Copy db_patch_standalone.jar from ..\WAM-V22060-FW-PREREQ-DB\ FW44020-HFix folder to any directory on your local machine under dbpatch_tools folder.

   UNIX

   ```
   cd /../dbpatch_tools
   jar xvf db_patch_standalone.jar
   ```

   Windows

   ```
   cd c:\..\dbpatch_tools
   jar xvf db_patch_standalone.jar
   ```

2. SET TOOLSBIN.

   UNIX

   ```
   export TOOLSBIN=/../dbpatch_tools/bin
   ```

   Windows

   ```
   SET TOOLSBIN=c:\..\dbpatch_tools\bin
   ```

3. Apply prerequisite Oracle Utilities Application Framework database single fixes from ..\ WAM-V22060-FW-PREREQ-DB \FW44020-HFix folder.

   UNIX

   a. Change the permission of ouafDatabasePatch.sh tool.

      ```
      chmod 755 ouafDatabasePatch.sh
      ```

   b. Run the ouafDatabasePatch.sh tool.

      ```
      sh ouafDatabasePatch.sh (or) ./ ouafDatabasePatch.sh
      ```

   Windows

   a. Run the ouafDatabasePatch.cmd tool.

      ```
      ouafDatabasePatch.cmd
      ```

   The utility prompts you to enter values for the parameters listed below:

   • Enter the target database type (O/M/D) [O]: <O>

   • Enter the username that owns the schema: <CISADM>

   • Enter the password for the cisadm user: <CISADM Password>

   • Enter the name of the Oracle Database Connection String: <DB_Server:DBPORT:ORACLE_SID>

Upgrading Oracle Utilities Work and Asset Management Database Component using OraDBI.jar

Prepare the following parameters before installation:

• The name of the database server in which the database is configured - DB_SERVER

- The listener port number of the database - PORT

- The target database name in which the product is to be installed - SID

- A database user that will own the application schema (for example: CISADM) - DBUSER

- Password of the database user that will own the application schema - DBPASS

- A database user that has read-write (select/update/insert/delete) privileges to the objects in the application schema (for example: CISUSER). The application will access the database as this user - RWUSER

- A database user with read-only privileges to the objects in the application schema. (for example: CISREAD) - RUSER

- A database role that has read-write (select/update/insert/delete) privileges to the objects in the application schema. (for example: CIS_USER) - RW_USER_ROLE

- A database role with read-only privileges to the objects in the application schema. (for example: CIS_READ) - R_USER_ROLE

- Location for jar files. (The Jar files are bundled with the database package) - CLASSPATH

- Java Home (for example: C:\Java\jdk1.8.0) - JAVA_HOME

- The database user password with read-write privileges - RWUSER_PASS

- The database user password with read-only privileges - RUSER_PASS

You can execute OraDBI.jar using either of the following methods:

- Using Interactive Mode to Install Using ORADBI.jar

- Using the Command Line to Install Using ORADBI.jar

Using Interactive Mode to Install Using ORADBI.jar

The following procedure lists the steps to install the schema for Oracle Utilities Work and Asset Management v2.2.0.6.0 using OraDBI.

Run the following commands with the defined parameters on the command prompt from ..\WAM\Install-Upgrade directory:

1. Open a command line prompt in the Windows environment.

2. Set Java Home.

   In the following example, JDK 1.8 is installed in C:\Program Files\Java\jdk1.8.0_101 directory.

   ```
   SET JAVA_HOME=C:\Program Files\Java\jdk1.8.0_101
   ```

3. Set the class path.

   In the following example, the required jarfiles, including OraDBI.jar, are available in the C:\InstallUpgrade\Jarfiles* directory.

   ```
   SET CLASSPATH=C:\InstallUpgrade\Jarfiles\*
   ```

4. Execute the following command:

   ```
   "%JAVA_HOME%"\bin\java -Xmx1500M -cp %CLASSPATH%
   com.oracle.ouaf.oem.install.OraDBI -p <RWUSER_PASS>,<RUSER_PASS>
   ```

```
(or)
```

```
"C:\Program Files\Java\jdk1.8.0_101"\bin\java -Xmx1500M -cp
C:\InstallUpgrade\Jarfiles\* com.oracle.ouaf.oem.install.OraDBI -p
<RWUSER_PASS>,<RUSER_PASS>
```

The utility prompts you to enter values for the following parameters as per your environment:

- Name of the database server: <DB SERVER>

- Port no: <PORT>

- Name of the target database: <SID>

- Name of the owner of the database schema: <DBUSER>

- Password of the user name: <DBPASS>

- Oracle user with read-write privileges to the Database Schema: <CISUSER>

- Password of the user with read-write privileges: <DBPASS>

- Oracle user with read-only privileges to the Database Schema: <CISREAD>

- Password of the user with read-only privileges: <DBPASS>

- Oracle database role with read-write privileges to the Database Schema: <CIS_USER>

- Oracle database role with read-only privileges to the Database Schema: <CIS_READ>

- Enter the name of the target schema where you want to install or upgrade: <CISADM>

This process generates log files in the directory Install-Upgrade\logs. Check the log files for any errors. Refer to Installation Notes for more information.

You can also run the process on a Linux machine which has Java installed by exporting the JAVA HOME and CLASSPATH.

In interactive mode, you can run the installer using the command:

```
java -Xmx1500M com.oracle.ouaf.oem.install.OraDBI -p
<RWUSER_PASS>,<RUSER_PASS>
```

Using the Command Line to Install Using ORADBI.jar

Run the following commands with the defined parameters at the command prompt from the ..\WAM\Install-Upgrade directory

```
"C:\Program Files\Java\jdk1.8.0_65\bin\java" -Xmx1500M -cp
%CLASSPATH% com.oracle.ouaf.oem.install.OraDBI -d
jdbc:oracle:thin:@<DatabaseServer_Name>:1521/<DatabaseName>,
%DBUSER%,%DBPASS%, CISUSER, CISREAD, CIS_USER, CIS_READ, %DBUSER% -
l 1,2 -j "C:\Program Files\Java\jdk1.8.0_65" -p
<RWUSER_PASS>,<RUSER_PASS>
```

OR

```
"%JAVA_HOME%"\bin\java -Xmx1500M -cp %CLASSPATH%
com.oracle.ouaf.oem.install.OraDBI -d jdbc:oracle:thin:@<Database
ServerName>:1521/
```

```
<Database_Name>,%DBUSER%,%DBPASS%,CISUSER,CISREAD,CIS_USER,CIS_REA
D,%DBUSER% -l 1,2 -j "%JAVA_HOME%" -p <RWUSER_PASS>,<RUSER_PASS>
```

This process generates log files in the directory Install-Upgrade\logs. Check the log files for any errors. Refer to Installation Notes for more information.

This process can also be run on a Linux machine which has Java installed by exporting the JAVA HOME and CLASSPATH.

Tasks Performed by ORADBI

The following tasks are performed by ORADBI:

- Interacts with the user to collect information about the name of Oracle account that will own the application schema (for example, CISADM), password of this account, password of the SYSTEM account in the database, and the name of the Oracle account that the application user will use (for example, CISUSER), and the name of the Oracle account that will be assigned read-only privileges to the application schema (for example, CISREAD).

- Verifies whether tablespace names already exist in the Storage.xml file (if not, the process will abort).

- Installs the schema, installs the system data, and configures security.

- Maintains upgrade log tables in the database.

- Updates release ID when the upgrade is completed successfully.

- If an error occurs while executing a SQL script or another utility, it logs and displays the error message and allows you to re-execute the current step. Log files ORADBI###.log are created in the same folder as ORADBI and contains all the SQL commands executed against the database along with the results. The log files are incremental so that the results are never overwritten. If warning messages are generated during the upgrade, ORADBI prompts the user at the end of the process. Users should check the log files to verify the warning messages. Warning messages are only alerts and do not necessary mean a problem exists.

- Stores the Schema owner and password in the feature configuration table. The password is stored in encrypted format.

- OraDBI can be executed by a non-schema owner.

- OraDBI automatically runs the Security Oragensec utility.

Generating Database Statistics

During an install process new database objects may be added to the target database. Before starting to use the database, generate the complete statistics for these new objects by using the DBMS_STATS package.

Enable USER_LOCK package

For inbound web services to work, the USER_LOCK must be enabled at the database level. This is a one-time step. If this is not already enabled, enable it using the following steps:

1. Login as SYS user.

2. On SQL prompt run.

    ```
    @<ORACLE_HOME>/rdbms/admin/userlock.sql
    ```

3. Grant permission by running following SQL:

    ```
    grant execute on USER_LOCK to public;
    ```

Note that grant can also be made to the database user that the application connects to (for example, cisuser), instead of 'public'.

Chapter 5

Installing the Database - Demo Environment

This chapter provides instructions for installing the Oracle Utilities Work and Asset Management demo database. It describes how to install demo database components, including:

- Copying and Decompressing Install Media
- Creating the Database
- Importing the WAM Demo Dump File
- Configuring Security

Copying and Decompressing Install Media

To copy and decompress the database:

1. Download the Oracle Utilities Application Framework 4.4.0.2.0 - Database Installation Media, Oracle Utilities Application Framework v4.4.0.2.0 Single Fix Prerequisite Database Rollup for Oracle Utilities Work and Asset Management v2.2.0.6.0 (if there is any) and Oracle Utilities Work and Asset Management v2.2.0.6.0 Oracle Database Multiplatform from Oracle Software Delivery Cloud.

2. Unzip the Oracle Utilities Application Framework 4.4.0.2.0 - Database Installation Media file to a temporary folder. This file contains Oracle Utilities Application Framework folders with all the database components required to install the Oracle Utilities Application Framework 4.4.0.2.0 database.

3. Unzip the Oracle Utilities Application Framework v4.4.0.2.0 Single Fix Prerequisite Database Rollup for Oracle Utilities Work and Asset Management v2.2.0.6.0 to a temporary folder. This file contains Oracle Utilities Application Framework prerequisite single fix database patches for Oracle Utilities Work and Asset Management v2.2.0.6.0.

4. Unzip the Oracle Utilities Work and Asset Management v2.2.0.6.0 Oracle Database Multiplatform file to a temporary folder. This file contains Oracle Utilities Work and Asset Management folders with all the database components required to install the database.

    Note: You must have Oracle Database Server 12.2.0.1+ installed on your machine to continue with the procedure below. If you do not have Oracle Database Server 12.2.0.1+ installed, please upgrade to Oracle Database Server 12.2.0.1+ first before proceeding.

Creating the Database

It is strongly recommended that you use DBCA to create the database.

> **Note:** You must have Oracle Database Server Enterprise Edition installed on your machine in order to create the database.

> **UNIX**
> Create the database using the Database Configuration Assistant (DBCA).

> **Windows**
> You should be logged in as a user who is a member of the local ORA_DBA group on that server. The ORA_DBA group should have "administrator" privileges assigned to it.

> Make sure to set character set for database as AL32UTF8.

Refer to Chapter 1: Creating the Database for steps to create the database on either Unix or Windows.

Importing the WAM Demo Dump File

After a successful database creation, demo data can be imported by following these steps:

1. Set the correct ORACLE_SID and ORACLE_HOME.

2. Run the following command to import the demo dump:

    ```
    impdp directory= data_pump_dir dumpfile= expdp_WAM_demo.dmp
    logfile=expdp_WAM_demo.log schemas=CISADM
    ```

    Note: The data_pump_dir must exist in the database created above before continuing with the import. You should also copy the expdp_WAM_demo.dmp file to the data_pump_dir. Decompress the expdp_WAM_demo.dmp.gz file first to extract the expdp_WAM_demo.dmp file. This file is in the ..\WAM\Demo directory.

Configuring Security

For interactive mode execution of OraGenSec Java Version, set the JAVA_HOME and CLASSPATH environment variables as follows. Make sure to change path to your Java Home location and jarfiles location.

```
set JAVA_HOME=C:\Program Files\Java\jdk1.8.0_65
set CLASSPATH=C:\FW4.4.0.2.0\jarfiles\*
```

Use the -h option to display the options available for OraGenSec.

For example:

```
C:\ Install-Upgrade>java -Xmx1500M -cp %CLASSPATH%
com.oracle.ouaf.oem.install.OraGenSec -h
```

To execute OraGenSec use the following:

```
java -Xmx1500M -cp C:\FW4.4.0.2.0\jarfiles\*
com.oracle.ouaf.oem.install.OraGenSec  -l oragensec.log -d
<DBUSER>,<DBPASS>,jdbc:oracle:thin:@<DB_Server>:PORT/
<SERVICE_NAME> -a A -r <RW_USER_ROLE>,<R_USER_ROLE> -u
<RWUSER>,<RUSER> -p <RWUSER_PASS>,<RUSER_PASS>
```

Note: Database vault must be disabled before running this utility.

Chapter 6

Database Design

The standard for database objects such as tables, columns, and indexes, for products using the Oracle Utilities Application Framework helps in smooth integration and upgrade processes by ensuring clean database design, promoting communications, and reducing errors.

Just as Oracle Utilities Application Framework goes through innovation in every release of the software, it is also inevitable that the product will take advantage of various database vendors' new features in each release. The recommendations in the database installation section include only the ones that have been proved by vigorous QA processes, field tests and benchmarks.

This chapter describes the following:

- Database Object Standard
- Column Data Type and Constraints
- Standard Columns

Database Object Standard

This section discusses the rules applied to naming database objects and the attributes that are associated with these objects.

Categories of Data

A table can belong to one of the three categories:

- Control (admin)
- Master
- Transaction

For purposes of physical table space design, metadata and control tables can belong to the same category.

Example of tables in each category:

- **Control:** SC_USER, CI_ADJ_TYPE, F1_BUS_OBJ
- **Master:** CI_PER, CI_PREM
- **Transaction:** F1_FACT, CI_FT

All tables have the category information in their index name. The second letter of the index carries this information. Refer to Indexes for more information.

Naming Standards

The following naming standards must be applied to database objects.

Table

Table names are prefixed with the owner flag value of the product. For customer modification **CM** must prefix the table name. The length of the table names must be less than or equal to 30 characters. A language table should be named by suffixing **_L** to the main table. The key table name should be named by suffixing **_K** to the main table.

It is recommended to start a table name with the 2-3 letter acronym of the subsystem name that the table belongs to. For example, **MD** stands for metadata subsystem and all metadata table names start with **CI_MD**.

Some examples are:

- CI_ADJ_TYPE
- CI_ADJ_TYPE_L

A language table stores language sensitive columns such as a description of a code. The primary key of a language table consists of the primary key of the code table plus language code (LANGAGUE_CD).

A key table accompanies a table with a surrogate key column. A key value is stored with the environment id that the key value resides in the key table.

The tables prior to V2.0.0 are prefixed with CI_ or SC_.

Columns

The length of a column name must be less than or equal to 30 characters. For customer modification, CM must prefix the column name. The following conventions apply when you define special types of columns in the database.

- Use the suffix **FLG** to define a lookup table field. Flag columns must be CHAR(4). Choose lookup field names carefully as these column names are defined in the lookup table (CI_LOOKUP_FLD) and must be prefixed by the product owner flag value.

- Use the suffix **CD** to define user-defined codes. User-defined codes are primarily found as the key column of the admin tables.

- Use the suffix **ID** to define system assigned key columns.

- Use the suffix **SW** to define Boolean columns. The valid values of the switches are 'Y' or 'N'. The switch columns must be CHAR(1)

- Use the suffix **DT** to define Date columns.

- Use the suffix **DTTM** to define Date Time columns.

- Use the suffix **TM** to define Time columns.

Some examples are:

- ADJ_STATUS_FLG

- CAN_RSN_CD

Indexes

Index names are composed of the following parts:

[OF][*application specific prefix*][C/M/T]NNN[P/S]n

- **OF**- Owner Flag. The standard is to use the two characters of the product's owner flag. Note that there may be some older indexes that use only the first character of the owner flag. For client specific implementation of index, use CM for Owner Flag. If implementation creates a CM Index on table-columns for which the base product already provides an index, then the CM Index will be overridden by the based index.

- Application specific prefix could be C, F, T or another letter.

- **C/M/T** - The second character can be either C or M or T. C is used for control tables (Admin tables). M is for the master tables. T is reserved for the transaction tables.

- **NNN** - A three-digit number that uniquely identifies the table on which the index is defined.

- **P/S** - P indicates that this index is the primary key index. S is used for indexes other than primary keys.

- **n** is the index number, unique across all indexes on a given table (0 for primary and 1, 2, etc., for the secondary indexes).

Some examples are:

- F1C066P0

- F1C066S1

- CMT206S2

    Warning! Do not use index names in the application as the names can change due to unforeseeable reasons.

Updating Storage.xml

The storage.xml file that comes with the product allocates all base tables and indexes to the default tablespace CISTS_01. If you decide to allocate some tables or indexes outside of the default tablespace, then this has to be reflected in the storage.xml file by changing the tablespace name from the default value to a custom value, according to the format shown below:

Format:

```
<Table_Name>
  <TABLESPACE>CISTS_01</TABLESPACE>
  <PARALLEL>1</PARALLEL>
- <LOB>
- <Column Name>
  <TABLESPACE>CISTS_01</TABLESPACE>
  <SECUREFILE>Y</SECUREFILE>
  <CHUNK>8192</CHUNK>
  <CACHE>N</CACHE>
  <LOGGING>Y</LOGGING>
  <INROW>Y</INROW>
  <COMPRESS>N</COMPRESS>
  </Column Name>
  </LOB>
  </Table_Name>
```

Where Parallel defines the number of threads, that Oracle DB Server will use to access a table or create an index.

We recommend creating CLOBs stored as SECUREFILE with Medium compression and Cache enabled. Please note that by default, medium compression is turned-off and must only be enabled if you have the Advanced compression license.

For instance, if a DBA decided to allocate table CI_ACCT in a tablespace MyTablespace, then they would have to change the storage.xml as follows:

```
<CI_ACCT>
<TABLESPACE>MyTablespace</TABLESPACE>
</CI_ACCT>
```

The oradbi process uses the storage.xml file to place the new database objects into defined tablespaces. A tablespace referenced in the storage.xml file must exist in the database.

The storage.xml file has to be adjusted before each upgrade and/or new installation as required to allocate the tables and indexes across those tablespaces.

Table name is included as a comment for each of the indexes for clarity.

For initial installs, information for each object should be reviewed by a DBA. For upgrades, only tablespace information for the objects added in the new release needs to be reviewed by a DBA.

Be careful while editing this file. Make sure that the tablespace names being used exist in the database. Do not change the basic format of this file.

Sequence

The base sequence name must be prefixed with the owner flag value of the product. For customer modification **CM** must prefix the sequence name. The sequence numbers should be named as below:

1. If the Sequence is used for a specific table, then use the following sequence name:

   [OF][C/M/T]NNN_SEQ

   • OF stands for Owner Flag. For example, for Framework its F1 and for WAM and ODM it is W1.

   • C/M/T stands for Control (Admin)/Master/Transaction Tables.

   • NNN is a three digit unique Identifier for a table on which the sequence is defined.

   For e.g: F1T220_SEQ

2. If more than one sequence is used for a specific table, then use the following Sequence Name:

   [OF][C/M/T]NNN_Column_Name_SEQ

   • C/M/T stands for Control (Admin)/Master/Transaction tables.

   • NNN is a three digit unique identifier for a table on which the sequence is defined.

   For Example: F1T220_BO_STATUS_CD_SEQ and F1T220_BUS_OBJ_CD_SEQ

3. If sequence is used for a generic requirement and not specific to a table, then use the following sequence name.

   [OF]Column_Name_SEQ

   For Example: F1FKVALID_SEQ

   • For a customer modification, CM must prefix the sequence name.

Trigger

The base trigger name must be prefixed with the owner flag value of the product.

When implementers add database objects, such as tables, triggers and sequences, the name of the objects should be prefixed by CM.

Column Data Type and Constraints

This section discusses the rules applied to column data type and constraints, and the attributes that are associated with these objects.

User Defined Code

User Defined Codes are defined as CHAR type. The length can vary by the business requirements but a minimum of eight characters is recommended. You will find columns defined in less than eight characters but with internationalization in mind, new columns should be defined as CHAR(10) or CHAR(12). Also note that when the code is referenced in the application the descriptions are shown to users in most cases.

System Assigned Identifier

System assigned random numbers are defined as CHAR type. The length of the column varies to meet the business requirements. Number type key columns are used when a sequential key assignment is allowed or number type is required to interface with external software. For example, Notification Upload Staging ID is a Number type because most EDI software uses a sequential key assignment mechanism. For sequential key assignment implementation, the DBMS sequence generator is used in conjunction with Number Type ID columns.

Date/Time/Timestamp

Date, Time and Timestamp columns are defined physically as DATE in Oracle. Non-null constraints are implemented only for the required columns.

Number

Numeric columns are implemented as NUMBER type in Oracle. The precision of the number should always be defined. The scale of the number might be defined. Non-null constraints are implemented for all number columns.

Fixed Length/Variable Length Character Columns

When a character column is a part of the primary key of a table define the column in CHAR type. For the non-key character columns, the length should be the defining factor. If the column length should be greater than 10, use VARCHAR2 type in Oracle.

Null Column Support

The product supports Nullable columns. This means that the application can write NULLs instead of a blank space or zero (for numeric columns) by using NULLABLE_SW on CI_MD_TBL_FLD. If REQUIRED_SW is set to 'N' and the NULLABLE_SW is set to 'Y', the application will write a NULL in that column. The artifact generator will create hibernate mapping files with appropriate parameters so that the framework hibernate mapping types will know if a given property supports a null value.

NULLABLE_SW is not new, but has previously been used for certain fields such as dates, and some string and number foreign-key columns. Because of this, there is the possibility that there is incorrect metadata for some columns, and that turning on this new feature could result in incorrect behavior when using that metadata. The upgrade script fixes the metadata to make sure that the existing tables will not be affected.

This new feature only supports tables maintained by Java but NOT a Java program converted from COBOL. Thus, enhancing any existing tables to use null columns must be done only after making sure that the tables are maintained by Java, and not Java converted COBOL programs.

XML Type Support

The product supports XML Type. XML Type provides following advantages

1. The ability to use XQuery for querying nodes in the XML document stored within a column defined as XMLType.

2. The option to use the XML engine, which is built into the Oracle Database, to create indexes using nodes within the XML document stored in the XMLType column.

Cache and Key Validation Flags

By default, the Cache Flag is set to NONE. For most of the admin tables the CACHE Flag should be 'Cached for Batch'. This specifies that the table is cached as L2 cache to reduce database trips.

By default the Key Validation Flag is set to ALL. For tables which have the user defined keys, the KEY_VALIDATION_FLG should be set as 'ALL'. This checks the existence of the key before inserting a new one.

Table Classification and Table Volume Flags

There are multiple types of tables in the application, namely Admin system tables, Admin non-system tables, master tables and transaction tables. The Table Classification flag (TBL_CLASSIFICATION_FLG) sets the appropriate value for this lookup field to give a better view of the table classification.

Table Volume flag (TBL_VOLUME_FLG) is a customer modifiable field which is initially populated by product, but can be overridden by implementation. The field gives an idea of the relative data volume (categorized as highVolume, lowVolume and mediumVolume) of the table to make informed decisions.

Default Value Setting

The rules for setting the database default values are as follows:

- When a predefined default value is not available, set the default value of Non-null CHAR or VARCHAR columns to blank except the primary key columns.

- When a predefined default value is not available, set the default value Non-null Number columns to 0 (zero) except the primary key columns.

- No database default values should be assigned to the Non Null Date, Time, and Timestamp columns.

Foreign Key Constraints

Referential integrity is enforced by the application. In the database do not define FK constraints. Indexes are created on most of Foreign Key columns to increase performance.

Standard Columns

This section discusses the rules applied to standard columns and the attributes that are associated with these objects.

Owner Flag

Owner Flag (OWNER_FLG) columns exist on the system tables that are shared by multiple products. Oracle Utilities Application Framework limits the data modification of the tables that have owner flag to the data owned by the product.

Version

The Version column is used to for optimistic concurrency control in the application code. Add the Version column to all tables that are maintained by a Row Maintenance program.

Chapter 7

Database Implementation Guidelines

The following sections outline the general implementation guidelines for the database components:

- Configuration Guidelines
- Oracle Database Implementation Guidelines

Refer to My Oracle Support for more information.

Configuration Guidelines

This section describes the general recommendations for configuring various database objects and includes a brief syntax overview. It covers the general aspects of the database objects and does not cover any specific implementation requirements.

- Index
- Table Partitioning Recommendations
- Transparent Data Encryption Recommendations
- Data Compression Recommendations
- Database Vault Recommendations
- Oracle Fuzzy Search Support
- Storage Recommendations
- Database Configuration Recommendations
- Database Syntax
- Database Initialization Parameters

Index

Index recommendations specify points that need to be considered when creating indexes on a table.

1. Indexes on a table should be created according to the functional requirements of the table and not in order to perform SQL tuning.

2. The foreign keys on a table should be indexes.

> **Note:** If the implementation creates a CM index on table-columns where the product already provides an index, then the CM index will be overridden by the base index.

Table Partitioning Recommendations

Oracle Utilities recommends using a minimum of 'n' partitions for selective database objects, where 'n' is number of RAC nodes.

Transparent Data Encryption Recommendations

Oracle Utilities supports Oracle Transparent Data Encryption (TDE). Oracle 11gR1 supports tablespace level encryption. The application supports tablespace level encryption for all application data. Make sure that the hardware resources are sufficiently sized for this as TDE uses additional hardware resources. The Oracle Advanced Security license is a prerequisite for using TDE.

Please consider the following when implementing TDE:

- Create a wallet folder to store the master key. By default, the wallet folder should be created under $ORACLE_BASE/admin/<sid>.

- The wallet containing the master key can be created using the following command:

```
alter system set encryption key authenticated by "keypasswd"
```

- The wallet can be closed or opened using the following commands:

```
alter system set wallet open identified by "keypasswd";
alter system set wallet close;
```

- Column level encryption can be achieved using the following commands:

```
create table <table_name>
(name varchar2(200) default ' ' not null,
bo_data_area CLOB encrypt using 'AES128',
bo_status_cd char(12)    encrypt using 'AES128')
lob (bo_data_area) store as securefile (cache compress)
tablespace <tablespace_name>;
```

- AES128 is the default encryption algorithm.

- Tablespace level encryption is also supported using the following command:

```
Create tablespace <tablespace_name> logging datafile '<datafile
location>' size <initial size> reuse autoextend on next <next
size>
maxsize unlimited extent management local uniform size
<uniform size> encryption using 'AES128' default
storage(encrypt);
```

- Indexed columns can only be encrypted using the NO SALT Option. Salt is a way to strengthen the security of encrypted data. It is a random string added to the data before it is encrypted, causing repetition of text in the clear to appear different when encrypted.

Data Compression Recommendations

Oracle Utilities supports Advanced Data Compression, available with Oracle 11gR1 onwards, to reduce the database storage footprint. Make sure that your resources are sufficiently sized for this as it uses additional system resources. Compression can be enabled at the Tablespace level or at the Table level.

Exadata Hardware

For Exadata hardware the compression recommendations are:

- For high volume tables, keep the current table partition uncompressed. All of the older partitions will be compressed based on QUERY HIGH compression.

- For high volume tables with CLOBs, always keep the CLOBs in securefiles with MEDIUM compression. Also keep the current table partition uncompressed. All of the older partitions will be compressed based on QUERY HIGH compression.

- Load data into the uncompressed table partitions using a conventional load and then, once data is loaded using a CTAS operation, load into a temporary heap table. Then truncate the original partition. Alter the original partition into HCC compressed and then partition exchange this with the temporary heap table.

- All multi column Indexes (primary as well as secondary) will be compressed using the default compression. HCC or OLTP compression is not applicable on the top of compressed Indexes.

Non- Exadata Hardware

For non-Exadata hardware the recommendations are the same as above, except that you cannot use HCC compression (it is only available in Exadata database machine). Instead of HCC you can use any other compression tool available to you for non-Exadata hardware.

CLOB Fields

All CLOB fields should be stored as SecureFiles and Medium compressed. This requires a separate license for Advanced Data Compression. As a part of the schema, we create the product-owned tables with compression turned OFF at the LOB level. If you have the license for Advanced Data Compression, you can enable compression by updating the storage.xml.

Compression Guidelines

- Admin and Metadata tables and their indexes will NOT be compressed.

- All Transactional Tables will be compressed.
  This includes ILM enabled MOs where applicable.

- Compression will be done at the tablespace level.

  - Different MOs will have different tablespaces.

  - Partitioned MOs will have one tablespace per partition.

  - Child tables will use reference partitioning with parent + children sharing the same tablespace. (parent and child will always be managed/archived together).

- All multicolumn indexes on transactional tables will be compressed.

  - Use 'compress advanced low'.

  - Local partitioned indexes will reside in the same tablespace as the table.

  - Each MO will have an index tablespace. All MO (Parent-Child Table) indexes will share this tablespace.

  - Do NOT specify standard index compression.

- Securefile medium compression in row for LOBs and CLOBs.

Database Vault Recommendations

The product supports Database Vault. All non-application User IDs can be prevented from using DDL or DML statements against the application schema. So SYS and SYSTEM cannot issue DDL or DML statements against CISADM schema.

The application-specific administration account can issue DDL statements but should not be able to perform any DML or DCL statements.

Application user must be given DML only permissions.

Database Vault can be used to control access during patch process and Install/Upgrade process.

Oracle Fuzzy Search Support

The product supports Oracle Fuzzy searches. To use this feature, Oracle Text must be installed. After Oracle Text is installed, an index must be created on the table where the fuzzy search needs to be performed from the application. This is only an Oracle database option and is not supported by other databases. Additionally, not all languages are supported. Refer to the Oracle database documentation for more information about fuzzy searching.

A typical syntax for implementation of fuzzy searching is as below. For the most updated syntax, please refer to Oracle Fuzzy documentation.

```
GRANT CTXAPP TO <Application schema owner e.g CISADM>;

GRANT EXECUTE ON CTX_DDL TO <Application schema owner e.g CISADM>;

create index <Application schema owner e.g CISADM>.<Index_Name> on
Application schema owner e.g CISADM>.<Table_Name> (<column_name>)
indextype is ctxsys.context parameters ('sync (on commit)');
begin
ctx_ddl.sync_index('Application schema owner e.g
CISADM>.<Index_Name>');
end
/
 chapter
```

Storage Recommendations

This section specifies recommended options for storing the database objects.

SecureFile for Storing LOBs

Beginning with Oracle 11g, tables having fields with data type of CLOB or BLOBS should have the LOB Columns stored as SecureFiles.

- The storage options with SecureFiles for Heap Tables should be ENABLE STORAGE IN ROW, CACHE and COMPRESS.

- For the IOT Table the PCTTHRESHOLD 50 OVERFLOW clause should be specified and the storage options with SecureFiles should be ENABLE STORAGE IN ROW, CACHE and COMPRESS.

- The PCTTHRESHOLD should be specified as a percentage of the block size. This value defines the maximum size of the portion of the row that is stored in the Index block when an overflow segment is used.

- The CHUNK option for storage, which is the data size used when accessing or modifying LOB values, can be set to higher than one database block size if big LOBs are used in the IO Operation.

- For SecureFiles, make sure that the initialization parameter db_securefile is set to ALWAYS.

- The Tablespace where you are creating the SecureFiles should be enabled with Automatic Segment Space Management (ASSM). In Oracle Database 11g, the default mode of Tablespace creation is ASSM so it may already be set for the Tablespace. If it's not, then you have to create the SecureFiles on a new ASSM Tablespace.

Note: To enable compression on SecureFiles, you must have an Oracle Advanced Compression license in addition to Oracle Database Enterprise Edition. This feature is not available for the standard edition of the Oracle database.

If you are using Oracle Database Enterprise Edition, please verify that the "COMPRESS" flag is turned on by setting it to "Y" in Storage.xml.

Refer to the Database Syntax section for more information on SecureFiles.

Database Configuration Recommendations

This section specifies the recommended methods for configuring the database with a focus on specific functional area.

Large Redo Log File Sizes

The Redo Log files are written by the Log Writer Background process. These log files are written in a serial manner. Once a log File is full, a log switch occurs and the next log file starts getting populated.

It is recommended that the size of the Redo log files should be sufficiently high so that you do not see frequent Log Switches in the alert logs of the database. Frequent Log Switches impact the IO performance and can be avoided by having a larger Redo log file size.

Frequent Log Switches impacts the IO performance and can be avoided by having a bigger Redo log File Size.

Database Syntax

SecureFile

```
CREATE TABLE <Table_Name>
     ( COLUMN1 ….,
       COLUMN2 (CLOB)
     )
LOB(COLUMN2) STORE AS SECUREFILE (CACHE COMPRESS);

CREATE TABLE <Table_Name>
     ( COLUMN1 ….,
       COLUMN2 (CLOB)
       CONTRAINT <> PRIMARY KEY(…)
     )
ORGANIZATION INDEX PCTTHRESHOLD 50 OVERFLOW
LOB(COLUMN2) STORE AS SECUREFILE (ENABLE STORAGE IN ROW CHUNK CACHE
COMPRESS);
```

Database Initialization Parameters

The recommended initialization parameters are given below. These parameters are a starting point for database tuning. An optimal value for a production environment may differ from one customer deployment to another.

db_block_size=8192

log_checkpoint_interval=0

db_file_multiblock_read_count=8

transactions=3000

open_cursors=3000

db_writer_processes=10

db_files=1024

dbwr_io_slaves=10   (Only if Asynchronous IO is not Supported)

sessions=4500

memory_target=0

memory_max_target=0

processes=3000

dml_locks=48600

_b_tree_bitmap_plans=FALSE

Oracle Database Implementation Guidelines

This section provides specific guidelines for implementing the Oracle database.

Oracle Partitioning

If you use a base index for the partitioning key, rename the index to CM**.

If you use the primary key index of the table as the partitioning key:

- Make the index non-unique.
- Primary constraints should still exist.

The upgrade on the partitioned table works best if the partitioning key is not unique. This allows the upgrade tool to drop the PK constraints if the primary key columns are modified and recreate the PK constraints without dropping the index.

Database Statistic

During an install process, new database objects may be added to the target database. Before starting to use the database, generate the complete statistics for these new objects by using the DBMS_STATS package. You should gather statistics periodically for objects where the statistics become stale over time because of changing data volumes or changes in column values. New statistics should be gathered after a schema object's data or structure are modified in ways that make the previous statistics inaccurate. For example, after loading a significant number of rows into a table, collect new statistics on the number of rows. After updating data in a table, you do not need to collect new statistics on the number of rows, but you might need new statistics on the average row length.

A sample syntax that can be used is as following:

```
BEGIN
SYS.DBMS_STATS.GATHER_SCHEMA_STATS (
OwnName => 'CISADM'
,Degree => 16
,Cascade => TRUE
,Method_opt => 'FOR ALL COLUMNS SIZE AUTO'
, Granularity => 'ALL' );
END;
/
```

Materialized View

Oracle Enterprise Edition supports query rewrite Materialized view. If you use Oracle Enterprise Edition, you can create following Materialized Views to improve performance of the Monitor batch jobs.

Prerequisites

Make sure to set up the following:

1. Set parameter QUERY_REWRITE_ENABLED=TRUE at database level.

    ```
    ALTER SYSTEM SET QUERY_REWRITE_ENABLED=TRUE; OR
    ALTER SYSTEM SET QUERY_REWRITE_ENABLED=TRUE SCOPE=BOTH;
    ```

2. To create a materialized view in another user's schema you must have the **CREATE ANY MATERIALIZED VIEW** system privilege. The owner of the materialized view must have the CREATE TABLE system privilege. The owner must also have access to any master tables of the materialized view that the schema owner does not own (for example: if the master tables are on a remote database) and to any materialized view logs defined on those master tables, either through a **SELECT** object privilege on each of the tables or through the **SELECT ANY TABLE** system privilege.

3. To create a refresh-on-commit materialized view (**ON COMMIT REFRESH** clause), in addition to the preceding privileges, you must have the **ON COMMIT REFRESH** object privilege on any master tables that you do not own or you must have the **ON COMMIT REFRESH** system privilege.

    To create the materialized view with query rewrite enabled, in addition to the preceding privileges: If the schema owner does not own the master tables, then the schema owner must have the **GLOBAL QUERY REWRITE** privilege or the **QUERY REWRITE** object privilege on each table outside the schema.

To debug materialized views, refer the below URLs:

- Oracle 11g - https://docs.oracle.com/cd/B28359_01/server.111/b28313/qrbasic.htm

- Oracle 12c - https://docs.oracle.com/database/121/DWHSG/qrbasic.htm#DWHSG01813

- Troubleshoot Materialized View - http://docs.oracle.com/database/121/ARPLS/d_mview.htm#ARPLS67193

```
CREATE MATERIALIZED VIEW F1_BO_LIFECYCLE_STATUS_MVW
(
BUS_OBJ_CD,
LIFE_CYCLE_BO_CD,
BO_STATUS_CD,
BATCH_CD
)
BUILD IMMEDIATE REFRESH ON COMMIT ENABLE QUERY REWRITE AS
SELECT
BO2.BUS_OBJ_CD,BO.LIFE_CYCLE_BO_CD,BOSA.BO_STATUS_CD,LCBOS.BATCH_CD as
LC_BATCH_CD
FROM
F1_BUS_OBJ BO2,
F1_BUS_OBJ BO,
F1_BUS_OBJ_STATUS LCBOS,
F1_BUS_OBJ_STATUS_ALG BOSA
WHERE
BO2.LIFE_CYCLE_BO_CD =BO.LIFE_CYCLE_BO_CD AND
BO.BUS_OBJ_CD = BOSA.BUS_OBJ_CD AND
BOSA.BO_STATUS_SEVT_FLG = 'F1AT' AND
LCBOS.BUS_OBJ_CD = BO.LIFE_CYCLE_BO_CD AND
LCBOS.BO_STATUS_CD = BOSA.BO_STATUS_CD
/
```

create synonym SPLUSR.F1_BO_LIFECYCLE_STATUS_MVW for SPLADM.F1_BO_LIFECYCLE_STATUS_MVW;

grant select on F1_BO_LIFECYCLE_STATUS_MVW to FW_DEV;

grant select on F1_BO_LIFECYCLE_STATUS_MVW to SPL_USER;

grant select on F1_BO_LIFECYCLE_STATUS_MVW to SPL_READ;

For more information, refer to the following documents:

- Basic Query Rewrite (Oracle 11g) - https://docs.oracle.com/cd/B28359_01/server.111/b28313/qrbasic.htm

- Basic Query Rewrite for Materialized Views (Oracle 12c) - https://docs.oracle.com/database/121/DWHSG/qrbasic.htm#DWHSG01813

- Troubleshooting Materialized Views - http://docs.oracle.com/database/121/ARPLS/d_mview.htm#ARPLS67193

- Debugging materialized Views - http://docs.oracle.com/cd/B28359_01/server.111/b28313/qradv.htm

Known Issues

The following are some of the known issues at the time of release. For more information, refer to these articles on My Oracle Support:

- Query Did Not Rewrite For A User Other Than The Owner Of the Materialized View (Doc ID 1594725.1) - A patch is available for bug report 14772096 for some platforms.

- Query rewrite not working as expected with SELECT DISTINCT (Doc ID 7661113.8) for Oracle version – 11.2.0.1 and 11.1.0.7 Fixed in version - 12.1.0.1 (Base Release), 11.2.0.2 (Server Patch Set)

Appendix A

Upgrades to the Oracle Utilities Work and Asset Management 2.2.0.6.0 Database

This section describes the database upgrade process for Oracle Utilities Work and Asset Management v2.2.0.6.0. It highlights changes made to the administrative tables and how those changes should be applied to the data in order for your current database to work with the v2.2.0.6.0 application, and to preserve the business logic implemented in the previous version of the application. The changes that do not require data upgrade are not described in this section of the document. The tasks that need to be performed after running the upgrade scripts are included.

The added functionality of v2.2.0.6.0 is not the scope of this documentation. The upgrade scripts do not turn on the newly added functionality by default. For new functionality, refer to the *Oracle Utilities Work and Asset Management User Guides* and *Release Notes*.

New System Data

This section lists the new system data that are added for business process configuration.

New Tables

There are no new tables in this Oracle Utilities Work and Asset Management release.

Dropped Tables

There are no dropped tables in this Oracle Utilities Work and Asset Management release.

Table_Name	Column_name	Description
W1_ACTIVITY_TYPE	ACT_INACT_FLG	CHAR(4)
W1_ACT_RESRC_REQMT	BC_LINE_ID	CHAR(14)
W1_ASSET	ASSET_OWNERSHIP _FLG	CHAR(4)
W1_PERMIT_TMPL_CHKLST	SEQ_NUM	NUMBER(3)

Updated Columns

There are no updated columns in this Oracle Utilities Work and Asset Management release.

Dropped Fields

The following fields are dropped in this Oracle Utilities Work and Asset Management this release:

Table_Name	Column_Name
W1_ASSET	OWNER_FLG

Dropped Tables

There are no dropped tables in this Oracle Utilities Work and Asset Management release.

Table_Name	Index	Column_Name
W1_ASSET	W1M00757	SPECIFICATION_CD

Updated Indexes

There are no updated indexes in this Oracle Utilities Work and Asset Management release.

Dropped Indexes

No indexes are dropped in this Oracle Utilities Work and Asset Management release.

New Views

The following views are added in this Oracle Utilities Work and Asset Management release:

ViewName
W1_BI_ASSETACTCOST_VW
W1_BI_ASSETCOST_VW
W1_BI_ASSET_BN_VW
W1_BI_ASTAVAILDWNTM_VW
W1_BI_ASTAVAILWO _VW
W1_BI_NODE_HIERARCHY_VW

New Materialized Views

The following materialized views are added in this Oracle Utilities Work and Asset Management release:

ViewName
W1_LEAF_ORG_MT_VW
W1_PARENT_ORG_MT_VW

Appendix B

Upgrades to the Oracle Utilities Application Framework Database

This section describes the database upgrade process for the Oracle Utilities Application Framework database since the last release. It highlights changes made to the administrative tables and how those changes should be applied to the data in order for your current database to work with the Oracle Utilities Application Framework application, and to preserve the business logic implemented in the previous version of the application. The changes that do not require data upgrade are not described in this document. The tasks that need to be performed after running the upgrade scripts are included.

> **Note**: Upgrade scripts do not automatically enable the newly added functionality by default. Please refer to the release notes for more information.

- Upgrades to Oracle Utilities Application Framework v4.3.0.3.0

- Upgrades to Oracle Utilities Application Framework v4.3.0.4.0

- Upgrades to Oracle Utilities Application Framework v4.3.0.5.0

- Upgrades to Oracle Utilities Application Framework v4.3.0.6.0

- Upgrades to Oracle Utilities Application Framework v4.4.0.0.0

- Upgrades to Oracle Utilities Application Framework v4.4.0.2.0

Upgrades to Oracle Utilities Application Framework v4.3.0.3.0

New Tables

Table	Type of Table
F1_LGCY_OBJ	Legacy Object
F1_PERF_TGT	Performance Target
F1_PERF_TGT_CHAR	Performance Target Characteristic
F1_PERF_TGT_L	Performance Target Language
F1_PERF_TGT_LOG	Performance Target Log
F1_PERF_TGT_LOG_PARM	Performance Target Log Parameter
F1_PERF_TGT_REL_OBJ	Performance Target Related Object
F1_PERF_TGT_TYPE	Performance Target Type
F1_PERF_TGT_TYPE_CHAR	Performance Target Type Characteristic
F1_PERF_TGT_TYPE_L	Performance Target Type Language
F1_STATS	Statistics Control
F1_STATS_CHAR	Statistics Control Characteristic
F1_STATS_L	Statistics Control Language
F1_STATS_LOG	Statistics Control Log
F1_STATS_LOG_PARM	Statistics Control Log Parameter
F1_STATS_REL_OBJ	Statistics Control Related Object
F1_STATS_SNPSHT	Statistics Snapshot
F1_STATS_SNPSHT_CHAR	Statistics Snapshot Characteristic
F1_STATS_SNPSHT_LOG	Statistics Snapshot Log
F1_STATS_SNPSHT_LOG_PARM	Statistics Snapshot Log Parameter
F1_STATS_SNPSHT_REL_OBJ	Statistics Snapshot Related Object
F1_SVC_CATALOG	Web Service Catalog

New Views
None

Dropped Tables
None

Unsupported Tables
None

Added Columns

Table	Column	Required
F1_EXTSYS_OUTMSG_PROF	NAMESPACE_FLG	N
F1_EXTSYS_OUTMSG_PROF	WSDL_FILE_NAME	N

Dropped Columns

None

Unsupported Table Columns

None

Column Format Change

None

Primary Key Change

None

Upgrades to Oracle Utilities Application Framework v4.3.0.4.0

New Tables

Table	Type of Table
F1_MIGR_REQ_INCL_REQ	Migration request Grouping

New Views
None

Dropped Tables
None

Unsupported Tables
None

Added Columns

Table	Column	Required
CI_BATCH_CTRL	APP_SVC_ID	Y
CI_XAI_RCVR_CTX	SEQNO	Y
CI_XAI_SNDR_CTX	SEQNO	Y
F1_IWS_SVC_ANN	SEQ_NUM	Y
F1_MIGR_REQ	MIGR_REQ_CAT_XFLG	N
F1_MIGR_REQ	MIGR_REQ_CLASS_FLG	Y
F1_MIGR_REQ_INSTR_ENTITY	COMMENT_LONG	N
F1_MIGR_REQ_INSTR_ENTITY	EXT_REFERENCE_ID	N

Dropped Columns

Table	Column
CI_XAI_RCVR_CTX	CTXT_VAL

Unsupported Table Columns

None

Column Format Change

Table Name	Column Name	From	To
F1_EXT_LOOKUP_ VAL_CHAR	F1_EXT_LOOKUP_ VALUE	VARCHAR2 (30)	VARCHAR2 (254)

Primary Key Change

Table	Primary Key Columns
CI_XAI_RCVR_CTX	XAI_RCVR_ID, SEQNO
CI_XA_SNDR_CTX	XAI_SENDER_ID, SEQNO

Upgrades to Oracle Utilities Application Framework v4.3.0.5.0

New Tables

Table	Description	Type of Table
F1_DEPLOYMENT	Deployment	Transaction
F1_DEPLOYMENT_ITEM	Deployment Item	Transaction
F1_DEPLOYMENT_ITEM_METADATA	Deployment Item Meta Data	Transaction
F1_DEPLOYMENT_PART	Deployment Part	Master
F1_DEPLOYMENT_PART_L	Deployment Part Language	Master
F1_DEPLOYMENT_TYPE	Deployment Type	Master
F1_DEPLOYMENT_TYPE_L	Deployment Type Language	Master
F1_DEPTYP_DEPPART	Deployment Type / Deployment Part	Transaction
F1_DEPTYP_MDT_TYPE	Deployment Type / MDT Type	Transaction
F1_DEPTYP_MSG_CAT	Deployment Type Message Category	Transaction
F1_DEPTYP_USR_GRP	Deployment Type User Group	Transaction
F1_MDT	Mobile Data Terminal	Transaction
F1_MDT_CHAR	Mobile Data Terminal Characteristics	Transaction
F1_MDT_TYPE	Mobile Data Terminal Type	Master
F1_MDT_TYPE_CHAR	Mobile Data Terminal Type Characteristics	Master
F1_MDT_TYPE_L	Mobile Data Terminal Type Language	Master
F1_MOB_COMP_CHAR	Mobile Component Characteristics	Admin - System
F1_MOB_COMP_CNT	Mobile Component Content	Admin - System
F1_MOBILE_COMPONENT	Mobile Component	Admin - System
F1_MOBILE_COMPONENT_L	Mobile Component Language	Admin - System

F1_REMOTE_MSG	Remote Message	Transaction
F1_REMOTE_MSG_CHAR	Remote Message Characteristics	Transaction
F1_REMOTE_MSG_LOG	Remote Message Log	Transaction
F1_REMOTE_MSG_LOG_PARM	Remote Message Log Parameters	Transaction
F1_WEB_CAT_L	Web Service Category Language	Admin - System
F1_WEB_CAT_INCL_SVC	Web Service Category - Included Services	Admin - System
F1_WEB_CAT	Web Service Category	Admin - System

Note that in addition, the following table was added to 4.3.0.4.0 via a hot fix, but was not included in 4.3.0.5.0 until after the final build and is therefore added as a hot fix. Clients upgrading to 4.3.0.5.0 may see that the table is dropped via the blueprint and then reinstated after applying the bug fixes.

Table	Description	Type of Table
F1_MIGR_OBJ_SQL_PK	Migration Object SQL Primary Key	Transaction

New Views
None

Dropped Tables

Table
F1_IWS_ANN_CHAR
F1_IWS_ANN_TYPE_CHAR

Unsupported Tables
None

Added Columns

Table	Column	Required
CI_MD_SVC	APP_SVC_ID	N
F1_OUTMSG	BO_XML_DATA_AREA	N
F1_OUTMSG_TYPE	OUTMSG_PRIOR_FLG	Y
F1_OUTMSG_TYPE	OWNER_FLG	N
F1_OUTMSG_TYPE	TYPE_BUS_OBJ_CD	N

Table	Column	Required
F1_OUTMSG_TYPE_L	OWNER_FLG	N

Dropped Columns

None

Unsupported Table Columns

None

Column Format Change

None

Primary Key Change

None

Index Changes

Index S1C675S1 for table F1_EXT_LOOKUP_VAL_CHAR has been renamed to
F1C675S1.

Upgrades to Oracle Utilities Application Framework v4.3.0.6.0

New Tables

Table	Description	Table Type
F1_ATTACHMENT_K	Attachment Key	Transaction
F1_CRYPTO_KEY	Key Ring Key	Admin
F1_CRYPTO_KEY_RING	Key Ring	Admin
F1_CRYPTO_KEY_RING_L	Key Ring Language	Admin
F1_CRYPTO_KEY_RING_LOG	Key Ring Log	Admin
F1_CRYPTO_KEY_RING_LOG_PARM	Key Ring Log Parameter	Admin
F1_CUBE_TYPE	Cube Type	Admin
F1_CUBE_TYPE_L	Cube Type Language	Admin
F1_CUBE_VIEW	Cube View	Transaction
F1_CUBE_VIEW_L	Cube View Language	Transaction
F1_CUBE_VIEW_LOG	Cube View Log	Transaction
F1_CUBE_VIEW_LOG_PARM	Cube View Log Parameters	Transaction
F1_DEPLOYMENT_K	Deployment Key	Transaction
F1_ERASURE_SCHED	Object Erasure Schedule	Transaction
F1_ERASURE_SCHED_K	Object Erasure Schedule Key	Transaction
F1_ERASURE_SCHED_LOG	Object Erasure Schedule Log	Transaction
F1_ERASURE_SCHED_LOG_PARM	Object Erasure Schedule Log Parameter	Transaction
F1_MDT_K	Mobile Data Terminal Key	Transaction
F1_MIGR_OBJ_SQL_PK	Migration Object SQL Primary Key	Transaction
F1_PROC_DEFN	Process Flow Type	Admin
F1_PROC_DEFN_L	Process Flow Type Language	Admin
F1_PROC_NEXT_PANEL	Next Panel	Admin
F1_PROC_PANEL	Process Panel	Admin
F1_PROC_STORE	Process Flow	Transaction

Table	Description	Table Type
F1_PROC_STORE_DTL_ELEMENTS	Process Flow Detail Elements	Transaction
F1_PROC_STORE_K	Process Flow Key	Transaction
F1_PROC_STORE_LOG	Process Flow Log	Transaction
F1_PROC_STORE_LOG_PARM	Process Flow Log Parameters	Transaction
F1_REMOTE_MSG_K	Mobile Remote Message Key	Transaction
F1_STATS_SNPSHT_K	Statistics Snapshot Key	Transaction

Note that the following tables have system generated keys but do not have a separate key table. Per the new table list, the key tables are provided and these tables are updated accordingly.

Table	Description
F1_ATTACHMENT	Attachment
F1_DEPLOYMENT	Deployment
F1_MDT	Mobile Data Terminal
F1_REMOTE_MSG	Mobile Remote Message
F1_STATS_SNPSHT	Statistics Snapshot

New Views

None

Dropped Tables

Table
F1_IWS_SVC_OPER_L

Unsupported Tables

The table below has been added for future functionality and is not currently in use.

Table
F1_CRYPTO_KEY_RING_LINK

Added Columns

Table	Column	Required
CI_BATCH_RUN	END_DTTM	N

Table	Column	Required
CI_BATCH_RUN	START_DTTM	N
CI_COUNTRY	ADDR1_USG_FLG	Y
CI_COUNTRY	ADDR2_USG_FLG	Y
CI_COUNTRY	ADDR3_USG_FLG	Y
CI_COUNTRY	ADDR4_USG_FLG	Y
CI_COUNTRY	CITY_USG_FLG	Y
CI_COUNTRY	COUNTY_USG_FLG	Y
CI_COUNTRY	GEO_CODE_USG_FLG	Y
CI_COUNTRY	HOUSE_TYPE_USG_FLG	Y
CI_COUNTRY	IN_CITY_LIM_USG_FLG	Y
CI_COUNTRY	NUM1_USG_FLG	Y
CI_COUNTRY	NUM2_USG_FLG	Y
CI_COUNTRY	POSTAL_USG_FLG	Y
CI_COUNTRY	STATE_USG_FLG	Y
F1_ATTACHMENT	ATTACHMENT_EXT_ID	N
F1_ATTACHMENT	COMMENT_LONG	N
F1_IWS_SVC	RESOURCE_CAT_XFLG	N
F1_IWS_SVC	WEB_SVC_CLASS_FLG	Y
F1_IWS_SVC_OPER	RESOURCE_URI	N
F1_IWS_SVC_OPER	REST_HTTP_METHOD_FLG	N
F1_SVC_CATALOG	WEB_SVC_CLASS_FLG	Y

Dropped Columns

None

Column Format Change

None

Primary Key Change

None

Index Changes

None

New Tables

Table	Description	Type of Table
F1_DEPLOYMENT	Deployment	Transaction
F1_DEPLOYMENT_ITEM	Deployment Item	Transaction
F1_DEPLOYMENT_ITEM_METADATA	Deployment Item Meta Data	Transaction
F1_DEPLOYMENT_PART	Deployment Part	Master
F1_DEPLOYMENT_PART_L	Deployment Part Language	Master
F1_DEPLOYMENT_TYPE	Deployment Type	Master
F1_DEPLOYMENT_TYPE_L	Deployment Type Language	Master
F1_DEPTYP_DEPPART	Deployment Type / Deployment Part	Transaction
F1_DEPTYP_MDT_TYPE	Deployment Type / MDT Type	Transaction
F1_DEPTYP_MSG_CAT	Deployment Type Message Category	Transaction
F1_DEPTYP_USR_GRP	Deployment Type User Group	Transaction
F1_MDT	Mobile Data Terminal	Transaction
F1_MDT_CHAR	Mobile Data Terminal Characteristics	Transaction
F1_MDT_TYPE	Mobile Data Terminal Type	Master
F1_MDT_TYPE_CHAR	Mobile Data Terminal Type Characteristics	Master
F1_MDT_TYPE_L	Mobile Data Terminal Type Language	Master
F1_MOB_COMP_CHAR	Mobile Component Characteristics	Admin - System
F1_MOB_COMP_CNT	Mobile Component Content	Admin - System
F1_MOBILE_COMPONENT	Mobile Component	Admin - System
F1_MOBILE_COMPONENT_L	Mobile Component Language	Admin - System
F1_REMOTE_MSG	Remote Message	Transaction
F1_REMOTE_MSG_CHAR	Remote Message Characteristics	Transaction

F1_REMOTE_MSG_LOG	Remote Message Log	Transaction
F1_REMOTE_MSG_LOG_PARM	Remote Message Log Parameters	Transaction
F1_WEB_CAT_L	Web Service Category Language	Admin - System
F1_WEB_CAT_INCL_SVC	Web Service Category - Included Services	Admin - System
F1_WEB_CAT	Web Service Category	Admin - System

Note that in addition, the following table was added to 4.3.0.4.0 via a hot fix, but was not included in 4.3.0.5.0 until after the final build and is therefore added as a hot fix. Clients upgrading to 4.3.0.5.0 may see that the table is dropped via the blueprint and then reinstated after applying the bug fixes.

Table	Description	Type of Table
F1_MIGR_OBJ_SQL_PK	Migration Object SQL Primary Key	Transaction

New Views

None

Dropped Tables

Table
F1_IWS_ANN_CHAR
F1_IWS_ANN_TYPE_CHAR

Unsupported Tables

None

Added Columns

Table	Column	Required
CI_MD_SVC	APP_SVC_ID	N
F1_OUTMSG	BO_XML_DATA_AREA	N
F1_OUTMSG_TYPE	OUTMSG_PRIOR_FLG	Y
F1_OUTMSG_TYPE	OWNER_FLG	N
F1_OUTMSG_TYPE	TYPE_BUS_OBJ_CD	N
F1_OUTMSG_TYPE_L	OWNER_FLG	N

Dropped Columns

None

Unsupported Table Columns

None

Column Format Change

None

Primary Key Change

None

Index Changes

Index S1C675S1 for table F1_EXT_LOOKUP_VAL_CHAR has been renamed to
F1C675S1.

Upgrades to Oracle Utilities Application Framework v4.4.0.0.0

New Tables
None

New Views
None

Dropped Tables
None

Unsupported Tables
None

Added Columns
None

Dropped Columns
None

Column Format Change
None

Primary Key Change
None

Index Changes
create index XT039S8 on
CI_TD_ENTRY(ENTRY_STATUS_FLG,TD_TYPE_CD,MESSAGE_CAT_NBR,MESSAGE_NBR)

Upgrades to Oracle Utilities Application Framework v4.4.0.2.0

New Tables

None

New Views

None

Dropped Tables

None

Unsupported Tables

None

Added Columns

None

Dropped Columns

None

Column Format Change

None

Primary Key Change

None

Index Changes

create index XT039S8 on
CI_TD_ENTRY(ENTRY_STATUS_FLG,TD_TYPE_CD,MESSAGE_CAT_NBR,MESSAGE_NBR)

Appendix C

Oracle Application Framework System Table Guide

This section lists the system tables owned by the Oracle Utilities Application Framework V4.4.0.2.0 and explains the data standards of the system tables. The data standards are required for the installation of Oracle Utilities Application Framework, development within the Oracle Utilities Application Framework, and the configuration and customization of Oracle Utilities products. Adhering to the data standards is a prerequisite for seamless upgrade to future releases.

This section includes:

- About the Application Framework System Tables
- System Table Standards
- Guidelines for System Table Updates
- System Table List

About the Application Framework System Tables

System tables are a subset of the tables that must be populated at the time the product is installed. They include metadata and configuration tables. The data stored in the system tables are the information that Oracle Utilities Application Framework product operations are based on.

As the product adds more functionality, the list of system tables can grow. The complete list of the system tables can be found in the System Table List section.

System Table Standards

System table standards must be observed for the following reasons:

- The product installation and upgrade process and customer modification data extract processes depend on the data prefix and owner flag values to determine the system data owned by each product.

- The standards ensure that there will be no data conflict in the product being developed and the future Oracle Utilities Application Framework release. They also ensure that there will be no data conflict between customer modifications and future Oracle Utilities product releases.

- The data prefix is used to prevent test data from being released to production.

  Developer's Note: All test data added to the system data tables must be prefixed by ZZ (all upper case) in order for the installation and upgrade utility to recognize them as test data.

Guidelines for System Table Updates

This section describes guidelines regarding the updating of the system table properties.

Business Configuration Tables

The majority of data in the tables in this group belongs to the customer. But these tables are shipped with some initial data in order for the customer to login to the system and begin configuring the product. Unless specified otherwise, the initial data is maintained by Oracle Utilities Application Framework and subject to subsequent upgrade.

Application Security and User Profile

These tables define the access rights of a User Group to Application Services and Application Users.

Properties	Description
Tables	SC_ACCESS_CNTL, SC USER, SC_USR_GRP_PROF, SC_USR_GRP_USR, SC_USER_GROUP, SC_USER_GROUP_L
Initial Data	User Group ALL_SERVICES and default system user SYSUSER. Upon installation the system default User Group ALL_SERVICES is given unrestricted accesses to all services defined in Oracle Utilities Application Framework.

Developer's Note: When a new service is added to the system, all actions defined for the service must be made available to the User Group ALL_ SERVICES.

Currency Code

The ISO 4217 three-letter codes are taken as the standard code for the representation of each currency.

Properties	Description
Tables	CI_CURRENCY_CD, CI_CURRENCY_CD_L
Initial Data	United States Dollar (USD)

Display Profile

The Display Profile Code is referenced in the User (SC_USER) table.

Properties	Description
Tables	CI_DISP_PROF, CI_DISP_PROF_L

Properties	Description
Initial Data	North America (NORTHAM) and Europe (EURO) and HIJRI Format (HIJRI)
	Configuration Note: In order to use HIJRI Format display profile, additional configuration is needed to define the mappings between Hijri dates and Gregorian dates.
	Refer to the Display Profile documentation for more information.

Configuration Note: In order to use HIJRI Format display profile, additional configuration is needed to define the mappings between Hijri dates and Gregorian dates.

Refer to the Display Profile documentation for more information.

Installation Options

Installation Option has only one row that is shipped with the initial installation of the Oracle Utilities Application Framework. The updatable columns in these tables are customer data and will not be overridden by the upgrade process unless a special script is written and included in the upgrade process.

Properties	Description
Tables	F1_INSTALLATION, CI_INSTALL_ALG, CI_INSTALL_MSG, CI_INSTALL_MSG_L, CI_INSTALL_PROD
Initial Data	Option 11111

Developer's Note: The system data owner of an environment is defined in the Installation Option. This Owner Flag value is stamped on all system data that is added to this environment. The installation default value is Customer Modification (CM). This value must be changed in the base product development environments.

Language Code

Language Code must be a valid code defined in ISO 639-2 Alpha-3. Adding a new language code to the table without translating all language dependent objects in the system can cause errors when a user chooses the language.

Properties	Description
Tables	CI_LANGUAGE
Initial Data	English (ENG)

Time Zone

The installation options require a valid time zone. A value for UTC (Coordinated Universl Time) is provided. Implementations should define the appropriate time zone and update the installation option value accordingly.

Properties	Description
Tables	CI_TIME_ZONE, CI_TIME_ZONE_L
Initial Data	UTC

To Do Priority and Role

New To Do Types released will be linked to the default To Do Role and set to the product assigned priority value initially. These initial settings can be overridden by the implementation.

Properties	Description
Tables	CI_ROLE(L), CI_TD_VAL_ROLE
Initial Data	F1_DFLT

Development and Implementation System Tables

This section defines the standards for the system tables that contain data for application development. The data in these tables implement business logic and UI functions shared by various products and product extensions in the same database.

Standards

When adding new data, the owner flag value of the environment must prefix certain fields of these tables. For example, when a developer adds a new algorithm type to an Oracle Utilities Work and Asset Management environment, C1 should prefix the new Algorithm Type code. The fields that are subject to this rule are listed in Standard Data Fields property.

The data that is already in these tables cannot be modified if the data owner is different than the environment owner. This prevents the developers from accidentally modifying system data that belongs to the Oracle Utilities Application Framework or the base products. However, some fields are exempt from this rule and can be modified by Customer Modification. These fields are listed in the Customer Modification Fields property.

Note that the system supports a system upgrade rule called Override Owner flag. If duplicate data rows (data row with same primary key values) are found at the time of upgrade, the owner flag values will get overridden. The lower level application system data will override the upper level system data. For example, F1 overrides C1, F1&C1 override CM, and so on. This rule will be applied to the following tables: CI_CHAR_ENTITY, CI_MD_MO_ALG, C1_PORTAL_OPT, F1_BUS_OBJ_ALG, F1_BUS_OBJ_STATUS_ALG, CI_MD_MO_OPT, F1_BUS_OBJ_OPT, F1_BUS_OBJ_STATUS_OPT, F1_BUS_OBJ_STATUS, F1_BUS_OBJ_STATUS_L

Algorithm Type

Properties	Description
Tables	CI_ALG_TYPE, CI_ALG_TYPE_L, CI_ALG_TYPE_PRM, CI_ALG_TYPE_PRM_L
Standard Data Fields	Algorithm Type (ALG_TYPE_CD)
Customer Modification	None

Algorithm

Properties	Description
Tables	CI_ALG, CI_ALG_L, CI_ALG_PARM, CI_ALG_VER
Standard Data Fields	Algorithm (ALG_CD)
Customer Modification	None

Application Security

Properties	Description
Tables	SC_APP_SERVICE, SC_APP_SERVICE_L, CI_APP_SVC_ACC
Standard Data Fields	Application Service ID (APP_SVC_ID).
Customer Modification	None

Batch Control

Properties	Description
Tables	CI_BATCH_CTRL, CI_BATCH_CTRL_L, CI_BATCH_CTRL_P, CI_BATCH_CTRL_P_L
Standard Data Fields	Batch Process (BATCH_CD), Program Name (PROGRAM_NAME)

Properties	Description
Customer Modification	Next Batch Number (NEXT_BATCH_NBR), Last Update Instance (LAST_UPDATE_INST), Last Update Date time (LAST_UPDATE_DTTM) and the batch process update these columns. Time Interval (TIMER_INTERVAL), Thread Count (BATCH_THREAD_CNT), Maximum Commit Records (MAX_COMMIT_RECS), User (USER_ID), Language (LANGUAGE_CD), Email Address (EMAILID), Start program debug tracing (TRC_PGM_STRT_SW), End Program Debug trace (TRC_PGM_END_SW), SQL debug tracing (TRC_SQL_SW) and Standard debug tracing (TRC_STD_SW) on CI_BATCH_CTRL Table. Batch Parameter Value (BATCH_PARM_VAL) and Security flag (TEXT_SECURITY_FLG) on Batch Control Parameters Table (CI_BATCH_CTRL_P)

Business Object

Properties	Description
Tables	F1_BUS_OBJ, F1_BUS_OBJ_L, F1_BUS_OBJ_ALG, F1_BUS_OBJ_OPT, F1_BUS_OBJ_STATUS, F1_BUS_OBJ_STATUS_L, F1_BUS_OBJ_STATUS_ALG, F1_BUS_OBJ_STATUS_OPT, F1_BUS_OBJ_STATUS_RSN, F1_BUS_OBJ_STATUS_RSN_L, F1_BUS_OBJ_STATUS_RSN_CHAR, F1_BUS_OBJ_TR_RULE, F1_BUS_OBJ_TR_RULE_L
Standard Data Fields	Business Object (BUS_OBJ_CD), Status Reason (BO_STATUS_REASON_CD)
Customer Modification	Batch Control (BATCH_CD), Alert (BO_ALERT_FLG), Sequence (SORT_SEQ5), Status Reason (STATUS_REASON_FLG) fields on Business Object Status Table (F1_BUS_OBJ_STATUS). Instance Control (INSTANCE_CTRL_FLG), Application Service (APP_SVC_ID) on Business Object Table (F1_BUS_OBJ). Status Reason Selection (STATUS_REASON_SELECT_FLG) on Status Reason Table (F1_BUS_OBJ_STATUS_RSN)

Business Service

Properties	Description
Tables	F1_BUS_SVC, F1_BUS_SVC_L
Standard Data Fields	Business Service (BUS_SVC_CD)
Customer Modification	Application Service (APP_SVC_ID)

Characteristics

Properties	Description
Tables	CI_CHAR_TYPE, CI_CHAR_TYPE_L, CI_CHAR_ENTITY, CI_CHAR_VAL, CI_CHAR_VAL_L
Standard Data Fields	Characteristic Type (CHAR_TYPE_CD) Characteristic Value (CHAR_VAL) on CI_CHAR_VAL

If the characteristic type is customizable, Customer Modification can insert new characteristic values. CM must prefix when implementers introduce a new characteristic value. |
| Customer Modification | Adhoc Characteristic Value Validation Rule (ADHOC_VAL_ALG_CD), Allow Search by Characteristic Value (SEARCH_FLG) |

Configuration Migration Assistant

Properties	Description
Tables	F1_MIGR_PLAN, F1_MIGR_PLAN_L, F1_MIGR_PLAN_INSTR, F1_MIGR_PLAN_INSTR_L, F1_MIGR_PLAN_INSTR_ALG, F1_MIGR_REQ, F1_MIGR_REQ_L, F1_MIGR_REQ_INSTR, F1_MIGR_REQ_INST R_L, F1_MIGR_REQ_INSTR_ENTITY, F1_MIGR_REQ_INCL_REQ
Standard Data Fields	Migration Plan Code (MIGR_PLAN_CD), Migration Request Code (MIGR_REQ_CD)
Customer Modification	None

Data Area

Properties	Description
Tables	F1_DATA_AREA, F1_DATA_AREA_L
Standard Data Fields	Data Area Code (DATA_AREA_CD)
Customer Modification	None

Deployment Part

Properties	Description
Tables	F1_DEPLOYMENT_PART, F1_DEPLOYMENT_PART_L, F1_DEPLOYMENT_ITEM
Standard Data Fields	Deployment ID (F1_DEPLOYMENT_ID)
Customer Modification	None

Display Icon

Properties	Description
Tables	CI_DISP_ICON, CI_DISP_ICON_L
Standard Data Fields	Display Icon Code (DISP_ICON_CD)
Customer Modification	None

Extendable Lookup

Properties	Description
Tables	F1_EXT_LOOKUP_VAL, F1_EXT_LOOKUP_VAL_L, F1_EXT_LOOKUP_VAL_CHAR
Standard Data Fields	Business Object (BUS_OBJ_CD), Extendable Lookup Value (F1_EXT_LOOKUP_VALUE)
Customer Modification	Business Object Data Area (BO_DATA_AREA) Override Description (DESCR_OVRD) on Extendable Lookup Field Value Language Table (F1_EXT_LOOKUP_VAL_L) **Note:** When the product releases base owned records in Extendable Lookup, if there are additional elements the business object will map the element to the BO_DATA_AREA if the value is allowed to be modified by an implementation.

Foreign Key Reference

Properties	Description
Tables	CI_FK_REF, CI_FK_REF_L
Standard Data Fields	FK reference code (FK_REF_CD)
Customer Modification	Info Program Name (INFO_PRG), Zone (ZONE_CD)

Inbound Web Service

Properties	Description
Tables	F1_IWS_SVC_L, F1_IWS_SVC, F1_IWS_SVC_OPER_L, F1_IWS_SVC_OPER, F1_IWS_SVC_OPER_PARM, F1_IWS_ANN_L, F1_IWS_ANN_PARM, F1_IWS_ANN, F1_IWS_ANN_TYPE_L, F1_IWS_ANN_TYPE, F1_IWS_ANN_TYPE_PARM, F1_IWS_ANN_TYPE_PARM_L
Standard Data Fields	Webservice Name (IN_SVC_NAME), Annotation (ANN_CD), Annotation Type (ANN_TYPE_CD)

Properties	Description
Customer Modification	Debug (DEBUG_SW), Active (ACTIVE_SW), Trace (TRACE_SW), Request XSL (REQUEST_XSL), Response XSL (RESPONSE_XSL)

Unsupported Metadata

Properties	Description
Tables	F1_LGCY_OBJ
Standard Data Fields	Object ID (LGCY_OBJ_ID)
Customer Modification	None

Lookup

Properties	Description
Tables	CI_LOOKUP_FIELD, CI_LOOKUP_VAL, CI_LOOKUP_VAL_L
Standard Data Fields	Field Name (FIELD_NAME) • A lookup field name must have corresponding field metadata. The name of the lookup field column must be assigned to avoid conflicts among different products. If you follow the standards for database field names, a Customer Modification lookup field name will be automatically Customer Modification prefixed. Field Value (FIELD_VALUE) • If a lookup field is customizable, Customer Modification can insert new lookup values. X or Y must prefix when implementers introduce a new lookup value. • Product development may add lookup values to a Oracle Utilities Application Framework owned lookup field's value. When extended new value is added, the Owner Flag is used to prefix the value. .
Customer Modification	Override Description (DESCR_OVRD) on Lookup Field Value Language Table (CI_LOOKUP_VAL_L)

Map

Properties	Description
Tables	F1_MAP, F1_MAP_L

Properties	Description
Standard Data Fields	UI Map (MAP_CD)
Customer Modification	None

Managed Content

Properties	Description
Tables	F1_MANAG_CONTENT, F1_MANAG_CONTENT_L
Standard Data Fields	Managed Content (MANAG_CONTENT_CD)
Customer Modification	None

Messages

Properties	Description
Tables	CI_MSG_CATEGORY, CI_MSG_CATEGORY_L, CI_MSG, CI_MSG_L

Properties	Description
Standard Data Fields	Message Category (MESSAGE_CAT_NBR) • Messages are grouped in categories and each category has message numbers between 1 and 99999. A range of message categories is assigned to a product. An implementation may only use categories assigned for customization use. • Implementer Message Categories are 80000 and 90000 • Reserved for Tests - 99999 Message Number (MESSAGE_NBR) for message categories • Message numbers below 1000 are reserved for common messages. Implementers must not use message numbers below 1000. Message Number (MESSAGE_NBR) for Java message categories • Subsystem Standard Messages - 00001 thru 02000 • Reserved - 02001 thru 09999 • Published Messages - 10001 thru 11000 • Package Messages - 10001 thru 90000 • Reserved - 90001 thru 99999 • Each package is allocated 100 message numbers, each starting from 101. • Published Messages are messages that are special-interest messages that implementations need to know about and are therefore published in the user docs. Examples of these include messages that are highly likely to be changed for an implementation, or messages that are embedded into other texts/messages and therefore the message number is never shown • Reserved message number ranges are for future use and therefore must not be used by all products.
Customer Modification	Override Description (DESCRLONG_OVRD), Message Text Override (MESSAGE_TEXT_OVRD)

Meta Data - Table and Field

Properties	Description
Tables	CI_MD_TBL, CI_MD_TBL_FLD, CI_MD_TBL_L, CI_MD_TBL_FLD_L, CI_MD_FLD, CI_MD_FLD_L, F1_DB_OBJECTS_REPO
Standard Data Fields	Table Name (TBL_NAME) • Table names must match with the physical table name or view name in the database. • Field Name (FLD_NAME) Field name must match with the physical column name in the database unless the field is a work field. Field name does not have to follow the prefixing standard unless the field is a work field or customer modification field. • F1_DB_OBJECTS_REPO Table stores information about Indexes, Sequences, Triggers and other database objects excluding Tables and Fields (as they are already stored in the other Metadata tables)
Customer Modification	AuditSwitches(AUDIT_INSERT_SW,AUDIT_UPDATE_SW, AUDIT_DELETE_SW), Override label (OVRD_LABEL) on MD Table Field Table (CI_MD_TBL_FLD). Audit Program Name (AUDIT_PGM_NAME), Audit Table Name (AUDIT_TBL_NAME), Audit Program Type (AUDIT_PGM_TYPE_FLG), Key Validation (KEY_VALIDATION_FLG) and Caching strategy (CACHE_FLG) on MD Table (CI_MD_TBL). Override Label (OVRD_LABEL) and Customer Specific Description (DESCRLONG_OVRD) on Field Table.

Meta Data - Constraints

Properties	Description
Tables	CI_MD_CONST, CI_MD_CONST_FLD
Standard Data Fields	Constraint Id (CONST_ID) • Index Name for Primary Constraints • <Index Name>Rnn for Foreign Key Constraints Where • nn: integer, 01 through 99
Customer Modification	None

Meta Data - Menu

Menus can be extended to support multiple products by adding a new menu line to an existing menu. The sequence number on the menu line language table

(CI_MD_MENU_LINE_L) determines the order the menu lines appear. Within the same sequence, alphabetic sorting is used.

Properties	Description
Tables	CI_MD_MENU, CI_MD_MENU_L, CI_MD_MENU_ITEM, CI_MD_MENU_ITEM_L, CI_MD_MENU_LINE, CI_MD_MENU_LINE_L
Standard Data Fields	Menu Name (MENU_NAME), Menu Item Id (MENU_ITEM_ID), Menu Line Id (MENU_LINE_ID)
Customer Modification	Override Label (OVRD_LABEL) on Menu Line Language Table (CI_MD_MENU_LINE_L)

Meta Data - Program, Location and Services

Properties	Description
Tables	CI_MD_PRG_COM, CI_MD_PRG_LOC, CI_MD_SVC, CI_MD_SVC_L, CI_MD_SVC_PRG, CI_MD_PRG_MOD, CI_MD_PRG_EL_AT, CI_MD_PRG_ELEM, CI_MD_PRG_SEC, CI_MD_PRG_SQL, CI_MD_PRG_VAR, CI_MD_PRG_TAB
Standard Data Fields	Program Component Id (PROG_COM_ID), Location Id (LOC_ID), Program Component Name (PROG_COM_NAME), Service Name (SVC_NAME), Navigation Key (NAVIGATION_KEY)
Customer Modification	User Exit Program Name (USER_EXIT_PGM_NAME) on Program Components Table (CI_MD_PRG_COM),

Meta Data - Maintenance Object

Properties	Description
Tables	CI_MD_MO, CI_MD_MO_L, CI_MD_MO_TBL, CI_MD_MO_OPT, CI_MD_MO_ALG
Standard Data Fields	Maintenance Object (MAINT_OBJ_CD)
Customer Modification	None

Meta Data - Work Tables

Properties	Description
Tables	CI_MD_WRK_TBL, CI_MD_WRK_TBL_L, CI_MD_WRK_TBLFLD, CI_MD_MO_WRK
Standard Data Fields	Work Table Name (WRK_TBL_NAME)
Customer Modification	None

Meta Data - Search Object

Properties	Description
Tables	CI_MD_SO, CI_MD_SO_L, CI_MD_SO_RSFLD, CI_MD_SO_RSFLDAT, CI_MD_SOCG, CI_MD_SOCG_FLD, CI_MD_SOCG_FLDAT, CI_MD_SOCG_L, CI_MD_SOCG_SORT
Standard Data Fields	Search Object (SO_CD)
Customer Modification	None

Mobile Component

Properties	Description
Tables	F1_MOBILE_COMPONENT, F1_MOBILE_COMPONENT_L, F1_MOB_COMP_CNT, F1_MOBILE_COMP_CHAR
Standard Data Fields	Mobile Component Code (F1_MOB_COMP_TYPE_CD)
Customer Modification	Expiration Days (F1_EXPIRATION_TIME_DUR)

Navigation Option

Properties	Description
Tables	CI_NAV_OPT, CI_NAV_OPT_L, CI_NAV_OPT_CTXT, CI_NAV_OPT_USG, CI_MD_NAV
Standard Data Fields	Navigation Option Code (NAV_OPT_CD), Navigation Key (NAVIGATION_KEY)
Customer Modification	None

Outbound Message Type

Properties	Description
Tables	F1_OUTMSG_TYPE, F1_OUTMSG_TYPE_L
Standard Data Fields	Outbound Message Type Code (OUTMSG_TYPE_CD)
Customer Modification	Priority (OUTMSG_PRIOR_FLG)

Portal and Zone

Properties	Description
Tables	CI_PORTAL, CI_PORTAL_L, CI_PORTAL_ZONE, CI_PORTAL_OPT, CI_ZONE, CI_ZONE_L, CI_ZONE_PRM, CI_ZONE_HDL, CI_ZONE_HDL_L, CI_ZONE_HDL_PRM, CI_ZONE_HDL_PRM_L, CI_UI_ZONE
Standard Data Fields	Portal Code (PORTAL_CD), Zone Code (ZONE_CD), Zone Type Code (ZONE_HDL_CD) • A new Zone can be added to the Product owned Portal Pages. • The existing Zones cannot be removed from the Product owned Portal Pages.
Customer Modification	Sort Sequence (SORT_SEQ) on Context Sensitive Zone Table (CI_UI_ZONE). Show on Portal Preferences (USER_CONFIG_FLG) on Portal Table (CI_PORTAL). Override Sequence (SORT_SEQ_OVRD) on Portal Zone Table (CI_PORTAL_ZONE). Customer Specific Description (DESCRLONG_OVRD) on Zone Language Table (CI_ZONE_L). Override Parameter Value (ZONE_HDL_PARM_OVRD) on Zone Type Parameters Table (CI_ZONE_HDL_PRM). Override Parameter Value (ZONE_PARM_VAL_OVRD) on Zone Parameters Table (CI_ZONE_PRM).

Process Flow Type

Properties	Description
Tables	F1_PROC_DEFN F1_PROC_DEFN_L F1_PROC_NEXT_PANEL F1_PROC_PANEL
Standard Data Fields	Process Flow Type (PROCESS_CD)
Customer Modification	None

Sequence

Properties	Description
Tables	CI_SEQ
Standard Data Fields	Sequence Name (SEQ_NAME)
Customer Modification	Sequence Number (SEQ_NBR) This field is updated by the application process and must be set to 1 initially.

Schema

Properties	Description
Tables	F1_SCHEMA
Standard Data Fields	Schema Name (SCHEMA_NAME)
Customer Modification	None

Script

Properties	Description
Tables	CI_SCR, CI_SCR_L, CI_SCR_CRT, CI_SCR_CRT_GRP, CI_SCR_CRT_GRP_L, CI_SCR_DA, CI_SCR_FLD_MAP, CI_SCR_PRMPT, CI_SCR_PRMPT_L, CI_SCR_STEP, CI_SCR_STEP_L
Standard Data Fields	Script (SCR_CD)
Customer Modification	None

To Do Type

Properties	Description
Tables	CI_TD_TYPE, CI_TD_TYPE_L, CI_TD_SRTKEY_TY, CI_TD_DRLKEY_TY, CI_TD_SRTKEY_TY_L
Standard Data Fields	To Do Type Code (TD_TYPE_CD)
Customer Modification	Creation Batch Code (CRE_BATCH_CD), Route Batch Code (RTE_BATCH_CD), Priority Flag (TD_PRIORITY_FLG) on To Do Type Table (CI_TD_TYPE)

Web Service Category

Properties	Description
Tables	F1_WEB_CAT, F1_WEB_CAT_L, F1_WEB_CAT_INCL_SVC
Standard Data Fields	Web Service Category code (WEB_SVC_CAT_CD)
Customer Modification	None

XAI Configuration

Properties	Description
Tables	CI_XAI_ADAPTER, CI_XAI_ADAPTER_L, CI_XAI_CLASS, CI_XAI_CLASS_L, CI_XAI_ENV_HNDL, CI_XAI_ENV_HNDL_L, CI_XAI_FORMAT, CI_XAI_FORMAT_L, CI_XAI_RCVR, CI_XAI_RCVR_L, CI_XAI_RCVR_CTX, CI_XAI_RCVR_RSP, CI_XAI_RCVR_RGRP, CI_XAI_SENDER, CI_XAI_SERNDER_L, CI_XAI_SNDR_CTX, CI_XAI_OPTION
Standard Data Fields	Adapter Id (XAI_ADAPTER_ID), Class Id (XAI_CLASS_ID), Envelope Handler Id (XAI_ENV_HNDL_ID), XAI Format Id (XAI_FORMAT_ID), Receiver Id (XAI_RCVR_ID), Sender Id (XAI_SENDER_ID)
Customer Modification	Option Value (OPTION_VALUE) on Message Option Table (CI_XAI_OPTION)

XAI Services

Properties	Description
Tables	CI_XAI_IN_SVC, CI_XAI_IN_SVC_L, CI_XAI_SVC_PARM
Standard Data Fields	XAI Inbound Service Id (XAI_IN_SVC_ID), XAI Inbound Service Name (XAI_IN_SVC_NAME)
Customer Modification	XAI Version (XAI_VERSION_ID), Trace (TRACE_SW), Debug (DEBUG_SW), Request XSL (INPUT_XSL), Response XSL (RESPONSE_XSL), Record XSL (RECORD_XSL and Post Error (POST_ERROR_SW) on XAI Inbound Service Table (CI_XAI_IN_SVC)

System Table List

This section contains names of system tables, upgrade actions, and a brief description of tables. The upgrade actions are explained below.

Keep (KP): The data in the table in the customer's database is kept untouched. No insert or delete is performed to this table by the upgrade process. The initial installation will add necessary data for the system

Merge (MG): The non-base product data in the table in the database is kept untouched. If the data belongs to the base product, any changes pertaining to the new version of the software are performed.

Refresh (RF): The existing data in the table is replaced with the data from the base product table. The product does not support customer specific data in these tables.

> **Note.** New product data is also inserted into tables marked as 'Merge'. If implementers add rows for a customer specific enhancement, it can cause duplication when the system data gets upgraded to the next version. We strongly recommend following the guidelines on how to use designated range of values or prefixes to segregate the implementation data from the base product data.

Table Name	Upgrade Action	Description
CI_ALG	MG	Algorithm
CI_ALG_L	MG	Algorithm Language
CI_ALG_PARM	MG	Algorithm Parameters
CI_ALG_TYPE	MG	Algorithm Type
CI_ALG_TYPE_L	MG	Algorithm Type Language
CI_ALG_TYPE_PRM	MG	Algorithm Type Parameter
CI_ALG_TYPE_PRM_L	MG	Algorithm Type Parameter Language
CI_ALG_VER	MG	Algorithm Version
CI_APP_SVC_ACC	MG	Application Service Access Mode
CI_BATCH_CTRL	MG	Batch Control
CI_BATCH_CTRL_ALG	MG	Batch Control Algorithm
CI_BATCH_CTRL_L	MG	Batch Control Language
CI_BATCH_CTRL_P	MG	Batch Control Parameters
CI_BATCH_CTRL_P_L	MG	Batch Control Parameters Language
CI_CHAR_ENTITY	MG	Characteristic Type Entity
CI_CHAR_TYPE	MG	Characteristic Type
CI_CHAR_TYPE_L	MG	Characteristic Type Language
CI_CHAR_VAL	MG	Characteristic Type Value
CI_CHAR_VAL_L	MG	Characteristic Type Value Language

Table Name	Upgrade Action	Description
CI_DISP_ICON	MG	Display Icon
CI_DISP_ICON_L	MG	Display Icon Language
CI_FK_REF	MG	Foreign Key Reference
CI_FK_REF_L	MG	Foreign Key Reference Language
CI_LANGUAGE	MG	Language Code
CI_LOOKUP_FIELD	MG	Lookup Field
CI_LOOKUP_VAL	MG	Lookup Field Value
CI_LOOKUP_VAL_L	MG	Lookup Field Value Language
CI_MD_CONST	MG	Constraints
CI_MD_CONST_FLD	MG	Constraint Fields
CI_MD_FLD	MG	Field
CI_MD_FLD_L	MG	Field Language
CI_MD_MENU	MG	Menu Information
CI_MD_MENU_IMOD	MG	Menu Item Module Maint
CI_MD_MENU_ITEM	MG	Menu Item
CI_MD_MENU_ITEM_L	MG	Menu Item Language
CI_MD_MENU_L	MG	Menu Language
CI_MD_MENU_LINE	MG	Menu Line
CI_MD_MENU_LINE_L	MG	Menu Line Language
CI_MD_MENU_MOD	MG	Menu Product Components
CI_MD_MO	MG	Maintenance Object
CI_MD_MO_ALG	MG	Maintenance Object Algorithm
CI_MD_MO_L	MG	Maintenance Object Language
CI_MD_MO_OPT	MG	Maintenance Object Option
CI_MD_MO_TBL	MG	Maintenance Object Table
CI_MD_MO_WRK	MG	Maintenance Object Work Tables
CI_MD_NAV	MG	Navigation Key
CI_MD_PRG_COM	MG	Program Components
CI_MD_PRG_ELEM	MG	UI Page Elements
CI_MD_PRG_EL_AT	MG	UI Page Element Attributes
CI_MD_PRG_LOC	MG	Program Location
CI_MD_PRG_MOD	MG	Program Module

Table Name	Upgrade Action	Description
CI_MD_PRG_SEC	MG	UI Page Sections
CI_MD_PRG_SQL	MG	MD SQL Meta Data
CI_MD_PRG_TAB	MG	UI Tab Meta Data
CI_MD_PRG_VAR	MG	Program Variable
CI_MD_SO	MG	Search Object
CI_MD_SOCG	MG	Search Object Criteria Group
CI_MD_SOCG_FLD	MG	Search Object Criteria Group Field
CI_MD_SOCG_FLDAT	MG	Search Criteria Group Field Attribute
CI_MD_SOCG_L	MG	Search Object Criteria Group Language
CI_MD_SOCG_SORT	MG	Search Criteria Group Result Sort Order
CI_MD_SO_L	MG	Search Object Language
CI_MD_SO_RSFLD	MG	Search Object Result Field
CI_MD_SO_RSFLDAT	MG	Search Object Result Field Attribute
CI_MD_SVC	MG	MD Service
CI_MD_SVC_L	MG	MD Service Language
CI_MD_SVC_PRG	MG	MD Service Program
CI_MD_TAB_MOD	MG	UI Tab Module
CI_MD_TBL	MG	MD Table
CI_MD_TBL_FLD	MG	MD Table Field
CI_MD_TBL_FLD_L	MG	MD Table Field Language
CI_MD_TBL_L	MG	MD Table Language
CI_MD_WRK_TBL	MG	Work Table
CI_MD_WRK_TBLFLD	MG	Work Table Field
CI_MD_WRK_TBL_L	MG	Work Table Language
CI_MSG	MG	Message
CI_MSG_CATEGORY	MG	Message Category
CI_MSG_CATEGORY_L	MG	Message Category Language
CI_MSG_L	MG	Message Language
CI_NAV_OPT	MG	Navigation Option
CI_NAV_OPT_CTXT	MG	Navigation Option Context
CI_NAV_OPT_L	MG	Navigation Option Language
CI_NAV_OPT_USG	MG	Navigation Option Usage

Table Name	Upgrade Action	Description
CI_PORTAL	MG	Portal
CI_PORTAL_L	MG	Portal Language
C1_PORTAL_OPT	MG	Portal Option
CI_PORTAL_ZONE	MG	Portal Zone
CI_SCR	MG	Script
CI_SCR_CRT	MG	Script Criteria
CI_SCR_CRT_GRP	MG	Script Criteria Group
CI_SCR_CRT_GRP_L	MG	Script Criteria Group Language
CI_SCR_DA	MG	Script Data Area
CI_SCR_FLD_MAP	MG	Script Field Mapping
CI_SCR_L	MG	Script Language
CI_SCR_PRMPT	MG	Script Prompt
CI_SCR_PRMPT_L	MG	Script Prompt Language
CI_SCR_STEP	MG	Script Step
CI_SCR_STEP_L	MG	Script Step Language
CI_SEQ	MG	Sequence
CI_TD_DRLKEY_TY	MG	To Do Type Drill Key
CI_TD_SRTKEY_TY	MG	To Do Type Sort Key
CI_TD_SRTKEY_TY_L	MG	To Do Type Sort Key Language
CI_TD_TYPE	MG	To Do Type
CI_TD_TYPE_L	MG	To Do Type Language
CI_UI_ZONE	MG	Context Sensitive Zone
CI_USR_NAV_LINK	MG	User Favorite Links
CI_XAI_ADAPTER	MG	XAI Adapter
CI_XAI_ADAPTER_L	MG	XAI Adapter Lang
CI_XAI_CLASS	MG	Message Class
CI_XAI_CLASS_L	MG	Message Class Language
CI_XAI_ENV_HNDL	MG	XAI Envelope Handler
CI_XAI_ENV_HNDL_L	MG	XAI Envelope Handler Language
CI_XAI_IN_SVC	MG	XAI Inbound Service
CI_XAI_IN_SVC_L	MG	XAI Inbound Service Language
CI_XAI_SVC_PARM	MG	XAI Inbound Service Parameters

Table Name	Upgrade Action	Description
CI_ZONE	MG	Zone
CI_ZONE_HDL	MG	Zone Type
CI_ZONE_HDL_L	MG	Zone Type Language
CI_ZONE_HDL_PRM	MG	Zone Type Parameters
CI_ZONE_HDL_PRM_L	MG	Zone Type Parameters Language
CI_ZONE_L	MG	Zone Language
CI_ZONE_PRM	MG	Zone Parameters
F1_BUS_OBJ	MG	Business Object
F1_BUS_OBJ_ALG	MG	Business Object Algorithm
F1_BUS_OBJ_L	MG	Business Object Language
F1_BUS_OBJ_OPT	MG	Business Object Option
F1_BUS_OBJ_STATUS	MG	Business Object Status
F1_BUS_OBJ_STATUS_ALG	MG	Business Object Status Algorithm
F1_BUS_OBJ_STATUS_L	MG	Business Object Status Language
F1_BUS_OBJ_STATUS_OPT	MG	Business Object Status Option
F1_BUS_OBJ_STATUS_RSN	MG	Status Reason
F1_BUS_OBJ_STATUS_RSN_L	MG	Status Reason Language
F1_BUS_OBJ_TR_RULE	MG	Business Object Transition Rule
F1_BUS_OBJ_TR_RULE_L	MG	Business Object Transition Rule Language
F1_BUS_SVC	MG	Business Service
F1_BUS_SVC_L	MG	Business Service Language
F1_DATA_AREA	MG	Data Area
F1_DATA_AREA_L	MG	Data Area Language
F1_DB_OBJECTS_REPO	MG	Database Objects Repository
F1_DEPLOYMENT_ITEM	MG	Deployment Part Item
F1_DEPLOYMENT_PART	MG	Deployment Part
F1_DEPLOYMENT_PART_L	MG	Deployment Part Language
F1_EXT_LOOKUP_VAL	MG	Extendable Lookup
F1_EXT_LOOKUP_VAL_L	MG	Extendable Lookup Language
F1_EXT_LOOKUP_VAL_CHAR	MG	Extendable Lookup Characteristics
F1_IWS_ANN	MG	Web Service Annotation

Table Name	Upgrade Action	Description
F1_IWS_ANN_L	MG	Web Service Annotation Language
F1_IWS_ANN_PARM	MG	Web Service Annotation Parameter
F1_IWS_ANN_TYPE	MG	Web Service Annotation Type
F1_IWS_ANN_TYPE_L	MG	Web Service Annotation Type Language
F1_IWS_ANN_TYPE_PARM	MG	Web Service Annotation Type Parm
F1_IWS_ANN_TYPE_PARM_L	MG	Web Service Annotation Type Parameter Language
F1_IWS_SVC	MG	Inbound Web Service
F1_IWS_SVC_L	MG	Inbound Web Service Language
F1_IWS_SVC_OPER	MG	Inbound Web Service Operations
F1_IWS_SVC_OPER_PARM	MG	Inbound Web Service Operations Parameter
F1_MANAG_CONTENT	MG	Managed Content
F1_MANAG_CONTENT_L	MG	Managed Content Language
F1_MAP	MG	UI Map
F1_MAP_L	MG	UI Map Language
F1_MIGR_PLAN	MG	Migration Plan
F1_MIGR_PLAN_INSTR	MG	Migration Plan Instruction
F1_MIGR_PLAN_INSTR_ALG	MG	Migration Plan Instruction Algorithm
F1_MIGR_PLAN_INSTR_L	MG	Migration Plan Instruction Language
F1_MIGR_PLAN_L	MG	Migration Plan Language
F1_MIGR_REQ	MG	Migration Request
F1_MIGR_REQ_INCL_REQ	MG	Migration Request Grouping
F1_MIGR_REQ_INSTR	MG	Migration Request Instruction
F1_MIGR_REQ_INSTR_ ENTITY	MG	Migration Request Instruction Entity
F1_MIGR_REQ_INSTR_L	MG	Migration Request Instruction Language
F1_MIGR_REQ_L	MG	Migration Request Language
F1_MOBILE_COMPONENT	MG	Mobile Component
F1_MOBILE_COMPONENT_L	MG	Mobile Component Language
F1_MOB_COMP_CHAR	MG	Mobile Component Characteristics
F1_MOB_COMP_CNT	MG	Mobile Component Content

Table Name	Upgrade Action	Description
F1_OUTMSG_TYPE	MG	Outbound Message Type
F1_OUTMSG_TYPE_L	MG	Outbound Message Type Language
F1_PROC_DEFN	MG	Process Flow Type
F1_PROC_DEFN_L	MG	Process Flow Type Language
F1_PROC_NEXT_PANEL	MG	Next Panel
F1_PROC_PANEL	MG	Process Panels
F1_SCHEMA	MG	Schema
F1_WEB_CAT	MG	Web Service Category
F1_WEB_CAT_L	MG	Web Service Category Language
F1_WEB_CAT_INCL_SVC	MG	Web Service Category Included Services
SC_ACCESS_CNTL	MG	User Group Access Control
SC_APP_SERVICE	MG	Application Service
SC_APP_SERVICE_L	MG	Application Service Language
SC_USR_GRP_PROF	MG	User Group Profile
CI_CURRENCY_CD	KP	Currency Code
CI_CURRENCY_CD_L	KP	Currency Code Language
CI_DISP_PROF	KP	Display Profile
CI_DISP_PROF_L	KP	Display Profile Language
CI_TIME_ZONE	KP	Time Zone
CI_TIME_ZONE_L	KP	Time Zone Language
CI_USR_PORTAL	KP	User Portal
CI_XAI_JNDI_SVR	KP	XAI JNDI Server
CI_XAI_JNDI_SVR_L	KP	XAI JNDI Server Language
CI_XAI_OPTION	KP	Message Option
CI_XAI_SENDER	KP	Message Sender
CI_XAI_SENDER_L	KP	Message Sender Language
CI_XAI_SNDR_CTX	KP	Message Sender Context
F1_BUS_OBJ_STATUS_RSN_ CHAR	KP	Status Reason Characteristic
F1_INSTALLATION	KP	Installation Option - Framework
SC_USER	KP	User
SC_USER_CHAR	KP	User Characteristic

Table Name	Upgrade Action	Description
SC_USER_GROUP	KP	User Group
SC_USER_GROUP_L	KP	User Group Language
SC_USR_GRP_USR	KP	User Group User
CI_MD_ATT_TY	RF	MD Element Attribute Type
CI_MD_AT_DTL	RF	MD Element Attribute Type Detail
CI_MD_AT_DTL_L	RF	MD Element Attribute Type Detail Language
CI_MD_CTL	RF	Generator Control
CI_MD_CTL_L	RF	Generator Control Language
CI_MD_CTL_TMPL	RF	Generator Control Template
CI_MD_ELTY	RF	MD Element Type
CI_MD_ELTY_AT	RF	Element Type Attributes
CI_MD_ELTY_L	RF	Element Type Language
CI_MD_LOOKUP_F	RF	MD Lookup Field
CI_MD_MSG	RF	MD Message
CI_MD_MSG_L	RF	MD Message Language
CI_MD_PDF	RF	Predefined Fields
CI_MD_PDF_VAL	RF	Predefined Values
CI_MD_SRC_TYPE	RF	Source Type
CI_MD_SRC_TYPE_L	RF	Source Type Language
CI_MD_TMPL	RF	Template
CI_MD_TMPL_ELTY	RF	Template Element Types
CI_MD_TMPL_L	RF	Template Language
CI_MD_TMPL_VAR	RF	Template Variable
CI_MD_TMPL_VAR_L	RF	Template Variable Language
CI_MD_VAR	RF	Variable
CI_MD_VAR_DTL	RF	Variable Detail
CI_MD_VAR_DTL_L	RF	Variable Detail Language
CI_XAI_EXECUTER	RF	XAI Executer
CI_XAI_EXECUTER_L	RF	XAI Executer Language
CI_XAI_FORMAT	RF	XAI Format
CI_XAI_FORMAT_L	RF	XAI Format Language

Table Name	Upgrade Action	Description
F1_LGCY_OBJ	RF	Unsupported Metadata