Oracle Utilities Work and Asset Management

Database Administrator's Guide

Release 25.4

G26733-01

April 2025

Also used with:

Oracle Utilities Operational Device Management v25.4

ORACLE

Oracle Utilities Work and Asset Management Database Administrator's Guide

Contents

Preface

Welcome to the Oracle Utilities Work and Asset Management Database Administrator's Guide. This document provides instructions for installing and maintaining the Oracle Utilities Work and Asset Management V25.4 database and is intended for database administrators who will be installing and maintaining the database.

The preface includes the following:

Audience

This guide is intended for database administrators who install and maintain the database for Oracle Utilities Work and Asset Management.

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 User Guides
- Oracle Utilities Operational Device Management User Guides

Supplemental Documents

- Server Administration Guide
- Security Guide

Updates to this Documentation

The complete Oracle Utilities Work and Asset Management documentation set is available from Oracle Help Center at https://docs.oracle.com/en/industries/energy-water/index.html.

Visit My Oracle Support for additional and updated information about the product.

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.

Convention	Meaning
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.

Acronyms

The following terms are used in this document:

Term	Definition
OUODM/ODM	Oracle Utilities Operational Device Management
OUWAM/WAM	Oracle Utilities Work and Asset Management
OUAF	Oracle Utilities Application Framework

Additional Resources

For more information and support, visit the Oracle Support website.

Chapter 1

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
Oracle Linux 8.x/9.x (64-bit)	Oracle 19c (64-bit) on-premises and cloud (ADB)

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 very small, pilot projects or development 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 issue patches and service packs for the operating systems, application servers and database servers on top of specific versions that Oracle Utilities Work and Asset Management has been tested with.

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 Oracle Utilities Work and Asset Management production environment.

Always contact Oracle Utilities Work and Asset Management 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

Installing the Database

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

- Privileges to Modify OUAF V25.4 Database
- Installation Overview
- Installing the Oracle Database

Privileges to Modify OUAF V25.4 Database

Note the following:

- Oracle Utilities Application Framework (OUAF) 25.4 database installation requires that schema owner (CISADM) has the execute privilege on dbms_crypto package.

  Refer to the Enabling the DBMS_CRYPTO Package section for details.

- Recommendation to interval partition tables, F1_MO_UPD, F1_MO_UPD_BACKLOG for performance reasons.

  Refer to the Table Partitioning Recommendations section for more details.

Installation Overview

See the Supported Database Platforms section 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 V25.4 from V2.4.0.1.1.
- **Demo Install**: A database populated with demo data

The database installation requires a supported version of the Java Development Kit V17 and Oracle 19c 32-bit Client installed on the Windows 64-bit or 32-bit desktop where the install package is staged and run.

Prerequisites

The database installation requires:

- Java Development Kit V17

  This must be installed on the Linux, UNIX, or Windows server 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 Linux, Unix, or Windows database server on which you operate the production instance of Oracle Utilities Work and Asset Management.

Note: It is strongly recommended to use DBCA to create the database.

1. Create the database.
   a. For traditional on-premises database:
      a. Create the database using the Database Configuration Assistant (DBCA).

b. Refer to the article *Master Note: Overview of Database Configuration Assistant (DBCA) (Doc ID 1488770.1)* on My Oracle Support for more information. Make sure to set character set for database as follows:

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

b. For cloud database (ADB):

The pre-requisites to create an autonomous database are as follows:

a.Access to Oracle Cloud Infrastructure (OCI) is a must.

a.Login using your Cloud Account Name/Tenancy and Identity Provider if Single Sign-on (SSO) or Oracle Cloud Infrastructure Direct Sign-In.

b.Create the database following your organization's standards.

Note: Take note of the administrator credentials. It will be needed upon connecting to the new ADB.

c.Set up the connection to ADB.

a.Download Instance Wallet from OCI.

b.Unzip the downloaded wallet and set the wallet location as TNS ADMIN.

```
export TNS_ADMIN=<wallet location>
```

c.Modify the files from ADB wallet.

Edit sqlnet.ora and update DIRECTORY parameter.

WALLET_LOCATION = (SOURCE = (METHOD = file) (METHOD_DATA = (DIRECTORY="<wallet location>")))

Edit ojdbc.properties. Update javax.net.ssl.trustStorePassword and javax.net.ssl.keyStorePassword with your wallet password. Comment this line: oracle.net.wallet_location=(SOURCE=(METHOD=FILE)(METHOD_DATA=(DIRECTORY=${TNS_ADMIN})))

File should look like this:

Connection property while using Oracle wallets.

#oracle.net.wallet_location=(SOURCE=(METHOD=FILE)(METHOD_DATA=(DIRECTORY=${TNS_ADMIN})))

FOLLOW THESE STEPS FOR USING JKS

(1) Uncomment the following properties to use JKS.

(2) Comment out the oracle.net.wallet_location property above

(3) Set the correct password for both trustStorePassword and keyStorePassword.

It's the password you specified when downloading the wallet from OCI Console or the Service Console.

javax.net.ssl.trustStore=<wallet location>/truststore.jks

javax.net.ssl.trustStorePassword=<wallet password>

javax.net.ssl.keyStore=<wallet location>/keystore.jks

javax.net.ssl.keyStorePassword=<wallet password>

d. Verify DB connection using TNSNAMES from tnsnames.ora.

```
sqlplus USERNAME/password@<tnsname>
```

2. Enable the mandatory software options.
   - Oracle Spatial OR Oracle Locator
   - Oracle Text

   Note: This step is applicable for on-premises databases **only**.
3. Run the following SQL to make sure it is successful.

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

   Note: This step is applicable for on-premises databases **only**.

   Note that starting with V4.4.0.0.0, Oracle Utilities Application Framework database users are assigned only the minimum set of privileges needed to be able to perform their required job functions. This privilege assignment is based on 'Principle of Least Privilege' and is implemented with a view to make the application more secure.
4. Enable the Extended Data Types by setting DB parameter, max_string_size = EXTENDED.

   Follow the instructions provided in Oracle database documentation.

   Note: This step is applicable for on-premises databases **only**.
5. Create default tablespace CISTS_01 and required users and roles.

   Note: For ADB, make sure the connection is working. Refer to step 1.b.a.c in this section.

   Create the required roles:

```
CREATE ROLE CIS_ADM NOT IDENTIFIED;
CREATE ROLE CIS_USER NOT IDENTIFIED;
CREATE ROLE CIS_READ NOT IDENTIFIED;
```

   Grant privileges to roles:

```
GRANT CREATE TABLE TO CIS_ADM;
GRANT CREATE VIEW TO CIS_ADM;
GRANT CREATE SYNONYM TO CIS_ADM;
GRANT CREATE MATERIALIZED VIEW TO CIS_ADM;
GRANT CREATE SEQUENCE TO CIS_ADM;
GRANT CREATE INDEXTYPE TO CIS_ADM;
GRANT CREATE ROLE TO CIS_ADM;
GRANT CREATE TRIGGER TO CIS_ADM;
GRANT CREATE PROCEDURE TO CIS_ADM;
GRANT CREATE SYNONYM TO CIS_USER;
GRANT CREATE SYNONYM TO CIS_READ;
```

Note: Follow the password requirement(s) for ADB user creation and update with the Oracle provided tablespace DATA.

Create schema owner (CISADM):

- On-premises:

```
CREATE USER CISADM IDENTIFIED BY CISADM DEFAULT TABLESPACE
CISTS_01 TEMPORARY TABLESPACE TEMP PROFILE DEFAULT;
ALTER USER CISADM QUOTA UNLIMITED ON CISTS_01;
```

- ADB:

```
CREATE USER CISADM IDENTIFIED BY CISADM DEFAULT TABLESPACE DATA
TEMPORARY TABLESPACE TEMP PROFILE DEFAULT;
ALTER USER CISADM QUOTA UNLIMITED ON DATA;
```

Grant privileges and roles to schema owner (CISADM):

```
GRANT CREATE SESSION TO CISADM;
GRANT CREATE TABLESPACE TO CISADM;
GRANT READ on DBA_TABLESPACES TO CISADM;
GRANT ALTER TABLESPACE TO CISADM;
GRANT UNLIMITED TABLESPACE TO CISADM;
GRANT DROP TABLESPACE TO CISADM;
GRANT EXECUTE on DBMS_UTILITY TO CISADM;
GRANT SELECT_CATALOG_ROLE TO CISADM;
GRANT CIS_ADM TO CISADM;
GRANT EXECUTE_CATALOG_ROLE TO CISADM;
```

Enable DBMS_CRYPTO package:

Execute the following command as SYS for on-premises and as ADMIN for ADB.

```
Grant execute on dbms_crypto to CISADM;
```

Enable DBMS_RLS package:

Execute the following command as SYS for on-premises and as ADMIN for ADB.

```
grant execute on dbms_rls to CISADM;
```

Create read write user (CISUSER):

- On-premises:

```
CREATE USER CISUSER PROFILE DEFAULT IDENTIFIED BY CISUSER
DEFAULT TABLESPACE CISTS_01 TEMPORARY TABLESPACE TEMP;
ALTER USER CISUSER QUOTA UNLIMITED ON CISTS_01;
```

- ADB:

```
CREATE USER CISUSER PROFILE DEFAULT IDENTIFIED BY CISUSER
DEFAULT TABLESPACE DATA TEMPORARY TABLESPACE TEMP;
ALTER USER CISUSER QUOTA UNLIMITED ON DATA;
```

Grant privileges and roles to read write user (CISUSER):

```
GRANT CIS_USER to CISUSER;
GRANT CREATE SESSION TO CISUSER;
```

Create read only user (CISREAD):

- On-premises:

```
CREATE USER CISREAD IDENTIFIED BY CISREAD DEFAULT TABLESPACE
CISTS_01 TEMPORARY TABLESPACE TEMP;
ALTER USER CISREAD QUOTA UNLIMITED ON CISTS_01;
```

- ADB:

```
CREATE USER CISREAD IDENTIFIED BY CISREAD DEFAULT TABLESPACE
DATA TEMPORARY TABLESPACE TEMP;
ALTER USER CISREAD QUOTA UNLIMITED ON DATA;
```

Grant privileges and roles to read only user (CISREAD):

```
GRANT CIS_READ to CISREAD;
GRANT CREATE SESSION TO CISREAD;
```

Create Operational User (CISOPR):

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

Note: Oracle Utilities Application Framework and Oracle Utilities Work and Asset Management do not use the database edition feature. Hence, the product does not make schema (CISADM) explicitly editionable.

6. Review the Storage.xml file under the FW\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 spl-oradbi-25.4.jar while installing the Oracle Utilities Work and Asset Management database objects. See the Updating Storage.xml section 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 example: TableSpace Name). spl-oradbi-25.4.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>
```

Note: For ADB, tablespaces cannot be created manually. Storage.xml file should be updated with the Oracle provided tablespace DATA.

```
<CI_ACCT>
<TABLESPACE>DATA</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 ensure that the Database Management System software is installed according to your site standards and the installation guide provided by the database vendor. Also, make sure 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.

Extended Datatypes

Some of the Oracle Utilities Application Framework application table varchar2 fields require byte size beyond 4000 bytes to store data for new application requirements. To support this requirement the Oracle Utilities Application Framework database should use the Extended Data Types - Oracle database 12c feature (EXTENDED - the 32767 byte limit that was introduced in Oracle Database 12c is applicable).

Enable the Extended Data Types by setting DB parameter, max_string_size = EXTENDED.

Follow the instructions provided in Oracle Database online documentation for including this change in your database.

Important! This change in your database environment is mandatory. If not included it will lead to errors during the V25.4 upgrade.

Installing the Oracle Database

This section describes how to install the Oracle database for Oracle Utilities Work and Asset Management 25.4, including:

- Database Scripts and Utilities
- Initial Install (Installing V25.4 for the First Time)
- Upgrade Install
- Demo Install

Note: The installation tools outlined in this guide run on Linux, Unix, or Windows only. See the Supported Database Platforms section in Database Overview for more information on supported database platforms.

Database Scripts and Utilities

Note that installation scripts can be run in a Linux or Windows server.

Initial Install (Installing V25.4 for the First Time)

This section describes an initial installation of the V25.4 database. It focuses on the following:

- Copying and Decompressing Install Media
- Database Creation
- Installing the CISADM Schema

Note: You must have a supported version of the Java Development Kit installed on the Windows desktop where you stage and run the database installation package. See the *Oracle Utilities Work and Asset Management Installation Guide* for more information.

Copying and Decompressing Install Media

To copy and decompress the database:

1. Download Oracle Utilities Application Framework V25.4-Database Installation Media, Oracle Utilities Application Framework V25.4 Single Fix Prerequisite Database Rollup for WAM V25.4 (if there is any), Oracle Utilities Work and Asset Management V25.4 Oracle Database Multiplatform, and Oracle Utilities Work and

Asset Management V25.4 Single Fix Database Rollup MultiPlatform (if there is any) from the Oracle Software Delivery Cloud.

2. Copy Oracle Utilities Application Framework V25.4 Database Installation Media, WAM-V25.4-FW-Database- PREREQ-MultiPlatform (if there is any), Oracle Utilities Work and Asset Management V25.4 Oracle Database Multiplatform, and WAM-V25.4-Database-Rollup-MultiPlatform (if there is any) directories to your local machine.

   These files contain all the database components required to install the Oracle Utilities Application Framework and Oracle Utilities Work and Asset Management database.

Database Creation

> **Note:** You must have Oracle Database Server installed on your machine in order to create the database. This step is not required if you are performing a database upgrade from a previous version of Oracle Utilities Work and Asset Management.

Creating the Database on Linux/UNIX

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

Refer to the Creating the Database section 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.

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

Database Globalization Support Consideration

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.

By default, the database is created with BYTE length semantics. This setting should be modified to use CHAR length semantics by setting NLS_LENGTH_SEMANTICS to CHAR at session level via a logon trigger during installation as shown below.

Example:

```
CREATE OR REPLACE TRIGGER RCU_INSTALL_TRIGGER after logon on
database
declare
user_name varchar2(100);
begin
select user into user_name from dual;
if ( user_name LIKE 'CISADM' or user_name LIKE 'STG%' ) THEN
execute immediate 'alter session set nls_length_semantics=CHAR';
END IF;
END;
```

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

- **By script**: For details, refer to the *Doc ID 313175.1* on My Oracle Support.
- **Alternative procedure**: Refer to the Initial Install section for steps to create a schema with char length semantics.

Initial Install

1. Create the database using DBCA. Refer to the Creating the Database section for steps to create the database.
2. Run the following statement to set nls_length_semantics=CHAR.

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

> **Note**: Make sure to set nls_length_semantics=CHAR on the pluggable DB level only.

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

```
SQL> SHOW PARAMETER nls_length_semantics
```

Extended Datatypes

> **Note**: This step is applicable only for the on-premises databases. By default, ADB is set to EXTENDED.

Some of the Oracle Utilities Application Framework application table varchar2 fields require byte size beyond 4000 bytes to store data for new application requirements. To support this requirement the Oracle Utilities Application Framework database should use the Extended Data Types - Oracle database 12c feature (EXTENDED - the 32767 byte limit introduced in Oracle Database 12c applies.).

Enable the Extended Data Types by setting DB parameter, max_string_size = EXTENDED.

Follow the instructions provided in Oracle database documentation for including this change in your database.

> **Important!** This change in your database environment is mandatory. If not included it will lead to errors during the upgrade.

Enabling DBMS_CRYPTO package

Before installing Oracle Utilities Application Framework 25.4 make sure to provide execute privilege on dbms_crypto package to CISADM user. Execute the following command as SYS for on-premises and ADMIN for ADB.

```
grant execute on dbms_crypto to CISADM;
```

Installing the CISADM Schema

You should install Oracle Utilities Application Framework V25.4 and Oracle Utilities Application Framework V25.4 Single Fix Prerequisite Database Rollup for WAM V25.4 (if there is any) prior to installing Oracle Utilities Work and Asset Management V25.4.

The files for Oracle Utilities Application Framework installers are located in the Oracle Utilities Application Framework V25.4 Database Installation Media Install-Upgrade folder.

The installation process prompts you for the following information:

- The target database name in which the product is to be installed.
- A database user that will own the application schema (Example: CISADM).
- A database user that has read-write (select/update/insert/delete) privileges to the objects in the application schema. (Example: CISUSER).

The application will access the database as this user.

- A database user with read-only privileges to the objects in the application schema. (Example: CISREAD).
- A database role that has read-write (select/update/insert/delete) privileges to the objects in the application schema. The application will access the database as this user. (Example: CIS_USER).
- A database role with read-only privileges to the objects in the application schema. (Example: CIS_READ).
- Location for jar files. (The Jar files are bundled with the database package.)
- Java Home (Example: C:\Java\jdk)

This section focuses on the following:

- Installing the Oracle Utilities Application Framework Database Component
- Installing Framework Prerequisite Database Single Fixes
- Installing Oracle Utilities Work and Asset Management Database Component
- Installing Oracle Utilities Work and Asset Management Database Rollup
- Tasks Performed by ORADBI
- Post-installation Tasks

Installing the Oracle Utilities Application Framework Database Component

Note: Oracle Utilities Application Framework Database Component can be installed using OraDBI.java. While prior versions of the product have included OraDBI.exe, this is no longer supported going forward as this does not support latest functionality/features introduced in spl-oradbi-25.4.jar is delivered in directory jarfiles.

This section includes the instructions to install the database component.

Installing the Oracle Utilities Application Framework Database Component Using OraDBI.java

OraDBI.java is a new tool to install and upgrade database components. It can be run from Linux, UNIX, or Windows machines that has the following installed:

- Oracle 19c
- Oracle 19c Client

Before installing the database component, make sure the following prerequisites are met:

- JDK 17
- Oracle Database
- Schema (such as CISADM) should exist in the database
- max_string_size is set to EXTENDED
- For ADB, the database connection should be working. For more information, seee step 1.b.a.c in the Creating the Database section.

To install the Oracle Utilities Application Framework V25.4:

1. Unzip the Oracle Utilities Application Framework V25.4 Database Installation Media package.
2. Set JAVA_HOME, PATH, and CLASSPATH.

   Note: For ADB installation, export/SET TNS_ADMIN=<wallet location>.

   Linux/UNIX:

   ```
   export JAVA_HOME=/scratch/software/jdk
   export PATH=$JAVA_HOME/bin:$PATH
   export CLASSPATH=/../FW-V25.4-Oracle- Database-Multiplatform/FW/
   jarfiles/*
   ```

   WINDOWS:

   ```
   SET JAVA_HOME=C:\Program Files\Java\jdk
   SET PATH=%JAVA_HOME%\bin:%PATH%
   SET CLASSPATH= C:\FW-V25.4-Oracle-Database-
   Multiplatform\FW\jarfiles\*
   ```

3. Run the following command from the command line or command prompt from the Oracle Utilities Application Framework V25.4 Database Installation Media Install-Upgrade folder.

   There are two options available to execute OraDBI.java:

   - Using interactive mode
   - Using command on command line

   Using Interactive Mode:

   Linux/UNIX:

   ```
   java -Xmx1500M com.oracle.ouaf.oem.install.OraDBI -p
   <RW_USERPASS>,<R_USERPASS>
   ```

   WINDOWS

   ```
   "C:\Program Files\Java\jdk"\bin\java -Xmx1500M -cp C:\FW-V25.4-
   Oracle-Database-Multiplatform\FW\jarfiles\*
   com.oracle.ouaf.oem.install.OraDBI -p <RW_USERPASS>,<R_USERPASS>
   ```

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

   - Enter the database server hostname:<SERVER NAME>
   - Enter the database port number:<PORT>
   - Enter the database name/SID:<DB NAME>
   - Enter your database username:<CISADM>
   - Enter your password for username CISADM:
   - Enter the Oracle user with read-write privileges to Database Schema:<CISUSER>
   - Enter your password for username CISUSER:
   - Enter the Oracle user with read-only privileges to Database Schema:<CISREAD>
   - Enter your password for username CISREAD:

- Enter the database role with read-write privileges to Database Schema:<CIS_USER>
- Enter the database role with read-only privileges to Database Schema:<CIS_READ>
- Enter the name of the target Schema where you want to install or upgrade:<CISADM>

Using the Command Line:

Linux/UNIX:

- On-premises:

```
java -Xmx1500M com.oracle.ouaf.oem.install.OraDBI -d
jdbc:oracle:thin:@<DB_SERVER>:<PORT>/
<SID>,<DBUSER>,<DBPASS>,<RW_USER>,<R_USER>,<RW_USER_ROLE>,<R_US
ER_
ROLE>,<DBUSER> -p <RW_USERPASS>,<R_USERPASS>
```

- ADB:

```
java -Xmx1500M com.oracle.ouaf.oem.install.OraDBI -d
jdbc:oracle:thin:@<tnsname>?TNS_ADMIN=<wallet
location>,<DBUSER>,<DBPASS>,<RW_USER>,<R_USER>,<RW_USER_ROLE>,<
R_US
ER_
ROLE>,<DBUSER> -p <RW_USERPASS>,<R_USERPASS>
```

Windows:

- On-premises:

```
"C:\Program Files\Java\jdk"\bin\java -Xmx1500M -cp C:\FW-V25.4-
Oracle-Database-Multiplatform\FW\jarfiles\*
com.oracle.ouaf.oem.install.OraDBI -d
jdbc:oracle:thin:@<DB_SERVER>:<PORT>/
<SID>,<DBUSER>,<DBPASS>,<RW_USER>,<R_USER>,<RW_USER_ROLE>,<R_USER
ROLE>,<DBUSER> -p <RW_USERPASS>,<R_USERPASS>
```

- ADB:

```
"C:\Program Files\Java\jdk"\bin\java -Xmx1500M -cp
C:\..\FW\jarfiles\* com.oracle.ouaf.oem.install.OraDBI -d
jdbc:oracle:thin:@<tnsname>?TNS_ADMIN=<wallet
location>,<DBUSER>,<DBPASS>,<RW_USER>,<R_USER>,<RW_USER_ROLE>,<R_U
S ER ROLE>,<DBUSER> -p <RW_USERPASS>,<R_USERPASS>
```

This process generates log files in the directory ..\FW\Install-Upgrade\logs.

4. Make sure to check the log files for any errors.

   Note: For OraDBI java, you may receive the following message in the display output or logs. These errors can be safely ignored and the process should proceed to completion.

   - 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 Framework Prerequisite Database Single Fixes

While prior versions of the product have included the cdxpatch.exe programs for applying DB Hot Fixes, this is no longer supported going forward; the ouafDatabasePatch.cmd or ouafDatabasePatch.sh should be used instead. The new tool can be run from Linux, UNIX, or Windows machines.

Important: Confirm if there are Prerequisite Database Single Fixes. Check if there is a WAM-V25.4-FW-Database-PREREQ-MultiPlatform folder. If none, skip this step and proceed to apply Oracle Utilities Work and Asset Management, else continue with the steps below.

Applying Hot Fixes

Note: Java 17 JDK should be installed on the machine to use the command. Make sure to install the JDK that is supported for your platform.

1. Extract ..\WAM-V25.4-FW-Database-PREREQ-MultiPlatform\FW254-HFix\db_patch_standalone.jar to any directory on your local machine under dbpatch_tools folder.

   Linux/**UNIX:**

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

   WINDOWS:

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

2. SET TOOLSBIN.

   Linux/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-V25.4-FW-Database-PREREQ-MultiPlatform\FW254-HFix folder.

   Note: For ADB, the database connection should be working. For more information, see step 1.b.a.c in the Creating the Database section and export/SET TNS_ADMIN=<wallet location>.

   Linux/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

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:
  - On-premises: <DB_Server:DBPORT/ORACLE_SID>
  - ADB: @<tnsname>?TNS_ADMIN=<wallet location>

Installing Oracle Utilities Work and Asset Management Database Component

Oracle Utilities Work and Asset Management Database Component can be installed using OraDBI.java.While prior versions of the product have included OraDBI.exe, this is no longer supported going forward as this does not support latest functionality/features introduced in OraDBI.java. OraDBI.jar is delivered in directory jarfiles.

Installing the Oracle Utilities Work and Asset Management Database Component Using OraDBI.java

OraDBI.java is a new tool to install and upgrade database components. It can be run from UNIX or Windows machines that has the following installed:

- Oracle 19c
- Oracle 19c Client

Before installing the database component, make sure the below prerequisites are met.

- JDK 17
- Oracle Database
- Schema (such as CISADM) should exist in the database
- max_string_size is set to EXTENDED
- For ADB, the database connection should be working. For more information, see in the Creating the Database section.

To install the Oracle Utilities Work and Asset Management V25.4:

1. Unzip WAM-V25.4-Oracle-Database-Multiplatform.zip package.
2. Set JAVA_HOME, PATH, and CLASSPATH.

   Note: For ADB installation, export/SET TNS_ADMIN=<wallet location>.

Linux/UNIX:

```
export JAVA_HOME=/scratch/software/jdk
export PATH=$JAVA_HOME/bin:$PATH
export CLASSPATH=/../FW-V25.4-Oracle- Database-Multiplatform/FW/
jarfiles/*
```

WINDOWS:

```
SET JAVA_HOME=C:\Program Files\Java\jdkSET
PATH=%JAVA_HOME%\bin:%PATH%
SET CLASSPATH= C:\ FW-V25.4-Oracle-Database-
Multiplatform\FW\jarfiles\*
```

3. Run the following command from the command line or command prompt from ..\WAM\Upgrade\Install-Upgrade folder.

   There are two options available to execute OraDBI.java

   - Using interactive mode
   - Using command on command line

   Using Interactive Mode:

   Linux/UNIX:

   ```
   java -Xmx1500M com.oracle.ouaf.oem.install.OraDBI -p
   <RW_USERPASS>,<R_USERPASS>
   ```

   WINDOWS:

   ```
   "C:\Program Files\Java\jdk"\bin\java -Xmx1500M -cp C:\ FW-V25.4-
   Oracle-Database-Multiplatform\FW\jarfiles\*
   com.oracle.ouaf.oem.install.OraDBI -p <RW_USERPASS>,<R_USERPASS>
   ```

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

   - Enter the database server hostname:<SERVER NAME>
   - Enter the database port number:<PORT>
   - Enter the database name/SID:<DB NAME>
   - Enter your database username:<CISADM>
   - Enter your password for username CISADM:
   - Enter the Oracle user with read-write privileges to Database Schema:<CISUSER>
   - Enter your password for username CISUSER:
   - Enter the Oracle user with read-only privileges to Database Schema:<CISREAD>
   - Enter your password for username CISREAD:
   - Enter the database role with read-write privileges to Database Schema:<CIS_USER>
   - Enter the database role with read-only privileges to Database Schema:<CIS_READ>
   - Enter the name of the target schema where you want to install or upgrade:<CISADM>

Using the Command Line:

Linux/UNIX:

- On-premises:

```
java -Xmx1500M com.oracle.ouaf.oem.install.OraDBI -d
jdbc:oracle:thin:@<DB_SERVER>:<PORT>/
<SID>,<DBUSER>,<DBPASS>,<RW_USER>,<R_USER>,<RW_USER_ROLE>,<R_US
ER_ROLE>,<DBUSER> -p <RW_USERPASS>,<R_USERPASS>
```

- ADB:

```
java -Xmx1500M com.oracle.ouaf.oem.install.OraDBI -d
jdbc:oracle:thin:@<tnsname>?TNS_ADMIN=<wallet
location>,<DBUSER>,<DBPASS>,<RW_USER>,<R_USER>,<RW_USER_ROLE>,<
R_USER_ROLE>,<DBUSER> -p <RW_USERPASS>,<R_USERPASS>
```

WINDOWS:

- On-premises:

```
"C:\Program Files\Java\jdk"\bin\java -Xmx1500M -cp C:\FW-V25.4-
Oracle-Database-Multiplatform\FW\jarfiles\*
com.oracle.ouaf.oem.install.OraDBI -d
jdbc:oracle:thin:@<DB_SERVER>:<PORT>/
<SID>,<DBUSER>,<DBPASS>,<RW_USER>,<R_USER>,<RW_USER_ROLE>,<R_US
ER ROLE>,<DBUSER> -p <RW_USERPASS>,<R_USERPASS>
```

- ADB:

```
"C:\Program Files\Java\jdk"\bin\java -Xmx1500M -cp
C:\..\FW\jarfiles\* com.oracle.ouaf.oem.install.OraDBI -d
jdbc:oracle:thin:@<tnsname>?TNS_ADMIN=<wallet
location>,<DBUSER>,<DBPASS>,<RW_USER>,<R_USER>,<RW_USER_ROLE>,<
R_USER ROLE>,<DBUSER> -p <RW_USERPASS>,<R_USERPASS>
```

This process generates log files in the directory ..\WAM\Upgrade\Install-Upgrade\logs.

4. Make sure to check the log files for any errors.

   Note: For OraDBI java, you may receive the following message in the display output or logs. These errors can be safely ignored and the process should proceed to completion.

   -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 Work and Asset Management Database Rollup

The previous versions of Oracle Utilities Work and Asset Management included cdxpatch.exe programs for applying the database hot fixes. This is not supported from the current version. Starting Oracle Utilities Work and Asset Management V25.4, use the ouafDatabasePatch.cmd or ouafDatabasePatch.sh. The new tool can be run from Linux, UNIX, or Windows machines.

> **Important!** Proceed with the steps in this section only if the installation package contains the WAM-V25.4-Database-Rollup-MultiPlatform folder.

Make sure Java 17 JDK is installed on the machine to use the commands. The JDK version that is supported for your platform should be installed.

To apply Oracle Utilities Work and Asset Management V25.4 Rollup:

1. Perform the steps 1 and 2 listed in the Installing Framework Prerequisite Database Single Fixes section.
2. Apply WAM 254 Rollup from the ..\ WAM-V25.4-Database-Rollup-MultiPlatform\WAM254-HFix folder.

   > **Note**: For ADB, the database connection should be working. For more information, see step 1.b.a.c in the Creating the Database section. export/SET TNS_ADMIN=<wallet location>.

   Linux/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:
  - On-premises: <DB_Server:DBPORT/ORACLE_SID>
  - ADB: @<tnsname>?TNS_ADMIN=<wallet location>

Tasks Performed by ORADBI

The tasks performed by ORADBI are as follows:

- Interacts with the user to collect information about the name of Oracle account that will own the application schema (example: CISADM), password of this account, and the name of the Oracle account that the application user will use

(example: CISUSER), and the name of the Oracle account that will be assigned read-only privileges to the application schema (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.

Post-installation Tasks

Enabling USER_LOCK Package

For the 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, please do so as follows:

1. Login as SYS user for on-premises and as ADMIN for ADB.

   Note: For ADB, the database connection should be working. For more information, see the step 1.b.a.c in the Creating the Database.

2. At the SQL prompt, run the following:

   ```
   @?/rdbms/admin/userlock.sql
   ```

3. Grant permission.

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

Note that grant can also be made to the database user which the application connects to only instead of public. Example: cisuser

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.

Upgrade Install

This section describes how to upgrade the database components for Oracle Utilities Work and Asset Management, including:

Copying and Decompressing Install Media

To copy and decompress the Oracle Utilities Work and Asset Management database:

1. Download Oracle Utilities Application Framework V25.4 Database Installation Media, Oracle Utilities Application Framework V25.4 Single Fix Prerequisite Database Rollup for WAM V25.4 (if there is any), Oracle Utilities Work and Asset Management V25.4 Oracle Database Multiplatform, and Oracle Utilities Work and Asset Management V25.4 Single Fix Database Rollup MultiPlatform (if there is any) from the Oracle Software Delivery Cloud.
2. Copy Oracle Utilities Application Framework V25.4 Database Installation Media, WAM-V25.4-FW-Database- PREREQ-MultiPlatform (if there is any), Oracle Utilities Work and Asset Management V25.4 Oracle Database Multiplatform, and WAM-V25.4-Database-Rollup-MultiPlatform (if there is any) directories to your local machine. These files contain all the database components required to install the Oracle Utilities Application Framework and Oracle Utilities Work and Asset Management database.

Migrating from BYTE Based Storage to CHAR Based Storage

This section is mandatory for every Oracle Utilities Work and Asset Management using BYTE semantic.

1. Set nls_length_semantics=CHAR.

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

   Note: Make sure to set nls_length_semantics=CHAR on the pluggable DB level only.

2. Restart the database.
3. Make sure nls_length_semantics is CHAR.

```
SQL> SHOW PARAMETER nls_length_semantics
```

4. Export schema from the database that has nls_length_semantics=BYTE.

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

5. Generate DDL from dump file using Oracle impdp utility.

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

6. Replace "Byte" with "Char" in <schema_name>DDL.sql.

   For vi editor (in Linux), use the following command to replace Byte to Char.

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

7. Replace the schema name also if it is required for environment.
8. Run <schema_name>DDL.sql (generated in step 6) that creates objects in the schema.

   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;
```

9. If an object is missing for any reason, create it by fixing DDL manually (DDL for each object is available in the file which was created in step 6).

   Run DDL for the objects that are not created.

10. Generate DDL to disable triggers.

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

11. Run the script generated from step 11 to disable all triggers.
12. Import the data only.

    Note: For ADB, follow the instructions in the Importing the Demo Dump File section.

    To import data only into the schema created to support CHAR based database storage:

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

13. Enable the triggers.

    To generate DDL for triggers:

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

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

Database Globalization Support Consideration

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.

By default, the database is created with BYTE length semantics. This setting should be modified to use CHAR length semantics by setting NLS_LENGTH_SEMANTICS to CHAR at session level via a logon trigger during installation as follows:

Example:

```
CREATE OR REPLACE TRIGGER RCU_INSTALL_TRIGGER after logon on
database
declare
user_name varchar2(100);
begin
select user into user_name from dual;
if ( user_name LIKE 'CISADM' or user_name LIKE 'STG%' ) THEN
execute immediate 'alter session set nls_length_semantics=CHAR';
END IF;
END;
```

Extended Datatypes

> **Note**: This step is applicable only for the on-premises databases. By default, ADB is set to EXTENDED.

Some of the Oracle Utilities Application Framework application table varchar2 fields require byte size beyond 4000 bytes to store data for new application requirements. To support this requirement the Oracle Utilities Application Framework database should use the Extended Data Types - Oracle database 12c feature (EXTENDED - the 32767 byte limit introduced in Oracle Database 12c applies.).

Enable the Extended Data Types by setting DB parameter, max_string_size = EXTENDED.

Follow the instructions provided in Oracle database documentation for including this change in your database.

> **Important!** This change in your database environment is mandatory. If not included, it will lead to errors during the upgrade.

Exclude Table/Index

To exclude an index or table during the upgrade process:

1. Edit the OraSchUpg.inp file in the Install-Upgrade directory.
2. Add the tables and indexes in the following format:

   ```
   -INDEX: 'INDEX_NAME','INDEX_NAME'
   -TABLE: 'TABLE1_NAME','TABLE2_NAME'
   ```

   Example: To exclude the F1_WEB_SVC table, use the following:

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

   If there are multiple tables, include them with separated commas.

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

   Similarly, for indexes:

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

Grant Privilege to Database Roles

Before running the upgrade, make sure to grant the Create Synonym to the database read write (CIS_USER) and read only (CIS_READ) roles as SYS for on-premises and as ADMIN for ADB.

```
grant CREATE SYNONYM to CIS_USER;
grant CREATE SYNONYM to CIS_READ;
```

Enabling the DBMS_CRYPTO Package

Before installing Oracle Utilities Application Framework V25.4 make sure to provide execute privilege on dbms_crypto package to CISADM user.

Execute the following command as SYS for on-premises and as ADMIN for ADB:

```
grant execute on dbms_crypto to CISADM;
```

Enabling DBMS_RLS Package

Before installing Oracle Utilities Application Framework V25.4 make sure to provide execute privilege on dbms_rls package to CISADM user. Execute the following command as SYS for on-premises and as ADMIN for ADB.

```
grant execute on dbms_rls to CISADM;
```

This upgrade section assumes an existing Oracle Utilities Work and Asset Management on top of Oracle Utilities Application Framework. The following upgrade paths are described:

- Upgrading from V2.4.0.1.1 to V25.4
-
-
-
-

Upgrading from V2.4.0.1.1 to V25.4

You must install the Oracle Utilities Application Framework V25.4 and Oracle Utilities Application Framework V25.4 Database Single Fix Prerequisite Rollup for Oracle Utilities Work and Asset Management V25.4 (if there is any) prior to Oracle Utilities Work and Asset Management V25.4. The files for the Oracle Utilities Application Framework installer are located in the Oracle Utilities Application Framework V25.4 Database Installation Media Install-Upgrade folder.

Installing the Oracle Utilities Application Framework Database Component Using OraDBI.java

For instructions, refer to Installing the Oracle Utilities Application Framework Database Component Using OraDBI.java.

Installing Prerequisite Database Single Fixes

For instructions, refer to Installing Framework Prerequisite Database Single Fixes.

Installing the Oracle Utilities Work and Asset Database Component

For instructions, refer to Installing Oracle Utilities Work and Asset Management Database Component.

Installing the Oracle Utilities Work and Asset Database Rollup

For instructions, refer to Installing Oracle Utilities Work and Asset Management Database Rollup.

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.

Tasks Performed by ORADBI

For more information, refer to Tasks Performed by ORADBI.

Demo Install

This section describes how to install the demo database components for Oracle Utilities Work and Asset Management, including:

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

Copying and Decompressing Install Media

To copy and decompress the database:

1. Download Oracle Utilities Application Framework V25.4 Database Installation Media, Oracle Utilities Application Framework V25.4 Single Fix Prerequisite Database Rollup for WAM V25.4 (if there is any), Oracle Utilities Work and Asset Management V25.4 Oracle Database Multiplatform, and Oracle Utilities Work and Asset Management V25.4 Single Fix Database Rollup MultiPlatform (if there is any) from the Oracle Software Delivery Cloud.
2. Copy Oracle Utilities Application Framework V25.4 Database Installation Media, WAM-V25.4-FW-Database-PREREQ-MultiPlatform (if there is any), Oracle Utilities Work and Asset Management V25.4 Oracle Database Multiplatform and WAM-V25.4-Database-Rollup-MultiPlatform (if there is any) directories to your local machine.

   These files contain all the database components required to install the Oracle Utilities Application Framework and Work and Asset Management database.

Creating the Database

Note: You must have Oracle Database Server 19c installed on your machine to create the database.

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

Creating the Database on Linux/UNIX

Create the database using the Database Configuration Assistant (DBCA). Refer to the Creating the Database section 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.

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

Database Storage BYTE/CHAR

Database created by default will store data in BYTE. This setting should be modified to use CHAR length semantics. To store data in CHAR, refer to the instructions below:

Initial Install

1. Create the database using DBCA.
2. Set nls_length_semantics=CHAR

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

> **Note**: Make sure to set nls_length_semantics=CHAR on the pluggable DB level only.

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

```
SQL> SHOW PARAMETER nls_length_semantics
```

Upgrade and Migration from BYTE Based Storage to CHAR Based Storage

1. Run the following statement to set nls_length_semantics=CHAR.

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

> **Note**: Make sure to set nls_length_semantics=CHAR on the pluggable DB level only.

2. Restart the database.
3. Make sure nls_length_semantics is CHAR.

```
SQL> SHOW PARAMETER nls_length_semantics
```

4. Export schema from the database that has nls_length_semantics=BYTE.

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

5. Generate DDL from dump file using Oracle impdp utility.

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

6. Replace "BYTE" with "CHAR" in <schema_name>DDL.sql.

   For vi editor (in Linux), use the following command to replace BYTE to CHAR.

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

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

8. Run <schema_name>DDL.sql (generated in step 6) that creates objects in the schema.

   Run the following command to ensure 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;
   ```

9. If an object is missing for any reason, create it by fixing DDL manually (DDL for each object is available in the file which was created in step 6).

   Run DDL for the objects that are not created.

10. Generate DDL to disable triggers using following command:

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

11. Run the script generated from step 11 to disable all triggers.

12. Import the data only.

    Note: For ADB, follow the instructions in the Importing the Demo Dump File section.

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

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

13. Enable the triggers.

    Use the following command to generate DDL for triggers.

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

14. Run the script generated from step No.13 to enable all triggers.

Extended Datatypes

Note: This step is applicable for on-premises databases only. By default, ADB is set to EXTENDED.

Some of the Oracle Utilities Application Framework application table varchar2 fields require byte size beyond 4000 bytes to store data for new application requirements. To support this requirement the Oracle Utilities Application Framework database should use the Extended Data Types - Oracle database 12c feature (EXTENDED - the 32767 byte limit introduced in Oracle Database 12c applies.).

Enable the Extended Data Types by setting DB parameter, max_string_size = EXTENDED.

Follow the instructions provided in Oracle database documentation for including this change in your database.

Important! This change in your database environment is mandatory. If not included, it will lead to errors during the V25.4 upgrade.

Importing the Demo Dump File

After a successful database creation, import the demo data.

1. For on-premises database:
   a. Set the correct ORACLE_SID and ORACLE_HOME.
   b. Make sure max_string_size is set to EXTENDED.

      Note: This step is applicable for on-premises databases only.

   c. Proceed to step 3.
2. For autonomous database (ADB):
   a. Make sure the database connection is working. For more information, see step 1.b.a.c in the Creating the Database section.
   b. Upload the dumpfile in OCI Storage bucket and obtain the dumpfile URL.
   c. Create the cloud credentials in the database.

      Note: Execute the script as a ADMIN user. The authentication token can be generated in **User Settings** in the OCI profile.

      ```
      SET DEFINE OFF
      BEGIN
        DBMS_CLOUD.CREATE_CREDENTIAL(
          credential_name => '<credential name>',
          username => '<OCI Profile>',
          password => '<authentication token>'
        );
      END;
      /
      ```

   d. Proceed to step 4.
3. Import the demo dump.

   Note: Make sure the ..\WAM-V25.4-DemoData\Demo\exp_demo.dmp.gz file is extracted and available in data_pump_dir's location before running the following import command.

   On-premises:

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

   ADB:

   ```
   impdp USERNAME/password@<tns_name> directory=DATA_PUMP_DIR
   credential=<credential name> dumpfile=<dumpfile url in OCI>
   logfile=import_demo.log SCHEMAS=CISADM
   ```

4. After the import is complete, enable the DBMS_CRYPTO and and DBMS_RLS packages. Execute the following command as SYS for on-premises and as ADMIN for ADB.

   ```
   grant execute on dbms_crypto to CISADM;
   grant execute on dbms_rls to CISADM;
   ```

Configuring Security

The configuration security utility and scripts are already part of the delivered jarfiles in the Oracle Utilities Application Framework V25.4 Database Installation Media jarfiles directory. It can be executed from a Linux or a Windows machine.

Please note the following:

- Database vault must be disabled before running.
- Interactive mode for this utility is currently not working.
- `-f` parameter is not working and will be deprecated soon.

The utility configures security for the application owner schema objects.

Oragensec, by default, grants permissions to CIS_USER and CIS_READ roles. To use site-specific roles, execute Oragensec after providing the command line options and specifying the specific roles.

OraGenSec Java Usage

```
java OraGenSec [-a <arg>] [-d <arg>] [-f <arg>] [-h] [-l <arg>] [-o
<arg>] [-q] [-r <arg>] [-u <arg>]
```

OraGenSec Help

- -a <arg>: Generate security for all objects in the database
- -d <arg>: DB connection string as: $DB_USER,$DB_PWD,$DB_CONNECTION_STRING,$TARGET_SCHEMA

  Note: Make sure the syntax for $DB_CONNECTION is as below:

  - On-premises: jdbc:oracle:thin: :@<DB_SERVER>:<PORT>/<SID>
  - ADB: jdbc:oracle:thin:@<tnsname>?TNS_ADMIN=<wallet location>
- -f <arg>: Generate security for specific objects from an input file
- -h: Print Help
- -l <arg>: Name of log file (optional)
- -o <arg>: Generate security for comma separated list of objects
- -p <arg>: Corresponding passwords of users to create synonyms for
- -q: Quiet mode
- -r <arg>: Roles corresponding to the users
- -u <arg>: Read Write user, Read Only user

To run the utility:

Note: For ADB, export/SET TNS_ADMIN=<wallet location>

1. Set JAVA_HOME, PATH, and CLASSPATH.

   Linux/UNIX:

   ```
   export JAVA_HOME=/scratch/software/jdk
   export PATH=$JAVA_HOME/bin:$PATH
   export CLASSPATH=/../FW-V25.4-Oracle-
   Database-Multiplatform/FW/jarfiles/*
   ```

   WINDOWS:

```
SET JAVA_HOME=C:\Program Files\Java\jdk
SET PATH=%JAVA_HOME%\bin:%PATH%
SET CLASSPATH= C:\ FW-V25.4-Oracle-Database-
Multiplatform\FW\jarfiles\*
```

2. From any directory, run Oragensec. Make sure that Step 1 is complete. Run the following command:

   Linux/UNIX

   - On-premises:

     ```
     java -Xmx1500M com.oracle.ouaf.oem.install.OraGenSec-d
     <DBUSER>,<DBPASS>, jdbc:oracle:thin:@<DB_SERVER>:<PORT>/
     <SID> -u <RW_USER>,<R_USER> -r <RW_USER_ROLE>,<R_USER_ROLE> -a A
     -p <RW_USERPASS>,<R_USERPASS>
     ```

   - ADB:

     ```
     java -Xmx1500M com.oracle.ouaf.oem.install.OraGenSec-d
     <DBUSER>,<DBPASS>,
     jdbc:oracle:thin:@<tnsname>?TNS_ADMIN=<wallet location> -u
     <RW_USER>,<R_USER> -r <RW_USER_ROLE>,<R_USER_ROLE> -a A -p
     <RW_USERPASS>,<R_USERPASS>
     ```

   WINDOWS:

   - On-premises:

     ```
     "%JAVA_HOME%"\bin\java -Xmx1500M
     com.oracle.ouaf.oem.install.OraGenSec -d <DBUSER>,<DBPASS>,
     jdbc:oracle:thin:@<DB_SERVER>:<PORT>/
     <SID> -u <RW_USER>,<R_USER> -r <RW_USER_ROLE>,<R_USER_ROLE> -a A
     -p <RW_USERPASS>,<R_USERPASS>
     ```

   - ADB:

     ```
     "%JAVA_HOME%"\bin\java -Xmx1500M
     com.oracle.ouaf.oem.install.OraGenSec -d <DBUSER>,<DBPASS>,
     jdbc:oracle:thin:@<tnsname>?TNS_ADMIN=<wallet location> -u
     <RW_USER>,<R_USER> -r <RW_USER_ROLE>,<R_USER_ROLE> -a A -p
     <RW_USERPASS>,<R_USERPASS>
     ```

Chapter 3

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

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 example: 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

   - OF stands for Owner Flag. For example: The framework is F1. Other examples are M1,C1,D1,D2, etc.

   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.

<<pDefault 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

In general, referential integrity is enforced by the application and the Foreign Key constraints are not defined in the database. Indexes are created on most of Foreign Key columns to make sure desired performance characteristics. However in the specific case of ILM implementation, some of the tables require Foreign Key constraints due to the referential partitioning.

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 4

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 includes 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. This section includes:

- Index
- Table Partitioning Recommendations
- Transparent Data Encryption Recommendations
- Data Compression Recommendations
- Database Vault Recommendations
- Oracle Fuzzy Search Support
- Storage Recommendations
- Database Statistic
- 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 server). 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
/
```

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=30000

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 (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 19c**: https://docs.oracle.com/en/database/oracle/oracle-database/19/dwhsg/basic-query-rewrite-materialized-views.html
   - **Troubleshoot Materialized View:** https://docs.oracle.com/en/database/oracle/oracle-database/19/admin/troubleshooting-problems-with-read-only-materialized-views.html

```
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_C
D 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;
```

Chapter 5

Conversion Tools

This chapter describes the following database conversion tools:

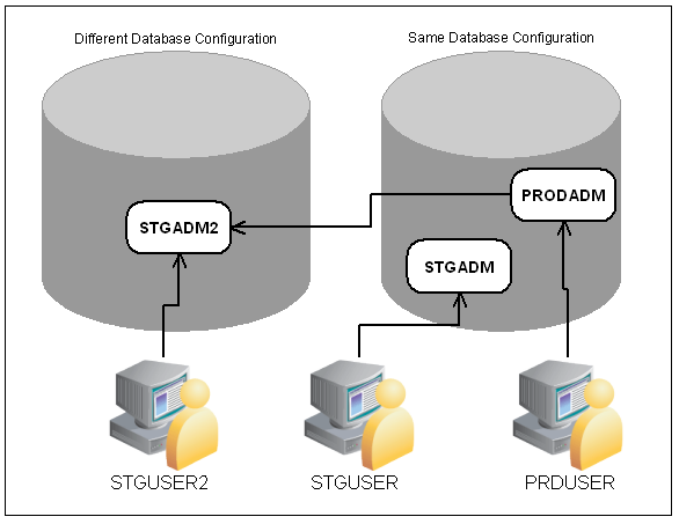
- Database Configuration
- Installing the Script
- Preparing the Production Database
- Preparing the Staging Database
- Upgrading the Production Database
- Upgrading the Staging Database

Note that all database related single fixes and service packs need to be applied against the production schema. Staging schema should not be updated with database single fixes or service packs. Staging schema need to be rebuilt for any fixes that contain DDL to create new database objects in production schema.

Database Configuration

The Conversion Tool Kit requires at least two sets of schema. One is to hold the staging data that the conversion tool gets the data from and performs validations. We call this schema the staging database. The target schema, which is referred to as the production database, is where the conversion tool inserts the validated data. Both the production database and the staging databases can reside in a single Oracle database or in different databases that are connected via a database link. Only the single database configuration is supported.

The following schematic diagram shows a sample configuration of both the production and staging environments in which the Conversion Tool Kit operates. The production and staging databases must be the same release level.



All the tables and views for the application are defined in the production database. The staging database has the same set of tables and views as the production database, except the tables that are grouped as part of the business configuration (control tables). Details on the differences of the tables of the two databases and of the conversion tool functionality are found in the Conversion Tool document.

Installing the Script

The Conversion Setup Utility is provided to set up conversion schemas. It can be run from UNIX or Windows machines that has the following installed:

- Either Oracle 19c or Oracle 19c Client as long as it can connect to the database
- JDK 17

The Conversion folder includes conversion setup utilities - ConvSetup.tar and scripts to create users.

ConversionSetup Java Usage

usage	com.oracle.ouaf.oem.conversion.ConversionSetup
-a, --apply	Apply Conversion setup on Staging Schema. '-s' is mandatory if '-a' is passed

-d,--dbConnString <arg>	DB connection string: Any "jdbc:oracle:thin"supported format [example: HOST:PORT/SID | HOST:PORT:SERVICE | <TNSSTRING>]
-o,--output <arg>	Output Directory for the generated files. Directory will be created if it doesn't already exist.
-p,--prodSchema <arg>	Primary/Production Schema and credentials to connect: CIS_ADMIN,CIS_PSWD. [example: CISADM,CISADM]
-r,--rollback	Revert Conversion setup on Staging Schema. '-s' is mandatory if '-r' is passed
-s,--stgSchema <arg>	Staging Schema and Staging users:STG_ADMIN,STG_PSWD,STG_RW_USER [example: STGADM,STGADM,STGUSER]

This section of the document describes how to create the databases for the conversion tool kit.

Preparing the Production Database

If the production database does not exist create the database under the production schema owner (CISADM).

> **Note:** If the production database is upgraded from the previous version of the application make sure all public synonyms that are created on the application tables are deleted. Instead, each application user should have private synonyms created on the application tables in order for the conversion tool configuration to work.

Preparing the Staging Database

After the staging owner (STGADM), application user (STGUSER) and read access user (STGREAD) are created, install the initial database option in the staging schema. The rest of the steps are listed below.

To run the utility:

1. Set JAVA_HOME, PATH, and CLASSPATH.

   For ADB, make sure the database connection is working. For more information, see step 1.b.a.c in the Creating the Database section and export/SET TNS_ADMIN=<wallet location>.

   Linux/UNIX:

   ```
   export JAVA_HOME=/scratch/software/jdk/
   export PATH=$JAVA_HOME/bin:$PATH
   export CLASSPATH=/../FW-V25.4-Oracle-Database- Multiplatform/FW/
   jarfiles/*
   ```

   Windows:

   ```
   SET JAVA_HOME=C:\Program Files\Java\jdk
   SET PATH=%JAVA_HOME%\bin:%PATH%
   ```

```
SET CLASSPATH= C:\FW-V25.4-Oracle-Database-
Multiplatform\FW\jarfiles\*
```

2. Run Conversion from any directory making sure Step 1 is complete. Run the command as necessary.

   Note: Execute the following command as ADMIN for ADB:

   ```
   grant dwrole to STGADM;
   grant dwrole to STGUSER;
   grant create any view to STGUSER;
   ```

   Linux/UNIX:

   If '-a' (apply on staging flag) is passed, it connects to the database as staging schema admin and performs the conversion setup, in addition to creation of the above files, by running the same SQLs present in the create* SQL files.

   On-premises:

   ```
   java com.oracle.ouaf.oem.conversion.ConversionSetup -d
   ${JDBC_CONN_STRING} -p ${CIS_ADMIN_USR},${CIS_ADMIN_PSWD} -a -s
   ${STG_ADMIN_USR},${STG_ADMIN_PSWD},${STG_RW_USER}
   ```

   ADB:

   ```
   java com.oracle.ouaf.oem.conversion.ConversionSetup -d
   jdbc:oracle:thin:@<tnsname>?TNS_ADMIN=<wallet location> -p
   ${CIS_ADMIN_USR},${CIS_ADMIN_PSWD} -a -s
   ${STG_ADMIN_USR},${STG_ADMIN_PSWD},${STG_RW_USER}
   ```

   If '-r' (revert staging flag) is passed, it connects to the database as staging schema admin and reverts the staging to a state prior to conversion setup, running the same SQLs present in drop_* or restore_* SQL files.

   On-premises:

   ```
   java com.oracle.ouaf.oem.conversion.ConversionSetup -d
   ${JDBC_CONN_STRING} -p ${CIS_ADMIN_USR},${CIS_ADMIN_PSWD} -r -s
   ${STG_ADMIN_USR},${STG_ADMIN_PSWD},${STG_RW_USER}
   ```

   ADB:

   ```
   java com.oracle.ouaf.oem.conversion.ConversionSetup -d
   jdbc:oracle:thin:@<tnsname>?TNS_ADMIN=<wallet location> -r -s
   ${STG_ADMIN_USR},${STG_ADMIN_PSWD},${STG_RW_USER}
   ```

   To generate ONLY the scripts, the program should be invoked with '-s' parameter with no other flags (no '-a' or '-r'):

   On-premises:

   ```
   java com.oracle.ouaf.oem.conversion.ConversionSetup -d
   ${JDBC_CONN_STRING} -p ${CIS_ADMIN_USR},${CIS_ADMIN_PSWD} -s
   ${STG_ADMIN_USR},${STG_ADMIN_PSWD},${STG_RW_USER}
   ```

   ADB:

   ```
   java com.oracle.ouaf.oem.conversion.ConversionSetup -d
   jdbc:oracle:thin:@<tnsname>?TNS_ADMIN=<wallet location> -s
   ${STG_ADMIN_USR},${STG_ADMIN_PSWD},${STG_RW_USER}
   ```

Windows:

If '-a' (apply on staging flag) is passed, it connects to the database as staging schema admin and performs the conversion setup, in addition to creation of the above files, by running the same SQLs present in the create* SQL files.

On-premises:

```
"%JAVA_HOME%"\bin\java
com.oracle.ouaf.oem.conversion.ConversionSetup -d
%{JDBC_CONN_STRING}% -p %{CIS_ADMIN_USR}%,%{CIS_ADMIN_PSWD}% -a -s
%{STG_ADMIN_USR}%,%{STG_ADMIN_PSWD}%,%{STG_RW_USER}%
```

ADB:

```
"%JAVA_HOME%"\bin\java
com.oracle.ouaf.oem.conversion.ConversionSetup -d
jdbc:oracle:thin:@<tnsname>?TNS_ADMIN=<wallet location> -p
%{CIS_ADMIN_USR}%,%{CIS_ADMIN_PSWD}% -a -s
%{STG_ADMIN_USR}%,%{STG_ADMIN_PSWD}%,%{STG_RW_USER}%
```

If '-r' (revert staging flag) is passed, it connects to the database as staging schema admin and reverts the staging to a state prior to conversion setup, running the same SQLs present in drop_* or restore_* SQL files.

On-premises:

```
"%JAVA_HOME%"\bin\java
com.oracle.ouaf.oem.conversion.ConversionSetup -d
%{JDBC_CONN_STRING}% -p %{CIS_ADMIN_USR}%,%{CIS_ADMIN_PSWD}% -r -s
%{STG_ADMIN_USR}%,%{STG_ADMIN_PSWD}%,%{STG_RW_USER}%
```

ADB:

```
"%JAVA_HOME%"\bin\java
com.oracle.ouaf.oem.conversion.ConversionSetup -d
jdbc:oracle:thin:@<tnsname>?TNS_ADMIN=<wallet location> -p
%{CIS_ADMIN_USR}%,%{CIS_ADMIN_PSWD}% -r -s
%{STG_ADMIN_USR}%,%{STG_ADMIN_PSWD}%,%{STG_RW_USER}%
```

To generate ONLY the scripts, the program should be invoked with '-s' parameter with no other flags (no '-a' or '-r'):

On-premises:

```
"%JAVA_HOME%"\bin\java
com.oracle.ouaf.oem.conversion.ConversionSetup -d
%{JDBC_CONN_STRING}% -p %{CIS_ADMIN_USR}%,%{CIS_ADMIN_PSWD}% -s
%{STG_ADMIN_USR}%,%{STG_ADMIN_PSWD}%,%{STG_RW_USER}%
```

ADB:

```
"%JAVA_HOME%"\bin\java
com.oracle.ouaf.oem.conversion.ConversionSetup -d
jdbc:oracle:thin:@<tnsname>?TNS_ADMIN=<wallet location> -p
%{CIS_ADMIN_USR}%,%{CIS_ADMIN_PSWD}% -s
%{STG_ADMIN_USR}%,%{STG_ADMIN_PSWD}%,%{STG_RW_USER}%
```

It creates the following files in current directory (or output directory if passed as '-o') if '-r' (revert staging flag) is not passed.

Filename	Description
mainFile.sql	Contains instructions on the order and how to run the files
create_schgrants.sql	Grant read permissions on production schema objects to staging admin user
create_cxviews.sql	Creates Conversion X views on staging schema
create_ctlviews.sql	Create Production Control views on staging schema
createck_tbls.sql	Creates Conversion Key tables on staging schema
createck_pkix.sql	Creates primary indexes on Conversion key tables
createck_secix.sql	Creates secondary indexes on Conversion key tables
createcr_tbls.sql	Creates XML/CLOB Resolution tables on staging schema
create_stggrants.sql	Grants select, insert permissions on above created staging objects to staging read/write user
restore_ctltbls.sql	Restores Control tables
drop_cxviews.sql	Restored Conversion X views
drop_tables.sql	Restores CK & CR tables

After the staging schema has been set up, generate the security for the staging user following the steps in the Configuring Security section.

Linux/UNIX:

- On-premises:

```
java -Xmx1500M com.oracle.ouaf.oem.install.OraGenSec-d
<STAGING_DBUSER>,< STAGING_DBPASS>,
jdbc:oracle:thin:@<DB_SERVER>:<PORT>/
<SID> -u <STAGING_RW_USER> -r <STAGING_RW_USER_ROLE>,<
STAGING_R_USER_ROLE> -a A -p
<RW_USERPASS>
```

- ADB:

```
java -Xmx1500M com.oracle.ouaf.oem.install.OraGenSec-d
<STAGING_DBUSER>,< STAGING_DBPASS>,
jdbc:oracle:thin:@<tnsname>?TNS_ADMIN=<wallet location> -u
<STAGING_RW_USER> -r <STAGING_RW_USER_ROLE>,<
STAGING_R_USER_ROLE> -a A -p <RW_USERPASS>
```

Windows:

- On-premises:

```
"%JAVA_HOME%"\bin\java -Xmx1500M
com.oracle.ouaf.oem.install.OraGenSec -d < STAGING_DBUSER>,<
STAGING DBPASS>, jdbc:oracle:thin:@<DB_SERVER>:<PORT>/
<SID> -u < STAGING RW_USER> -r <STAGING RW_USER_ROLE>,<STAGING
R_USER_ROLE> -a A -p
<STAGING_RW_USERPASS>
```

- ADB:

```
"%JAVA_HOME%"\bin\java -Xmx1500M
com.oracle.ouaf.oem.install.OraGenSec -d < STAGING_DBUSER>,<
STAGING DBPASS>, jdbc:oracle:thin:@<tnsname>?TNS_ADMIN=<wallet
location>  -u < STAGING RW_USER> -r <STAGING
RW_USER_ROLE>,<STAGING R_USER_ROLE> -a A -p
<STAGING_RW_USERPASS>
```

Upgrading the Production Database

For instructions to update the Production schema (example: CISADM), refer to the Upgrade Install section in Installing the Database.

Upgrading the Staging Database

This section describes how to upgrade the Staging schema on a Conversion environment in V2.4.0.0.0 and higher.

Restore the Staging schema before upgrading/updating an existing environment:

> **Note**: Execute this step if you are upgrading/updating an already-setup-for-Conversion environment.

1. Extract ConvSetup.tar from the ..\WAM\Conversion\ folder.

   ```
   tar xvf ConvSetup.tar
   ```

2. On the Staging schema, execute the following script:

   ```
   script restore_ctltbls.sql
   ```

   > **Note**: This script would have been generated during the conversion setup in previous versions starting and is also a part of ConvSetup.tar.

3. Upgrade the staging schema (example: STGADM). For instructions, refer to the Upgrade Install section in Installing the Database.
4. To set up conversion again, follow the instructions in the Preparing the Staging Database section.

Appendix A

Upgrades to the Oracle Utilities Work and Asset Management 25.4 Database

This chapter provides a list of database changes in the following:

- Oracle Utilities Work and Asset Management V2.3.0.0.0 to V2.4.0.1.1
- Oracle Utilities Work and Asset Management V2.4.0.0.0 to V2.4.0.1.1
- Oracle Utilities Work and Asset Management V2.4.0.1.1 to V25.4

Oracle Utilities Work and Asset Management V2.3.0.0.0 to V2.4.0.1.1

New Tables

Table_Name
W1_ACTIVITY_MEASUREMENT_TYPE
W1_AST_DERMS_EVTCTL_CSTHIST
W1_CU_CU_USAGE
W1_DESIGN_ELEMENT_CHKLST_TYP
W1_DESIGN_ELEMENT_DEP
W1_EU_TIMESHEET
W1_PROJECT_CATEGORY_LOG
W1_PROJECT_CATEGORY_LOG_PARM
W1_TMPL_WO_DOCUMENT
W1_VENDOR_LOC_PAYTO_LOC
W1_WO_DOCUMENT

New Fields

Table_Name	Column_Name	Data Type	Data Length
W1_ACTIVITY_MEASUREMENT_TYPE	ACT_ID	CHAR	14
W1_ACTIVITY_MEASUREMENT_TYPE	MEASUREMENT_TYPE_CD	VARCHAR2	30
W1_ACTIVITY_MEASUREMENT_TYPE	VERSION	NUMBER	5
W1_AST_DERMS_EVTCTL_CSTHIST	ASSET_ID	CHAR	12
W1_AST_DERMS_EVTCTL_CSTHIST	CST_SCORE	NUMBER	16
W1_AST_DERMS_EVTCTL_CSTHIST	VERSION	NUMBER	5
W1_AST_DERMS_EVTCTL_CSTHIST	W1_DYN_OPT_EVENT_ID	CHAR	12
W1_AST_DERMS_EVTCTL_CSTHIST	ASSET_AVAILABILITY_FLG	CHAR	1

Table_Name	Column_Name	Data Type	Data Length
W1_ W1_CU_CU_USAGE	CU_ID	CHAR	12
W1_ W1_CU_CU_USAGE	CU_USAGE_CD	VARCHAR2	30
W1_ W1_CU_CU_USAGE	VERSION	NUMBER	5
W1_DESIGN_ELEMENT_CHKLST_TYP	DESIGN_ELEMENT_ID	CHAR	12
W1_DESIGN_ELEMENT_CHKLST_TYP	CHECKLIST_TYPE_CD	VARCHAR2	30
W1_DESIGN_ELEMENT_CHKLST_TYP	VERSION	NUMBER	5
W1_DESIGN_ELEMENT_DEP	DESIGN_ELEMENT_ID	CHAR	12
W1_DESIGN_ELEMENT_DEP	PRED_DESIGN_ELEMENT_ID	CHAR	12
W1_DESIGN_ELEMENT_DEP	TIMING_OPTION_FLG	CHAR	4
W1_DESIGN_ELEMENT_DEP	W1_MIN_OFFSET	NUMBER	6
W1_DESIGN_ELEMENT_DEP	W1_MAX_OFFSET	NUMBER	6
W1_DESIGN_ELEMENT_DEP	RESOURCE_OPTION_FLG	CHAR	4
W1_DESIGN_ELEMENT_DEP	VERSION	NUMBER	5
W1_EU_TIMESHEET	EU_ID	CHAR	12
W1_EU_TIMESHEET	EU_DT	DATE	7
W1_EU_TIMESHEET	HOURS	NUMBER	13
W1_EU_TIMESHEET	VERSION	NUMBER	5
W1_PROJECT_CATEGORY_LOG	PRJ_CAT_CD	VARCHAR2	120
W1_PROJECT_CATEGORY_LOG	SEQNO	NUMBER	22
W1_PROJECT_CATEGORY_LOG	BO_STATUS_CD	CHAR	48
W1_PROJECT_CATEGORY_LOG	BO_STATUS_REASON_CD	VARCHAR2	120
W1_PROJECT_CATEGORY_LOG	ADHOC_CHAR_VAL	VARCHAR2	1016
W1_PROJECT_CATEGORY_LOG	CHAR_TYPE_CD	CHAR	32
W1_PROJECT_CATEGORY_LOG	CHAR_VAL	CHAR	64
W1_PROJECT_CATEGORY_LOG	CHAR_VAL_FK1	VARCHAR2	200
W1_PROJECT_CATEGORY_LOG	CHAR_VAL_FK2	VARCHAR2	200
W1_PROJECT_CATEGORY_LOG	CHAR_VAL_FK3	VARCHAR2	200

Table_Name	Column_Name	Data Type	Data Length
W1_PROJECT_CATEGORY_LOG	CHAR_VAL_FK4	VARCHAR2	200
W1_PROJECT_CATEGORY_LOG	CHAR_VAL_FK5	VARCHAR2	200
W1_PROJECT_CATEGORY_LOG	DESCRLONG	VARCHAR2	16000
W1_PROJECT_CATEGORY_LOG	LOG_DTTM	DATE	7
W1_PROJECT_CATEGORY_LOG	LOG_ENTRY_TYPE_FLG	CHAR	16
W1_PROJECT_CATEGORY_LOG	MESSAGE_CAT_NBR	NUMBER	22
W1_PROJECT_CATEGORY_LOG	MESSAGE_NBR	NUMBER	22
W1_PROJECT_CATEGORY_LOG	USER_ID	CHAR	32
W1_PROJECT_CATEGORY_LOG	VERSION	NUMBER	22
W1_PROJECT_CATEGORY_LOG_PARM	PRJ_CAT_CD	VARCHAR2	120
W1_PROJECT_CATEGORY_LOG_PARM	SEQNO	NUMBER	22
W1_PROJECT_CATEGORY_LOG_PARM	PARM_SEQ	NUMBER	22
W1_PROJECT_CATEGORY_LOG_PARM	MSG_PARM_TYP_FLG	CHAR	16
W1_PROJECT_CATEGORY_LOG_PARM	F1_MSG_PARM_VLONG	VARCHAR2	8000
W1_PROJECT_CATEGORY_LOG_PARM	VERSION	NUMBER	22
W1_TMPL_WO_DOCUMENT	TMPL_WO_ID	CHAR	48
W1_TMPL_WO_DOCUMENT	ATTACHMENT_ID	CHAR	56
W1_TMPL_WO_DOCUMENT	DOCUMENT_ID	CHAR	48
W1_TMPL_WO_DOCUMENT	CRE_DTTM	DATE	7
W1_TMPL_WO_DOCUMENT	USER_ID	CHAR	32
W1_TMPL_WO_DOCUMENT	VERSION	NUMBER	22
W1_VENDOR_LOC_PAYTO_LOC	VENDOR_LOC_ID	CHAR	48
W1_VENDOR_LOC_PAYTO_LOC	PAY_TO_LOC	CHAR	48
W1_VENDOR_LOC_PAYTO_LOC	VERSION	NUMBER	22
W1_WO_DOCUMENT	WO_ID	CHAR	56
W1_WO_DOCUMENT	ATTACHMENT_ID	CHAR	56
W1_WO_DOCUMENT	DOCUMENT_ID	CHAR	48
W1_WO_DOCUMENT	CRE_DTTM	DATE	7

Table_Name	Column_Name	Data Type	Data Length
W1_WO_DOCUMENT	USER_ID	CHAR	32
W1_WO_DOCUMENT	VERSION	NUMBER	22

Updated Fields

Table_Name	Column_Name	Data Type	Data Length
W1_BC_LINE	PI_RATIO	NUMBER (3)	NUMBER (8)
W1_WORK_LOC	CITY	VARCHAR2 (90)	VARCHAR2 (30)
W1_WORK_LOC	W1_CITY_UPR	VARCHAR2 (90)	VARCHAR2 (30)
W1_WORK_REQ	CITY	VARCHAR2 (90)	VARCHAR2 (30)
W1_WORK_REQ	W1_CITY_UPR	VARCHAR2 (90)	VARCHAR2 (30)

New Indexes

Table_Name	Index_Name
W1_ACTIVITY	W1T244S18
W1_ACTIVITY	W1T244S19
W1_ACTIVITY	W1T244S20
W1_ACTIVITY_MEASUREMENT_TYPE	W1T962P0
W1_ACTIVITY_TYPE	W1C237S3
W1_ACT_CONSTR_LOC	W1T780S1
W1_APPROVAL_PROF	W1C556S1
W1_AST_DERMS_EVTCTL_CSTHIST	W1M967P0
W1_COMPLIANCE_TYPE_CHAR	W1C845S1
W1_CU	W1M746S3
W1_CU_CU_USAGE	W1M968P0
W1_CU_SET	W1C757S1
W1_DESIGN_ELEMENT_CHAR	W1M799S1
W1_DESIGN_ELEMENT_CHKLST_TYP	W1T966P0
W1_DESIGN_ELEMENT_DEP	W1T965P0
W1_DESIGN_ELEMENT_LOG	W1M801S1

Table_Name	Index_Name
W1_DESIGN_ELEMENT_LOG	W1M801S2
W1_EMPLOYEE	W1M500S5
W1_EMPLOYEE	W1M500S6
W1_EQUIP_GROUP	W1C768S1
W1_EU_TIMESHEET	W1M961P0
W1_FT	W1T312S1
W1_INVOICE_HEADER	W1T500S2
W1_INVOICE_LINE	W1T509S3
W1_INVOICE_LINE	W1T509S4
W1_INVOICE_LINE	W1T509S5
W1_MAINT_TRIGGER	W1M658S3
W1_NODE	W1M205S10
W1_NODE	W1M205S11
W1_NODE_TYPE	W1C200S1
W1_PI_CNT_MST	W1T324S3
W1_PO_HEADER	W1T410S5
W1_PO_LINE	W1T418S5
W1_PO_LINE	W1T418S6
W1_PO_LINE	W1T418S7
W1_PO_LINE	W1T418S8
W1_PROJECT_CATEGORY_LOG	W1C963P0
W1_PROJECT_CATEGORY_LOG_PARM	W1C964P0
W1_PR_HDR	W1T434S6
W1_PR_LINE	W1T443S3
W1_RCPT_HDR	W1T460S3
W1_RCPT_LINE	W1T468S3
W1_RR	W1T405S2
W1_RTN_LINE	W1T483S2
W1_RTN_LINE	W1T483S3
W1_RTN_LINE	W1T483S4
W1_RTN_LINE	W1T483S5
W1_RTN_LINE	W1T483S6
W1_STOCK_ITEM_DTL	W1M568S4

Table_Name	Index_Name
W1_STOCK_ITEM_DTL	W1M568S5
W1_TIMEKEEPER	W1C948S1
W1_TIMESHEET_DETAIL	W1M595S3
W1_TMPL_ACT	W1M617S1
W1_TMPL_WO_DOCUMENT	W1T970P0
W1_VENDOR_LOC_PAYTO_LOC	W1M955P0
W1_WO	W1T232S13
W1_WO	W1T232S14
W1_WORK_DESIGN_CHAR	W1M790S1
W1_WORK_DESIGN_LOG	W1M792S1
W1_WORK_DESIGN_LOG	W1M792S2
W1_WORK_REQ	W1T298S8
W1_WO_DOCUMENT	W1T969P0

Dropped Indexes

Table_Name	Index_Name
W1_RTN_LINE	W1M637S2
W1_RTN_LINE	W1M637S3
W1_STOCK_ITEM_DTL	W1M601S13

Oracle Utilities Work and Asset Management V2.4.0.0.0 to V2.4.0.1.1

New Tables

Table_Name
W1_AST_DERMS_EVTCTL_CSTHIST
W1_CU_CU_USAGE
W1_DESIGN_ELEMENT_CHKLST_TYP
W1_DESIGN_ELEMENT_DEP
W1_TMPL_WO_DOCUMENT
W1_WO_DOCUMENT

New Fields

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

Table_Name	Column_Name	Data Type	Data Length
W1_AST_DERMS_EVTCTL_CSTHIST	ASSET_ID	CHAR	12
W1_AST_DERMS_EVTCTL_CSTHIST	CST_SCORE	NUMBER	16
W1_AST_DERMS_EVTCTL_CSTHIST	VERSION	NUMBER	5
W1_AST_DERMS_EVTCTL_CSTHIST	W1_DYN_OPT_EVENT_ID	CHAR	12
W1_AST_DERMS_EVTCTL_CSTHIST	ASSET_AVAILABILITY_FLG	CHAR	1
W1_ W1_CU_CU_USAGE	CU_ID	CHAR	12
W1_ W1_CU_CU_USAGE	CU_USAGE_CD	VARCHAR2	30
W1_ W1_CU_CU_USAGE	VERSION	NUMBER	5
W1_DESIGN_ELEMENT_CHKLST_TYP	DESIGN_ELEMENT_ID	CHAR	12
W1_DESIGN_ELEMENT_CHKLST_TYP	CHECKLIST_TYPE_CD	VARCHAR2	30
W1_DESIGN_ELEMENT_CHKLST_TYP	VERSION	NUMBER	5

Table_Name	Column_Name	Data Type	Data Length
W1_DESIGN_ELEMENT_DEP	DESIGN_ELEMENT_ID	CHAR	12
W1_DESIGN_ELEMENT_DEP	PRED_DESIGN_ELEMENT_ID	CHAR	12
W1_DESIGN_ELEMENT_DEP	TIMING_OPTION_FLG	CHAR	4
W1_DESIGN_ELEMENT_DEP	W1_MIN_OFFSET	NUMBER	6
W1_DESIGN_ELEMENT_DEP	W1_MAX_OFFSET	NUMBER	6
W1_DESIGN_ELEMENT_DEP	RESOURCE_OPTION_FLG	CHAR	4
W1_DESIGN_ELEMENT_DEP	VERSION	NUMBER	5
W1_TMPL_WO_DOCUMENT	TMPL_WO_ID	CHAR	48
W1_TMPL_WO_DOCUMENT	ATTACHMENT_ID	CHAR	56
W1_TMPL_WO_DOCUMENT	DOCUMENT_ID	CHAR	48
W1_TMPL_WO_DOCUMENT	CRE_DTTM	DATE	7
W1_TMPL_WO_DOCUMENT	USER_ID	CHAR	32
W1_TMPL_WO_DOCUMENT	VERSION	NUMBER	22
W1_WO_DOCUMENT	WO_ID	CHAR	56
W1_WO_DOCUMENT	ATTACHMENT_ID	CHAR	56
W1_WO_DOCUMENT	DOCUMENT_ID	CHAR	48
W1_WO_DOCUMENT	CRE_DTTM	DATE	7
W1_WO_DOCUMENT	USER_ID	CHAR	32
W1_WO_DOCUMENT	VERSION	NUMBER	22

Updated Fields

Table_Name	Column_Name	Data Type	Data Length
W1_BC_LINE	PI_RATIO	NUMBER (3)	NUMBER (8)
W1_WORK_LOC	CITY	VARCHAR2 (90)	VARCHAR2 (30)
W1_WORK_LOC	W1_CITY_UPR	VARCHAR2 (90)	VARCHAR2 (30)

Table_Name	Column_Name	Data Type	Data Length
W1_WORK_REQ	CITY	VARCHAR2 (90)	VARCHAR2 (30)
W1_WORK_REQ	W1_CITY_UPR	VARCHAR2 (90)	VARCHAR2 (30)

New Indexes

Table_Name	Index_Name
W1_ACT_CONSTR_LOC	W1T780S1
W1_APPROVAL_PROF	W1C556S1
W1_AST_DERMS_EVTCTL_CSTHIST	W1M967P0
W1_COMPLIANCE_TYPE_CHAR	W1C845S1
W1_CU	W1M746S3
W1_CU_CU_USAGE	W1M968P0
W1_CU_SET	W1C757S1
W1_DESIGN_ELEMENT_CHAR	W1M799S1
W1_DESIGN_ELEMENT_CHKLST_TYP	W1T966P0
W1_DESIGN_ELEMENT_DEP	W1T965P0
W1_DESIGN_ELEMENT_LOG	W1M801S1
W1_DESIGN_ELEMENT_LOG	W1M801S2
W1_EMPLOYEE	W1M500S5
W1_EMPLOYEE	W1M500S6
W1_EQUIP_GROUP	W1C768S1
W1_EU_TIMESHEET	W1M961P0
W1_FT	W1T312S1
W1_PR_HDR	W1T434S6
W1_PR_LINE	W1T443S3
W1_TIMEKEEPER	W1C948S1
W1_TMPL_WO_DOCUMENT	W1T970P0
W1_WORK_DESIGN_CHAR	W1M790S1
W1_WORK_DESIGN_LOG	W1M792S1
W1_WORK_DESIGN_LOG	W1M792S2

Table_Name	Index_Name
W1_WO_DOCUMENT	W1T969P0

Dropped Indexes

Table_Name	Index_Name
W1_RTN_LINE	W1M637S2
W1_RTN_LINE	W1M637S3
W1_STOCK_ITEM_DTL	W1M601S13

Oracle Utilities Work and Asset Management V2.4.0.1.1 to V25.4

New Tables

Table	Name	Description
W1_DESIGN_ELEMENT_PLAN_SH_TYPE	Design Element Planned Service History	Used to capture planned service histories for template construction work activities
W1_ASSET_TYPE_SPEC_ATTRIBUTE	Asset Type Specification Attribute	Used to capture attributes for asset type specifications.
W1_NODE_ATTRIBUTE	Location Attribute	Used to capture attributes for locations.
W1_NODE_TYPE_ATTRIBUTE	Location Type Attributes	Used to capture attributes for location types.
W1_SPECIFICATION_ATTRIBUTE	Specification Attribute	Used to capture attributes for specifications.
W1_BC_LINE_NOTE	Blanket Contract Line Note	Used to capture notes for blanket contract lines

New Columns

Table_Name	Column_Name	Data Type	Data Length
W1_ACTIVITY	PERCENT_COMPLETE	NUMBER	22
W1_ACT_RECON_COST_CENTER	W1_AMOUNT	NUMBER	22
W1_ACT_RECON_PROP_UNIT	COST_CENTER_CD	VARCHAR2	120
W1_ACT_RECON_PROP_UNIT	CU_USAGE_CD	VARCHAR2	120
W1_ASSET	ADDITIONAL_COST	NUMBER	22
W1_ASSET	ERP_PRJ_NUM	VARCHAR2	240
W1_ASSET	ERP_SUB_PRJ_NUM	VARCHAR2	240
W1_ASSET	MEMBER_ASSET_FLG	CHAR	16
W1_ASSET_TYPE	ASSET_NO_CAL_PREFIX_FLG	CHAR	16
W1_ASSET_TYPE	ASSET_NO_GEN_FLG	CHAR	16
W1_ASSET_TYPE	ASSET_NO_PREFIX	CHAR	16
W1_ASSET_TYPE	ASSET_NO_SEQ_LEN	NUMBER	22
W1_ASSET_TYPE	BADGE_NO_CAL_PREFIX_FLG	CHAR	16
W1_ASSET_TYPE	BADGE_NO_GEN_FLG	CHAR	16
W1_ASSET_TYPE	BADGE_NO_PREFIX	CHAR	16
W1_ASSET_TYPE	BADGE_NO_SEQ_LEN	NUMBER	22

Table_Name	Column_Name	Data Type	Data Length
W1_ASSET_TYPE_ATTRIBUTE	ATT_SEQ_NUM	NUMBER	22
W1_ASSET_TYPE_SPEC_ATTRIBUTE	ASSET_TYPE_CD	VARCHAR2	120
W1_ASSET_TYPE_SPEC_ATTRIBUTE	ATT_SEQ_NUM	NUMBER	22
W1_ASSET_TYPE_SPEC_ATTRIBUTE	CHAR_TYPE_CD	CHAR	32
W1_ASSET_TYPE_SPEC_ATTRIBUTE	VERSION	NUMBER	22
W1_BC_LINE_NOTE	BC_LINE_ID	CHAR	56
W1_BC_LINE_NOTE	NOTE_DTTM	DATE	7
W1_BC_LINE_NOTE	NOTE_TYPE_FLG	CHAR	16
W1_BC_LINE_NOTE	NOTE_VALUE	VARCHAR2	16000
W1_BC_LINE_NOTE	SEQNO	NUMBER	22
W1_BC_LINE_NOTE	USER_ID	CHAR	32
W1_BC_LINE_NOTE	VERSION	NUMBER	22
W1_CMPL_EVT	CONSTR_LOC_ID	CHAR	56
W1_DESIGN_ELEMENT	ACT_DURATION	NUMBER	22
W1_DESIGN_ELEMENT	BACK_LOG_GRP_FLG	CHAR	16
W1_DESIGN_ELEMENT	UOT_FLG	CHAR	16
W1_DESIGN_ELEMENT_PLAN_SH_TYPE	DESIGN_ELEMENT_ID	CHAR	48
W1_DESIGN_ELEMENT_PLAN_SH_TYPE	SVC_HIST_TYPE_CD	VARCHAR2	120
W1_DESIGN_ELEMENT_PLAN_SH_TYPE	SVC_HIST_TYPE_REQ_FLG	CHAR	16
W1_DESIGN_ELEMENT_PLAN_SH_TYPE	VERSION	NUMBER	22
W1_NODE	DESCRLONG	VARCHAR2	16000
W1_NODE_ATTRIBUTE	ADHOC_CHAR_VAL	VARCHAR2	1016
W1_NODE_ATTRIBUTE	CHAR_TYPE_CD	CHAR	32
W1_NODE_ATTRIBUTE	CHAR_VAL	CHAR	64
W1_NODE_ATTRIBUTE	CHAR_VAL_FK1	VARCHAR2	200
W1_NODE_ATTRIBUTE	CHAR_VAL_FK2	VARCHAR2	200
W1_NODE_ATTRIBUTE	CHAR_VAL_FK3	VARCHAR2	200
W1_NODE_ATTRIBUTE	CHAR_VAL_FK4	VARCHAR2	200
W1_NODE_ATTRIBUTE	CHAR_VAL_FK5	VARCHAR2	200
W1_NODE_ATTRIBUTE	NODE_ID	CHAR	48

Table_Name	Column_Name	Data Type	Data Length
W1_NODE_ATTRIBUTE	SEQ_NUM	NUMBER	22
W1_NODE_ATTRIBUTE	SRCH_CHAR_VAL	VARCHAR2	200
W1_NODE_ATTRIBUTE	VERSION	NUMBER	22
W1_NODE_TYPE_ATTRIBUTE	CHAR_TYPE_CD	CHAR	32
W1_NODE_TYPE_ATTRIBUTE	NODE_TYPE_CD	VARCHAR2	120
W1_NODE_TYPE_ATTRIBUTE	VERSION	NUMBER	22
W1_SPECIFICATION_ATTRIBUTE	ADHOC_CHAR_VAL	VARCHAR2	1016
W1_SPECIFICATION_ATTRIBUTE	CHAR_TYPE_CD	CHAR	32
W1_SPECIFICATION_ATTRIBUTE	CHAR_VAL	CHAR	64
W1_SPECIFICATION_ATTRIBUTE	CHAR_VAL_FK1	VARCHAR2	200
W1_SPECIFICATION_ATTRIBUTE	CHAR_VAL_FK2	VARCHAR2	200
W1_SPECIFICATION_ATTRIBUTE	CHAR_VAL_FK3	VARCHAR2	200
W1_SPECIFICATION_ATTRIBUTE	CHAR_VAL_FK4	VARCHAR2	200
W1_SPECIFICATION_ATTRIBUTE	CHAR_VAL_FK5	VARCHAR2	200
W1_SPECIFICATION_ATTRIBUTE	SEQ_NUM	NUMBER	22
W1_SPECIFICATION_ATTRIBUTE	SPECIFICATION_CD	VARCHAR2	120
W1_SPECIFICATION_ATTRIBUTE	SRCH_CHAR_VAL	VARCHAR2	200
W1_SPECIFICATION_ATTRIBUTE	VERSION	NUMBER	22
W1_TIMEKEEPER	OWNING_ACCESS_GRP_CD	CHAR	48

New Indexes

Table_Name	Index_Name	Column_Name
W1_ACTIVITY	W1T244S21	SVC_HIST_ID
W1_ASSET_TYPE_SPEC_ATTRIBUTE	W1C972P0	ASSET_TYPE_CD
W1_ASSET_TYPE_SPEC_ATTRIBUTE	W1C972P0	CHAR_TYPE_CD
W1_BC_LINE_NOTE	W1T976P0	BC_LINE_ID
W1_BC_LINE_NOTE	W1T976P0	SEQNO
W1_CREW_SHIFT_ACT_SCHED	W1T311S1	ACT_RESRC_REQMT_ID
W1_CREW_SHIFT_ACT_SCHED	W1T311S1	CREW_SHIFT_ID
W1_DESIGN_ELEMENT_PLAN_SH_TYPE	W1T971P0	DESIGN_ELEMENT_ID

Table_Name	Index_Name	Column_Name
W1_DESIGN_ELEMENT_PLAN_SH_TYPE	W1T971P0	SVC_HIST_TYPE_CD
W1_DOCUMENT	W1M577S2	DOCUMENT_CLASS_FLG
W1_MAT_RET_LINE	W1T400S2	STOCK_ITEM_DTL_ID
W1_MEASUREMENT	W1T273S5	ASSET_ID
W1_NODE_ATTRIBUTE	W1M975P0	CHAR_TYPE_CD
W1_NODE_ATTRIBUTE	W1M975P0	NODE_ID
W1_NODE_ATTRIBUTE	W1M975P0	SEQ_NUM
W1_NODE_ATTRIBUTE	W1M975S1	SRCH_CHAR_VAL
W1_NODE_ATTRIBUTE	W1M975S2	CHAR_TYPE_CD
W1_NODE_ATTRIBUTE	W1M975S3	CHAR_TYPE_CD
W1_NODE_ATTRIBUTE	W1M975S3	CHAR_VAL
W1_NODE_ATTRIBUTE	W1M975S4	ADHOC_CHAR_VAL
W1_NODE_ATTRIBUTE	W1M975S4	CHAR_TYPE_CD
W1_NODE_TYPE_ATTRIBUTE	W1C974P0	CHAR_TYPE_CD
W1_NODE_TYPE_ATTRIBUTE	W1C974P0	NODE_TYPE_CD
W1_ODC_DTL	W1T340S3	ACT_RESRC_REQMT_ID
W1_PO_LINE	W1T418S9	ACT_RESRC_REQMT_ID
W1_RESRC_TYPE	W1M555S2	W1_RESRC_CLASS_FLG
W1_SPECIFICATION_ATTRIBUTE	W1C973P0	CHAR_TYPE_CD
W1_SPECIFICATION_ATTRIBUTE	W1C973P0	SEQ_NUM
W1_SPECIFICATION_ATTRIBUTE	W1C973P0	SPECIFICATION_CD
W1_SPECIFICATION_ATTRIBUTE	W1C973S1	SRCH_CHAR_VAL
W1_SPECIFICATION_ATTRIBUTE	W1C973S2	CHAR_TYPE_CD
W1_SPECIFICATION_ATTRIBUTE	W1C973S3	CHAR_TYPE_CD
W1_SPECIFICATION_ATTRIBUTE	W1C973S3	CHAR_VAL
W1_SPECIFICATION_ATTRIBUTE	W1C973S4	ADHOC_CHAR_VAL
W1_SPECIFICATION_ATTRIBUTE	W1C973S4	CHAR_TYPE_CD
W1_TIMESHEET_DETAIL	W1M595S4	ACT_RESRC_REQMT_ID

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

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

Table	Description	Type of Table
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 v4.3.0.6.0

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
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

Table	Description	Type of Table
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

Table	Description	Table Type
F1_MKTPROC	Market Process	Transaction
F1_MKTPROC_LOG	Market Process Log	Transaction
F1_MKTPROC_LOG_PARM	Market Process Log Parameter	Transaction
F1_MKTPROC_CHAR	Market Process Characteristics	Transaction
F1_MKTPROC_ID	Market Process Identifiers	Transaction
F1_MKTPROC_REL_OBJ	Market Process Related Objects	Transaction
F1_MKTPROC_REL_PROC	Related Market Processes	Transaction
F1_MKTMSG_IN	Inbound Market Message	Transaction
F1_MKTMSG_IN_LOG	Inbound Market Message Log	Transaction
F1_MKTMSG_IN_LOG_PARM	Inbound Market Message Log Parameter	Transaction
F1_MKTMSG_IN_CHAR	Inbound Market Message Characteristics	Transaction
F1_MKTMSG_IN_DATA	Inbound Market Message Data	Transaction
F1_MKTMSG_IN_EXT_REF	Inbound Market Message References	Transaction
F1_MKTMSG_IN_ID	Inbound Market Message Identifiers	Transaction
F1_MKTMSG_IN_REL_OBJ	Inbound Market Message Related Objects	Transaction
F1_MKTCFG	Market Configuration	Admin
F1_MKTCFG_L	Market Configuration Language	Admin(non system)
F1_MKTCFG _CHAR	Market Configuration Characteristics	Master
F1_MKTMSG_TYPE	Market Message Type	Admin(non system)

Table	Description	Table Type
F1_MKTMSG_TYPE_L	Market Message Type	Admin (non system)
F1_MKTMSG_TYPE_CHAR	Market Message Type Characteristics	Admin (non system)
F1_MKTPROC_EVT	Market Process Event	Transaction
F1_MKTPROC_EVT _LOG	Market Process Event Log	Transaction
F1_MKTPROC_EVT _LOG_PARM	Market Process Event Log Parameter	Transaction
F1_MKTPROC_EVT _CHAR	Market Process Event Characteristics	Transaction
F1_MKTPROC_EVT _REL_OBJ	Market Process Event Related Objects	Transaction
F1_MKTMSG_OUT	Outbound Market Message	Transaction
F1_MKTMSG_OUT_LOG	Outbound Market Message Log	Transaction
F1_MKTMSG_OUT_LOG_PARM	Outbound Market Message Log Parameter	Transaction
F1_MKTMSG_OUT_CHAR	Outbound Market Message Characteristics	Transaction
F1_MKTMSG_OUT_DATA	Outbound Market Message Data	Transaction
F1_MKTMSG_OUT_EXT_REF	Outbound Market Message References	Transaction
F1_MKTMSG_OUT_ID	Outbound Market Message Identifiers	Transaction
F1_MKTMSG_OUT_REL_OBJ	Outbound Market Message Related Objects	Transaction
F1_MKTPROC_TYPE	Market Process Type	Admin(non system)
F1_MKTPROC_TYPE_L	Market Process Type Language	Admin(non system)
F1_MKTPROC_TYPE_CHAR	Market Process Type Characteristics	Admin(non system)
F1_MKTPROC_TYPE_EVT	Market Process Type Events	Admin(non system)
F1_CALENDAR_D	Calendar Dimension	Transaction
F1_TIME_D	Time Dimension	Transaction
F1_IWS_SVC_OPER_PARM	Inbound Web Service Operations Parameter	Admin

New Views
None

Dropped Tables
None

Unsupported Tables
None

Added Columns

Table	Description	Table Type
CI_BATCH_THD	THD_RETRY_CNT	N
F1_IWS_SVC	URI_COMPONENT	N

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)

CREATE INDEX XT026S5 ON CI_NT_DWN (SPR_CD,CRE_DTTM)

PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS STORAGE(INITIAL 262144 NEXT 262144 MINEXTENTS 1 MAXEXTENTS 2147483645 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1

BUFFER_POOL DEFAULT FLASH_CACHE DEFAULT CELL_FLASH_CACHE DEFAULT) TABLESPACE "CISTS_01"

Upgrades to Oracle Utilities Application Framework v4.4.0.3.0

New Tables

Table	Description	Table Type
F1_INSIGHT_TYPE	Insight Type	Admin System
F1_INSIGHT_TYPE_CHAR	Insight Type Characteristic	Admin System
F1_INSIGHT_TYPE_L	Insight Type Language	Admin System
F1_INSIGHT_TYPE_ALG	Insight Type Algorithm	Admin System
F1_INSIGHT_TYPE_USER_ACT	Insight Type User Action	Admin System
F1_INSIGHT_TYPE_USER_ACT_ALG	Insight Type User Action Algorithm	Admin System
F1_INSIGHT_TYPE_VISUAL_SETTING	Insight Type Visual Setting	Admin System
F1_COLOR_OPT	Color Option	Admin System
F1_COLOR_OPT_L	Color Option Language	Admin System
F1_COLOR_OPT_ITEM	Color Option Item	Admin System
F1_TREE	Tree	Admin System
F1_TREE_L	Tree Language	Admin System
F1_TREE_NODE	Tree Node	Admin System
F1_TREE_NODE_L	Tree Node Language	Admin System
F1_TREE_NODE_ALG	Tree Node Algorithms	Admin System
F1_INSIGHT_GRP	Insight Group	Admin
F1_INSIGHT_GRP_L	Insight Group Language	Admin
F1_INSIGHT_GRP_TYPE	Insight Group Insight Type	Admin
F1_INSIGHT_GRP_VISUAL_STR	Insight Group Valid Visual Structure	Admin
F1_INSIGHT_GRP_CHAR	Insight Group Characteristics	Admin

New Views

None

Dropped Tables

None

Unsupported Tables

None

Added Columns

Table	Field	Required
SC_USER	DASHBOARD_LOC_FLG	Yes
SC_USER	DASHBOARD_STATE_FLG	Yes
CI_BATCH_INST	START_DTTM	No
CI_BATCH_INST	END_DTTM	No
F1_MD_BI_TBL_FLD	LOGICAL_DM_NAME	No

Dropped Columns

None

Column Format Change

None

Primary Key Change

None

Index Changes

- Modified secondary index, F1T819S1 on table F1_ATTACHMENT to include MO + PK_Val 1-5 + attachment ID.

  ```
  CREATE UNIQUE INDEX F1T819S1 ON F1_ATTACHMENT (MAINT_OBJ_CD,
  PK_VAL1, PK_VAL2, PK_VAL3, PK_VAL4, PK_VAL5, ATTACHMENT_ID) ;
  ```

- Added new index secondary index, F1T819S3 on table F1_ATTACHMENT for columns ATTACHMENT_FILE_NAME + ATTACHMENT_ID

  ```
  CREATE UNIQUE INDEX F1T819S3 ON F1_ATTACHMENT
  (ATTACHMENT_FILE_NAME, ATTACHMENT_ID) ;
  ```

Upgrades to Oracle Utilities Application Framework v4.5.0.0.0

New Tables

Table	Description	Table Type
F1_INSIGHT_TYPE	Insight Type	Admin System
F1_INSIGHT_TYPE_CHAR	Insight Type Characteristic	Admin System
F1_INSIGHT_TYPE_L	Insight Type Language	Admin System
F1_INSIGHT_TYPE_ALG	Insight Type Algorithm	Admin System
F1_INSIGHT_TYPE_USER_ACT	Insight Type User Action	Admin System
F1_INSIGHT_TYPE_USER_ACT_ALG	Insight Type User Action Algorithm	Admin System
F1_INSIGHT_TYPE_VISUAL_SETTING	Insight Type Visual Setting	Admin System
F1_COLOR_OPT	Color Option	Admin System
F1_COLOR_OPT_L	Color Option Language	Admin System
F1_COLOR_OPT_ITEM	Color Option Item	Admin System
F1_TREE	Tree	Admin System
F1_TREE_L	Tree Language	Admin System
F1_TREE_NODE	Tree Node	Admin System
F1_TREE_NODE_L	Tree Node Language	Admin System
F1_TREE_NODE_ALG	Tree Node Algorithms	Admin System
F1_INSIGHT_GRP	Insight Group	Admin
F1_INSIGHT_GRP_L	Insight Group Language	Admin
F1_INSIGHT_GRP_TYPE	Insight Group Insight Type	Admin
F1_INSIGHT_GRP_VISUAL_STR	Insight Group Valid Visual Structure	Admin
F1_INSIGHT_GRP_CHAR	Insight Group Characteristics	Admin

New Views

None

Dropped Tables

None

Unsupported Tables

None

Added Columns

Table	Field	Required
SC_USER	DASHBOARD_LOC_FLG	Yes
SC_USER	DASHBOARD_STATE_FLG	Yes
CI_BATCH_INST	START_DTTM	No
CI_BATCH_INST	END_DTTM	No
F1_MD_BI_TBL_FLD	LOGICAL_DM_NAME	No

Dropped Columns

None

Column Format Change

None

Primary Key Change

None

Index Changes

- Modified secondary index, F1T819S1 on table F1_ATTACHMENT to include MO + PK_Val 1-5 + attachment ID.

  ```
  CREATE UNIQUE INDEX F1T819S1 ON F1_ATTACHMENT (MAINT_OBJ_CD,
  PK_VAL1, PK_VAL2, PK_VAL3, PK_VAL4, PK_VAL5, ATTACHMENT_ID) ;
  ```

- Added new index secondary index, F1T819S3 on table F1_ATTACHMENT for columns ATTACHMENT_FILE_NAME + ATTACHMENT_ID

  ```
  CREATE UNIQUE INDEX F1T819S3 ON F1_ATTACHMENT
  (ATTACHMENT_FILE_NAME, ATTACHMENT_ID) ;
  ```

Upgrades to Oracle Utilities Application Framework v4.5.0.1.1

New Tables
None

New Views

View Name	Description
F1_MENU_APP_SVCS_VW	Lists all application services that are associated with menu lines.

Updated Views
None

Dropped Tables
None

Removed Tables
None

Unsupported Tables
None

New Columns

Table	Field	Required
CI_BATCH_RUN	LOG_FILE_NAME	No
F1_BATCH_RUN_ANALYTICS_SNAP	ELAPSED_SECS	No

Removed Columns

Table	Field
F1_BATCH_RUN_ANALYTICS_SNAP	ELAPSED_TIME
F1_BATCH_THD_ANALYTICS_SNAP	JOB_END_MINS
F1_BATCH_THD_ANALYTICS_SNAP	JOB_START_MINS
F1_BATCH_THD_ANALYTICS_SNAP	JOB_TOTAL_MINS

Dropped Columns

None

Column Format Changes

None

Primary Key Change

None

Upgrades to Oracle Utilities Application Framework v25.4

New Indexes

Table_Name	Index_Name
CI_BATCH_RUN	XT028S2
F1_BUS_FLG	F1T681S2

New Tables
None

New Views
None

Updated Views
None

Dropped Tables
None

Removed Tables
None

Unsupported Tables
None

New Columns
None

Removed Columns
None

Dropped Columns
None

Column Format Changes
None

Primary Key Change

None

Appendix C

Oracle Application Framework System Table Guide

This section lists the system tables owned by the Oracle Utilities Application Framework V25.4 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 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

Properties	Description
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
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 columns that can be updated 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

Properties	Description
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 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 during the upgrade process, if the system finds a record in the target environment with a primary key that matches system data, the record will be updated with the system data. For example: If an implementation adds an option to the MO table (CI_MD_MO_OPT) and subsequently, the product releases that same option configuration, the product's row overrides the "taking ownership" CM row.

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)
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

Properties	Description
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

Data Export Control

Properties	Description
Tables	F1_DATA_EXPORT_CTRL, F1_DATA_EXPORT_CTRL_LOG, F1_DATA_EXPORT_CTRL_LOG_PARM
Standard Data Fields	F1_DATA_EXPORT_CTRL (DATA_EXPORT_CD)
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.

File Integration

Properties	Description
Tables	F1_FILE_INT_REC, F1_FILE_INT_REC_L, F1_FILE_INT_REC_ALG, F1_FILE_INT_TYPE, F1_FILE_INT_TYPE_L
Standard Data Fields	File Integration Record (FILE_INT_REC_CD), File Integration Type (FILE_INT_TYPE_CD)

Properties	Description
Customer Modification	None

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

Key Ring

Properties	Description
Tables	F1_CRYPTO_KEY_RING, F1_CRYPTO_KEY_RING_L
Standard Data Fields	Key Ring (KEY_RING_CD)
Customer Modification	None

Lookup

Properties	Description
Tables	CI_LOOKUP_FIELD, CI_LOOKUP_VAL, CI_LOOKUP_VAL_L

Properties	Description
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
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

Market Configuration

Properties	Description
Tables	F1_MKTCFG
Standard Data Fields	Market Configuration Code (MKTCFG_CD)

Properties	Description
Customer Modification	None

Messages

Properties	Description
Tables	CI_MSG_CATEGORY, CI_MSG_CATEGORY_L, CI_MSG, CI_MSG_L
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 - BI

Properties	Description
Tables	F1_MD_BI_TBL, F1_MD_BI_TBL_L, F1_MD_BI_TBL_FLD
Standard Data Fields	Table Name (BI_TBL_NAME)
Customer Modification	None

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, CI_MD_IDX, CI_MD_IDX_FLD, 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. • Index Code (IDX_CD): Index name must match the physical index name in the database. • F1_DB_OBJECTS_REPO: Table stores information about Indexes, Sequences, Triggers and other database objects excluding Tables and Fields and Indexes (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

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

Properties	Description
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)

Unsupported Metadata

Properties	Description
Tables	F1_LGCY_OBJ

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

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

Web Service Configuration (Additional)

Properties	Description
Tables	CI_XAI_ADAPTER, CI_XAI_ADAPTER_L, CI_XAI_CLASS, CI_XAI_CLASS_L, 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), Sender Id (XAI_SENDER_ID)
Customer Modification	Option Value (OPTION_VALUE) on Message Option Table (CI_XAI_OPTION)

XAI Inbound 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)

Product Metric Type

Properties	Description
Tables	F1_PROD_METRIC_TYPE,F1_PROD_METRIC_TYPE_L

Properties	Description
Standard Data Fields	Product Metric Type (PROD_METRIC_TYPE_CD)
Customer Modification	

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_CURRENCY_CD	KP	Currency Code
CI_CURRENCY_CD_L	KP	Currency Code Language
CI_DISP_ICON	MG	Display Icon
CI_DISP_ICON_L	MG	Display Icon Language
CI_DISP_PROF	KP	Display Profile
CI_DISP_PROF_L	KP	Display Profile 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_AT_DTL	RF	MD Element Attribute Type Detail
CI_MD_AT_DTL_L	RF	MD Element Attribute Type Detail Language
CI_MD_ATT_TY	RF	MD Element Attribute Type
CI_MD_CONST	MG	Constraints
CI_MD_CONST_FLD	MG	Constraint Fields
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_FLD	MG	Field
CI_MD_FLD_ADDTL	MG	Field Additional Attributes Table
CI_MD_FLD_L	MG	Field Language
CI_MD_LOOKUP_F	RF	MD Lookup Field
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

Table Name	Upgrade Action	Description
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_MSG	RF	MD Message
CI_MD_MSG_L	RF	MD Message Language
CI_MD_NAV	MG	Navigation Key
CI_MD_PDF	RF	Predefined Fields
CI_MD_PDF_VAL	RF	Predefined Values
CI_MD_PRG_COM	MG	Program Components
CI_MD_PRG_EL_AT	MG	UI Page Element Attributes
CI_MD_PRG_ELEM	MG	UI Page Elements
CI_MD_PRG_LOC	MG	Program Location
CI_MD_PRG_MOD	MG	Program Module
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_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_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

Table Name	Upgrade Action	Description
CI_MD_SOCG_SORT	MG	Search Criteria Group Result Sort Order
CI_MD_SRC_TYPE	RF	Source Type
CI_MD_SRC_TYPE_L	RF	Source Type Language
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_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_MD_WRK_TBL	MG	Work Table
CI_MD_WRK_TBL_L	MG	Work Table Language
CI_MD_WRK_TBLFLD	MG	Work Table Field
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
CI_PORTAL	MG	Portal
CI_PORTAL_L	MG	Portal Language

Table Name	Upgrade Action	Description
CI_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_TIME_ZONE	KP	Time Zone
CI_TIME_ZONE_L	KP	Time Zone Language
CI_UI_ZONE	MG	Context Sensitive Zone
CI_USR_NAV_LINK	MG	User Favorite Links
CI_USR_PORTAL	KP	User Portal
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

Table Name	Upgrade Action	Description
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
CI_XAI_SVC_PARM	MG	XAI Inbound Service Parameters
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_CHAR	KP	Status Reason Characteristic
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_CRYPTO_KEY_RING	KP	Key Ring Table used to store cryptographic keys

Table Name	Upgrade Action	Description
F1_CRYPTO_KEY_RING_L	KP	Key Ring Language Table
F1_DATA_AREA	MG	Data Area
F1_DATA_AREA_L	MG	Data Area Language
F1_DB_OBJECTS_REPO	MG	Database Objects Repository
F1_EXT_LOOKUP_VAL	MG	Extendable Lookup
F1_EXT_LOOKUP_VAL_CHAR	MG	Extendable Lookup Characteristics
F1_EXT_LOOKUP_VAL_L	MG	Extendable Lookup Language
F1_FILE_INT_REC	MG	File Integration Record Table
F1_FILE_INT_REC_ALG	MG	File Integration Record Algorithm Table
F1_FILE_INT_REC_L	MG	File Integration Record Language Table
F1_FILE_INT_TYPE	MG	File Integration Type
F1_FILE_INT_TYPE_L	MG	File Integration Language Table
F1_INSTALLATION	KP	Installation Option - Framework
F1_IWS_ANN	MG	Web Service Annotation
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_LGCY_OBJ	RF	Unsupported Metadata
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_MD_BI_TBL	MG	MD Table for BI
F1_MD_BI_TBL_FLD	MG	MD Field Table for BI

Table Name	Upgrade Action	Description
F1_MD_BI_TBL_L	MG	MD Language Table for BI
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_MKTCFG	MG	Market Configuration
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_INCL_SVC	MG	Web Service Category Included Services
F1_WEB_CAT_L	MG	Web Service Category Language
SC_ACCESS_CNTL	MG	User Group Access Control
SC_APP_SERVICE	MG	Application Service
SC_APP_SERVICE_L	MG	Application Service Language
SC_USER	KP	User
SC_USER_CHAR	KP	User Characteristic
SC_USER_GROUP	KP	User Group
SC_USER_GROUP_L	KP	User Group Language
SC_USR_GRP_PROF	MG	User Group Profile

Table Name	Upgrade Action	Description
SC_USR_GRP_USR	KP	User Group User
F1_PROD_METRIC_TYPE	MG	Product Metric Type
F1_PROD_METRIC_TYPE_L	MG	Product Metric Type Language