

Oracle Utilities Smart Grid Gateway

Database Administrator's Guide

Release 2.4.0.0.0

F38824-01

April 2021

Oracle Utilities Smart Grid Gateway Database Administrator's Guide, Release 2.4.0.0.0

Copyright © 2000, 2021 Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software" or "commercial computer software documentation" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Contents

Preface	i
Related Documents	ii
Updates to this Documentation	ii
Conventions.....	iii
Acronyms	iii
Additional Resources	iii
Chapter 1	
Database Overview	1-1
Supported Platforms Summary Table	1-2
Support for Software Patches and Upgrades	1-2
Database Maintenance Rules	1-3
Permitted Database Changes.....	1-3
Non-Permitted Database Changes	1-3
Chapter 2	
Installing the Database	2-1
Initial Install.....	2-4
Copying and Decompressing Install Media	2-4
Creating the Database.....	2-5
Installing the CISADM Schema.....	2-8
Post-installation Tasks.....	2-13
Upgrade Install.....	2-15
Supported Upgrade Paths	2-15
Copying and Decompressing Install Media	2-15
Installing the CISADM Schema.....	2-16
Post-installation Tasks.....	2-21
Demo Install.....	2-23
Copying and Decompressing Install Media	2-24
Creating the Database and Importing the Dump File.....	2-24
Post-installation Tasks.....	2-25
Chapter 3	
Database Design	3-1
Database Object Standards	3-2
Categories of Data.....	3-2
Naming Standards	3-2
Column Data Type and Constraints	3-5
User Defined Code	3-5
System Assigned Identifier	3-6
Date/Time/Timestamp	3-6
Number.....	3-6
Fixed Length/Variable Length Character Columns	3-6
Null Column Support.....	3-6

XML Type Support.....	3-7
Cache and Key Validation Flags	3-7
Table Classification and Table Volume Flags.....	3-7
Default Value Setting.....	3-7
Foreign Key Constraints	3-7
Standard Columns	3-8
Owner Flag.....	3-8
Version.....	3-8

Chapter 4

Database Implementation Guidelines.....	4-1
Index	4-2
Table Partitioning Recommendations.....	4-2
Transparent Data Encryption Recommendations	4-2
Data Compression Recommendations	4-3
Database Vault Recommendations	4-4
Oracle Fuzzy Search Support.....	4-5
Information Lifecycle Management (ILM) and Data Archiving Support.....	4-5
Storage Recommendations	4-5
ILM Enabled Tablespace Requirements	4-6
Database Configuration Recommendations	4-6
Database Syntax.....	4-7
Database Initialization Parameters	4-7
Oracle Database Implementation Guidelines	4-8
Oracle Partitioning.....	4-8
Database Statistic.....	4-8
Materialized View.....	4-8

Chapter 5

Conversion Tools	5-1
Installing the Script.....	5-2
Preparing the Production Database.....	5-3
Preparing the Staging Database	5-3

Chapter 6

Information Lifecycle Management and Data Archiving in SGG	6-1
ILM Implementation Overview	6-2
ILM Implementation Components	6-2
ILM Database Administrator's Tasks.....	6-3
Preparation Phase.....	6-3
On-going Maintenance Phase	6-40
Naming Convention.....	6-43

Appendix A

Sample SQL for Enabling ILM in SGG (Initial Installation).....	A-1
Maintenance Object: TO DO ENTRY	A-1
Parent Table: CI_TD_ENTRY	A-1
Child Table: CI_TD_DRLKEY	A-3
Child Table: CI_TD_ENTRY_CHA.....	A-4
Child Table: CI_TD_LOG.....	A-4
Child Table: CI_TD_MSG_PARM	A-5
Child Table: CI_TD_SRTKEY	A-5
Parent Table: F1_SYNC_REQ_IN	A-6
Child Table: F1_SYNC_REQ_IN_CHAR.....	A-9
Child Table: F1_SYNC_REQ_IN_EXCP	A-9
Child Table: F1_SYNC_REQ_IN_EXCP_PARM.....	A-10
Child Table: F1_SYNC_REQ_IN_LOG	A-10
Child Table: F1_SYNC_REQ_IN_LOG_PARM.....	A-11
Child Table: F1_SYNC_REQ_IN_REL_OBJ	A-11

Parent Table: D1_INIT_MSRMT_DATA.....	A-12
Child Table: D1_INIT_MSRMT_DATA_CHAR.....	A-16
Child Table: D1_INIT_MSRMT_DATA_LOG.....	A-17
Child Table: D1_INIT_MSRMT_DATA_LOG_PARM.....	A-17
Child Table: D1_INIT_MSRMT_DATA_K.....	A-18

Appendix B

Sample SQL for Enabling ILM in SGG (Existing Installation).....	B-1
---	-----

Appendix C

Sample SQL for Enabling ILM with Sub Retention in SGG (Existing Installation)	C-1
---	-----

Appendix D

Sample SQL for Periodic Maintenance	D-1
Archive Partition.....	D-2
Restore Partition.....	D-5

Appendix E

Partitioning and Compression Recommendations	E-1
Partitioning Recommendations	E-2
D1_MSRMT.....	E-2
D1_MSRMT_CHAR.....	E-4
D1_MSRMT_LOG.....	E-5
D1_MSRMT_LOG_PARM.....	E-7
D1_INIT_MSRMT_DATA.....	E-8
D1_INIT_MSRMT_DATA_CHAR.....	E-11
D1_INIT_MSRMT_DATA_K.....	E-12
D1_INIT_MSRMT_DATA_LOG.....	E-12
D1_INIT_MSRMT_DATA_LOG_PARM.....	E-13
Compression Recommendations	E-14

Appendix F

Database Changes in Oracle Utilities Smart Grid Gateway.....	F-1
Upgrading from Oracle Utilities Smart Grid Gateway V2.3.0.2.0 to V2.4.0.0.0.....	F-2
Upgrading from Oracle Utilities Smart Grid Gateway V2.3.0.2.3 to V2.4.0.0.0.....	F-7

Appendix G

Upgrades to the Oracle Utilities Application Framework Database	G-1
Upgrading from Oracle Utilities Application Framework v4.3.0.1.0 to v4.3.0.4.0.....	G-2
Upgrading from Oracle Utilities Application Framework v4.3.0.2.0 to v4.3.0.4.0.....	G-4
Upgrading from Oracle Utilities Application Framework v4.3.0.3.0 to v4.3.0.4.0.....	G-6
Upgrading from Oracle Utilities Application Framework v4.3.0.4.0 to v4.3.0.5.0.....	G-8
Upgrading from Oracle Utilities Application Framework v4.3.0.5.0 to v4.3.0.6.0.....	G-11
Upgrading from Oracle Utilities Application Framework v4.3.0.6.0 to v4.4.0.0.0.....	G-14
Upgrading from Oracle Utilities Application Framework v4.4.0.0.0 to v4.4.0.2.0.....	G-15
Upgrading from Oracle Utilities Application Framework v4.4.0.2.0 to v4.4.0.3.0.....	G-18

Appendix H

Oracle Application Framework System Table Guide.....	H-1
System Table Standards.....	H-2
Business Configuration Tables.....	H-3
Development and Implementation System Tables.....	H-5

Preface

Welcome to the Oracle Utilities Smart Grid Gateway Database Administrator's Guide.

This guide provides instructions for installing and maintaining the database for Oracle Utilities Smart Grid Gateway.

The preface includes:

- [Audience](#)
- [Related Documents](#)
- [Updates to this Documentation](#)
- [Conventions](#)
- [Acronyms](#)
- [Additional Resources](#)

Audience

This guide is intended for database administrators who will be installing and maintaining the database for Oracle Utilities Smart Grid Gateway.

To complete installation you should have:

- Administrative privileges on the host where you are installing the software.
- Experience installing and configuring application servers and other software.

Related Documents

The following documentation is included with this release.

Installation Guides and Release Notes

- *Oracle Utilities Smart Grid Gateway Release Notes*
- *Oracle Utilities Smart Grid Gateway Quick Install Guide*
- *Oracle Utilities Smart Grid Gateway Installation Guide*
- *Oracle Utilities Smart Grid Gateway Database Administrator's Guide*
- *Oracle Utilities Smart Grid Gateway Licensing Information User Manual*

Configuration and User Guides

- *Oracle Utilities Meter Solution Business User Guide*
- *Oracle Utilities Meter Solution Administrative User Guide*

Supplemental Documents

- *Server Administration Guide*
- *Security Guide*

Updates to this Documentation

This documentation is provided with the version of the product indicated. For latest updates to documentation visit [Oracle Technology Network](#).

Additional and updated information about the operations and configuration of the product is available on [My Oracle Support](#).

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.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Acronyms

The following acronyms and terms are used in this document:

Acronym	Definition
OUAF	Oracle Utilities Application Framework
OIM	Oracle Identity Management
SGG	Oracle Utilities Smart Grid Gateway
ILM	Information Lifecycle Management

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 Smart Grid Gateway database, including:

- [Supported Database Platforms](#)
- [Database Maintenance Rules](#)

Supported Database Platforms

This section defines the platforms on which Oracle Utilities Smart Grid Gateway is verified to operate.

Supported Platforms Summary Table

Oracle Utilities Smart Grid Gateway is certified on the following platforms:

Platform	Database Version
AIX 7.1 TL5/7.2. TL3+ (POWER 64-bit)	Oracle DB 19c (64-bit) Oracle Database Server 19c
Oracle Enterprise Linux 7.x/8.x x86_64 (64-bit)	Oracle DB 19c (64-bit) Oracle Database Server 19c
Oracle Solaris 12 (SPARC 64-bit)	Oracle DB 19c (64-bit) Oracle Database Server 19c
Windows Server 2012 R2 (x86_64 64-bit)	Oracle DB 19c (64-bit) Oracle Database Server 19c

* Oracle Utilities Smart Grid Gateway is tested and supported on the versions of Oracle Linux specified. Because Oracle Linux is 100% userspace-compatible with Red Hat Enterprise Linux, Oracle Utilities Smart Grid Gateway also is supported on Red Hat Enterprise Linux for this release.

Note: For Oracle Linux 7.x applicable end of support dates, refer to the [Oracle Lifetime Support Policy: Oracle and Sun System Software and Operating Systems](#) document.

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 19c

Note: Oracle Database Enterprise Edition and the Partitioning and Advanced Compression options are strongly recommended in all situations.

Visit [My Oracle Support](#) for additional details.

Support for Software Patches and Upgrades

Due to the ongoing nature of software improvement, vendors issue patches and service packs for the operating systems, application servers, and database servers, on top of specific versions that Oracle Utilities Smart Grid Gateway 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 Smart Grid Gateway production environment.

The exceptions from this rule are Hibernate software V4.0 GA and the Oracle Client V12.1.0.1. These versions should not be upgraded.

Always contact Oracle Utilities Smart Grid Gateway 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 the instructions to install the Oracle Utilities Smart Grid Gateway database, including:

- [Installation Overview](#)
- [Initial Install](#)
- [Upgrade Install](#)
- [Demo Install](#)

Installation Overview

Refer to [Supported Database Platforms](#) for information about the supported platforms on which Oracle Utilities Smart Grid Gateway is verified to operate.

The following types of installation are available for Oracle Utilities Smart Grid Gateway:

- [Initial Install](#) - a database with no demo data
- [Upgrade Install](#) - a database upgrade from V2.3.0.2.0 to V2.4.0.0.0.
- [Demo Install](#) - a database populated with demo data.

Note: The Oracle Utilities Meter Data Management Database Component is a cumulative package comprising of data used by all the Oracle Utilities Smart Grid Gateway and Oracle Utilities Service Order Management adapters.

The database installation requires a supported version of the Java Development Kit to be installed on the Windows desktop where the install package is staged and run from. Refer to the **Supported Platforms** section in the *Oracle Utilities Smart Grid Gateway Installation Guide* for the required Java versions.

For an Initial Install or Demo Install, create an empty database on the Unix or Windows server, and then populate the database with data. For a database Upgrade Install, upgrade the current Oracle Utilities Smart Grid Gateway database.

Create the database using the Database Configuration Assistant (DBCA). Refer to the *Master Note: Overview of Database Configuration Assistant (DBCA)* (Doc ID 1488770.1) on [My Oracle Support](#).

Make sure to set the character set for database as follows:

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

While prior versions of the product have included the cdxdba programs (cdxdba.plx for UNIX or CDXDBA.exe for Windows), this is no longer supported going forward, and the Database Configuration Assistant should be used instead.

Enable Mandatory Software Options

- Oracle Spatial OR Oracle Locator
- Oracle Text

Run the following SQL to ensure it is successful:

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

Create default tablespace CISTS_01 and required users and roles as follows:

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

Create required role:

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

Create Users:

```

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

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

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

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

```

Review the Storage.xml file prior to an Initial Install or Upgrade Install. Information in this file is used by OraDBI.jar while installing and upgrading the Oracle Utilities Smart Grid Gateway database objects.

Note: Even if you have created the database without using the CDXDBA script, you will still need to review the Storage.xml file, prior to an initial install, to update the default values to custom values (for example: TableSpace Name).

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

The Storage.xml file is as shown:

```

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

```

For optimum storage allocation, database administrators should create multiple tablespaces with extents sized to store different types of tables/indexes. They can then edit this file before each upgrade and 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, securefile options, Advanced Compression, and parallel information are used only for new objects. Therefore, for initial installs, information for

each object should be reviewed. For upgrades, only tablespace information for objects added in the current release needs to be reviewed. Be careful while editing the Storage.xml 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.

Initial Install

This section describes how to install the database components of Oracle Utilities Smart Grid Gateway, including:

- [Copying and Decompressing Install Media](#)
- [Creating the Database](#)
- [Installing the CISADM Schema](#)
- [Post-installation Tasks](#)

Note: The Oracle Utilities Meter Data Management Database Component is a cumulative package comprising of data used by all the Oracle Utilities Smart Grid Gateway and Oracle Utilities Service Order Management adapters.

Copying and Decompressing Install Media

To copy and decompress the Oracle Utilities Application Framework and Oracle Utilities Smart Grid Gateway databases:

1. Download the Oracle Utilities Application Framework V4.4.0.3.0 Multiplatform and Oracle Utilities Meter Data Management V2.4.0.0.0 Multiplatform from Oracle Software Delivery Cloud to install any of the following adapters:
 - Oracle Utilities Smart Grid Gateway Adapter Development Kit
 - Oracle Utilities Smart Grid Gateway Adapter MV-90 Adapter for Itron
 - Oracle Utilities Smart Grid Gateway Adapter for Networked Energy Services
 - Oracle Utilities Smart Grid Gateway Adapter for Itron OpenWay
 - Oracle Utilities Smart Grid Gateway Adapter for Sensus
 - Oracle Utilities Smart Grid Gateway Adapter for Landis+Gyr
 - Oracle Utilities Smart Grid Gateway Adapter for Silver Spring Networks
2. Create a temporary directory, such as C:\OUMSM\temp or /OUMSM/temp (referred to below as <TEMPDIR>).

This directory must be located outside any current working Oracle Utilities application environment. All the files that are placed in this directory as a part of the installation can be deleted after a successful installation.

3. Copy the FW_V4.4.0.3.0.zip and MDM_V2.4.0.0.0.zip files from the downloaded package to the <TEMPDIR>/DB directory.
4. Unzip the FW_V4.4.0.3.0.zip and MDM_V2.4.0.0.0.zip files to a temporary folder named DB. These files contain the database components required to install Oracle Utilities Application Framework and Oracle Utilities Meter Data Management databases.

Creating the Database

You must have Oracle Database Server installed on your machine to create the database. The database can be created using Database Configuration Assistant.

Using DBCA

To create an Initial Install or production database it is recommended that you use the Database Configuration Assistant. Once the database is created the instance configuration can be done according to the environment requirements and based on the production recommendations.

The script to create product users is located under the relevant database version subdirectory of the DatabaseCreation directory.

1. Create the CISTS_01 tablespace before running the script to create the product users.
2. Run the .../DB/MDM_V2.4.0.0.0/DatabaseCreation/users.sql after logging into the database as sys user to create the product users.

The UNIX and Windows database creation utilities create an empty database with AL32UTF8 character set and at least one tablespace for storing the application objects before running the installation. The default name of the application tablespace is CISTS_01.

This section includes:

- [Creating the Database on Unix](#)
- [Creating the Database on Windows](#)

Creating the Database on Unix

Create the database using the Database Configuration Assistant (DBCA). Refer to the *Overview of Database Configuration Assistant (DBCA) (Doc ID 1488770.1)* document on [My Oracle Support](#).

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

Refer to the [Installation Overview](#) section for instructions to create the database.

Creating the Database on Windows

The files for creating the database are located in the .../DB/MDM.V2.4.0.0.0/DatabaseCreation/Windows directory in directory.

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.

Create the database using the Database Configuration Assistant (DBCA). Refer to the *Overview of Database Configuration Assistant (DBCA) (Doc ID 1488770.1)* document on [My Oracle Support](#).

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

Refer to the [Installation Overview](#) section for instructions to create the database.

Database Globalization Support Consideration

The Oracle Utilities Application Framework is a multilingual capable application that supports the storage, processing and retrieval of data in multiple languages by leveraging the Oracle Database globalization support architecture. Use of the AL32UTF8 Unicode character encoding system allows the database to support multiple languages.

By default the database is created with BYTE length semantics. If the data is being stored using CHARACTER length semantics, set NLS_LENGTH_SEMANTICS to CHAR at session level via a login trigger during installation.

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: Refer to Doc ID 313175.1 on [My Oracle Support](#).
- Alternative procedure: The following is an alternate way to create a schema with character-length semantics and to import the data from a byte-based export.

Migrating from BYTE based storage to CHARACTER based storage

1. Create database using DBCA.

2. Run following statement to set nls_length_semantics=CHAR.

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

3. Restart the database.

4. Make sure nls_length_semantics is CHAR.

```
SQL> SHOW PARAMETER nls_length_semantics
```

Note: For pluggable database make sure to set nls_length_semantics=CHAR for pluggable database.

5. Export schema from database which has nls_semantics_legth=BYTE.

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

6. 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
```

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

For vi editor in Linux environment use the following command to replace Byte with Char.

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

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

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

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

10. If any object is missing for any reason, create it, by fixing DDL manually (DDL for each object is available in the file which was created in step 6). Then, execute DDL for the objects which are not created.

11. Generate DDL to disable triggers using following command.

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

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

14. Enable triggers.

To generate DDL for triggers:

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

15. Run the script generated in step 14 to enable all triggers.

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 needs to make use of 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

Make sure to follow the instructions provided in the [Oracle Database](#) documentation for making this change in your database.

Exclude Table/ Index

To exclude an index or table during the upgrade process:

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

```
INDEX: 'INDEX_NAME', 'INDEX_NAME'
TABLE: 'TBAL1_NAME', 'TABLE2_NAME'
```

Installing the CISADM Schema

Install Oracle Utilities Application Framework V4.4.0.3.0 and then install Oracle Utilities Smart Grid Gateway V2.4.0.0.0. The files for Oracle Utilities Application Framework installation are located in the FW_V4.4.0.3.0 folder.

This section includes the following:

- [Installing the Oracle Utilities Application Framework Database Component using OraDBI.jar](#)
- [Installing the Oracle Utilities Smart Grid Gateway Database Component](#)

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

Before proceeding with the installation make sure to prepare the following parameters:

- The name of the database server in which the database is configured - DB_SERVER
- The listener port number of the database - PORT
- The target database name in which the product is to be installed - SERVICE_NAME
- A database user that will own the application schema (for example: CISADM) - DBUSER
- Password of the database user that will own the application schema - DBPASS
- A database user that has read-write (select/update/insert/delete) privileges to the objects in the application schema (for example: CISUSER). The application will access the database as this user - RWUSER
- A database user with read-only privileges to the objects in the application schema. (for example: CISREAD) - RUSER
- A database role that has read-write (select/update/insert/delete) privileges to the objects in the application schema. (for example: CIS_USER) - RW_USER_ROLE
- A database role with read-only privileges to the objects in the application schema. (for example: CIS_READ)- R_USER_ROLE
- Location for jar files. (The Jar files are bundled with the database package) - CLASSPATH
- Java Home (for example: <Java Install location>) - JAVA_HOME

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

- [Using Interactive Mode](#)
- [Using the Command Line Mode](#)

Using Interactive Mode

This procedure lists the steps to install the Oracle Utilities Application Framework V4.4.0.3.0 component using OraDBI.

Run the following command with the defined parameters on the command prompt from the ../DB/FW_V4.4.0.3.0/Install-Upgrade directory.

1. Open a command line prompt.

2. Set Java Home.

```
export JAVA_HOME=<Java Install location>/jdk1.8.0
```

3. Set the class path.

```
export CLASSPATH=<TEMPDIR>/DB/FW_V4.4.0.3.0/jarfiles/*
```

4. Run the following command:

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

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

- Name of the database server: <DB SERVER>
- Port no: <PORT>
- Name of the target database: <SERVICE_NAME>
- Name of the owner of the database schema: <DBUSER>
- Password of the user name: <DBPASS>
- Location of Java Home: (example: <Java Install location>): <Java Home>
- Oracle user with read-write privileges to the Database Schema: <CISUSER>
- Oracle user with read-only privileges to the Database Schema: <CISREAD>
- Oracle database role with read-write privileges to the Database Schema: <CIS_USER>
- Oracle database role with read-only privileges to the Database Schema: <CIS_READ>
- Enter the name of the target schema where you want to install or upgrade: <CISADM>
- Enter the password for the target schema: <CISADM password>

This process generates log files in the directory Install-Upgrade/logs. Make sure to check log files for any errors.

You are now ready to install the product using the Java based OraDBI.jar tool.

Using the Command Line Mode

Run the following command with the defined parameters on the command prompt from `../DB/FW_V4.4.0.3.0/Install-Upgrade` directory.

```
java com.oracle.ouaf.oem.install.OraDBI -d
jdbc:oracle:thin:@<DB_Server>:1521/
<SERVICE_NAME>,<DBUSER>,<DBPASS>,<RWUSER>,<RUSER>,<RW_USER_ROLE>,<
R_USER_ROLE>,<DBUSER> -l 1,2 - q true -p <RWUSER_PASS>,<RUSER_PASS>
```

This process generates log files in the directory `Install-Upgrade/logs`. Make sure to check log files for any errors.

Note: For OraDBI jar, 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 the Oracle Utilities Smart Grid Gateway Database Component

Note: The Oracle Utilities Meter Data Management Database Component is a cumulative package comprising of data used by all the Oracle Utilities Smart Grid Gateway and Oracle Utilities Service Order Management adapters.

Prepare the following parameters before installation:

- The name of the database server in which the database is configured - DB_SERVER
- The listener port number of the database - PORT
- The target database name in which the product is to be installed - SERVICE_NAME
- A database user that will own the application schema (for example: CISADM) - DBUSER
- Password of the database user that will own the application schema - DBPASS
- A database user that has read-write (select/update/insert/delete) privileges to the objects in the application schema (for example: CISUSER). The application will access the database as this user - RWUSER
- A database user with read-only privileges to the objects in the application schema. (for example: CISREAD) - RUSER
- A database role that has read-write (select/update/insert/delete) privileges to the objects in the application schema. (for example: CIS_USER) - RW_USER_ROLE
- A database role with read-only privileges to the objects in the application schema. (for example: CIS_READ) - R_USER_ROLE

- Location for jar files. (The Jar files are bundled with the database package) - CLASSPATH
- Java Home (for example: <Java Install location>) - JAVA_HOME

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

- [Using Interactive Mode](#)
- [Using the Command Line Mode](#)

Using the Interactive Mode

This procedure lists the steps to install the Oracle Utilities Smart Grid Gateway component using OraDBI.

Run the following command with the defined parameters on the command prompt from the ../DB/MDM_V2.4.0.0.0/Install-Upgrade/ directory.

1. Open a command line prompt.

2. Set Java Home.

```
export JAVA_HOME=<Java Install location>/jdk1.8.0
```

3. Set the class path.

```
export CLASSPATH=<TEMPDIR>/DB/FW_V4.4.0.3.0/jarfiles/*
```

4. Run the following command:

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

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

- Name of the database server: <DB SERVER>
- Port no: <PORT>
- Name of the target database: <SERVICE_NAME>
- Name of the owner of the database schema: <DBUSER>
- Password of the user name: <DBPASS>
- Location of Java Home: (example: <Java Install location>): <Java Home>
- Oracle user with read-write privileges to the Database Schema: <CISUSER>
- Oracle user with read-only privileges to the Database Schema: <CISREAD>
- Oracle database role with read-write privileges to the Database Schema: <CIS_USER>
- Oracle database role with read-only privileges to the Database Schema: <CIS_READ>
- Enter the name of the target schema where you want to install or upgrade: <CISADM>
- Enter the password for the target schema: <CISADM password>

This process generates log files in the directory Install-Upgrade/logs. Make sure to check log files for any errors.

Note: For OraDBI jar, 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)
```

Using the Command Line Mode

Run the following command with the defined parameters on the command prompt from ../DB/MDM_V2.4.0.0.0/Install-Upgrade/ directory.

```
java com.oracle.ouaf.oem.install.OraDBI -d
jdbc:oracle:thin:@<DB_Server>:1521/
<SERVICE_NAME>,<DBUSER>,<DBPASS>,<RWUSER>,<RUSER>,<RW_USER_ROLE>,<
R_USER_ROLE>,<DBUSER> -l 1,2 - q true -p <RWUSER_PASS>,<RUSER_PASS>
```

This process generates log files in the directory Install-Upgrade/logs. Make sure to check log files for any errors.

Note: For OraDBI jar, 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)
```

If you chose to continue, OraDBI first checks for the existence of each of the users specified and prompts for their password, default tablespace, and temporary tablespace.

OraDBI Performs the Following Tasks

- Interacts with the user to collect information about the name of Oracle account that will own the application schema (for example: CISADM), password of this account, the name of the Oracle account that the application user will use (for example: CISUSER), and the name of the Oracle account that will be assigned read-only privileges to the application schema (for example: CISREAD).
- Connects to the database as CISADM account, checks whether the user already has the application schema installed to verify whether this is an initial installation.
- 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.

Post-installation Tasks

- [Generating Database Statistics](#)
- [Enabling USER_LOCK Package](#)
- [Creating Activity Statistics Materialized View](#)
- [Configuring Security](#)
- [Populating the CI_INSTALL_PROD Table](#)
- [Creating Index D1T304S3 for Payload Statistic Functionality \(Optional\)](#)

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.

Enabling USER_LOCK Package

For In-bound 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 using the following steps.

1. Login as SYS user.
2. On SQL prompt run:

```
@?/rdbms/admin/userlock.sql
```
3. Grant permission by running following SQL:

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

Please note that grant can also be made to the database user which the Application connects to only instead of to public. For example: cisuser

Creating Activity Statistics Materialized View

To improve the performance of drill down queries, use the following procedure to create the materialized view and then refresh the materialized view.

From the command prompt, go to `../DB/MDM_V2.4.0.0.0/Post-Upgrade/` and run the scripts below.

1. Login to SQLPlus as CISADM user.
2. On SQL prompt run:


```
@Materialized_View_Creation.sql
```
3. Exit SQLPlus.

Configuring Security

To configure the security, follow these instructions:

1. Set PATH.


```
export JAVA_HOME=<Java Install location>/jdk1.8.0
export PATH=$JAVA_HOME/bin:$PATH
```
2. Set CLASSPATH.


```
export CLASSPATH=<TEMPDIR>/DB/FW_V4.4.0.3.0/jarfiles/*
```
3. Run the following command with the defined parameters on the command prompt.


```
java com.oracle.ouaf.oem.install.OraGenSec -d
<DBUSER>,<DBPASS>,jdbc:oracle:thin:@<DB_Server>:1521/
<SERVICE_NAME> -a A -r <R_USER_ROLE>,<RW_USER_ROLE> -u
<RUSER>,<RWUSER>
```

Populating the CI_INSTALL_PROD Table

After completing the Oracle Utilities Smart Grid Gateway database installation, run the scripts based on the adapter being upgraded.

Navigate to `../DB/MDM_V2.4.0.0.0/Post-Upgrade/` and connect to the CISADM schema and run the relevant script.

Adapter	Script
Oracle Utilities Smart Grid Gateway Adapter for Landis+Gyr	SGG_D3_Update.sql
Oracle Utilities Smart Grid Gateway Adapter for Networked Energy Services	SGG_D4_Update.sql
Oracle Utilities Smart Grid Gateway Adapter for Itron	SGG_D5_Update.sql
Oracle Utilities Smart Grid Gateway Adapter for Sensus RNI	SGG_D6_Update.sql
Oracle Utilities Smart Grid Gateway Adapter for Silver Spring Networks	SGG_D7_Update.sql
Oracle Utilities Smart Grid Gateway Adapter for Itron OpenWay	SGG_D8_Update.sql
Oracle Utilities Smart Grid Gateway Adapter Development Kit	SGG_DG_Update.sql

Creating Index D1T304S3 for Payload Statistic Functionality (Optional)

For an initial installation, this index does not exist. If you are using the payload statistic functionality, create the index. Connect to CISADM schema and execute the following:

```
CREATE UNIQUE INDEX D1T304S3 ON D1_INIT_MSRMT_DATA
(IMD_EXT_ID, INIT_MSRMT_DATA_ID);
```

Upgrade Install

This section describes how to upgrade the database components of Oracle Utilities Smart Grid Gateway, including:

- [Supported Upgrade Paths](#)
- [Copying and Decompressing Install Media](#)
- [Installing the CISADM Schema](#)
- [Post-installation Tasks](#)

Note that the Oracle Utilities Meter Data Management Database Component is a cumulative package comprising of data used by all the Oracle Utilities Smart Grid Gateway and Oracle Utilities Service Order Management adapters.

Supported Upgrade Paths

Direct upgrade to Oracle Utilities Smart Grid Gateway V2.4.0.0.0 is supported from Oracle Utilities Smart Grid Gateway V2.3.0.2.0.

The following section assumes an existing Oracle Utilities Smart Grid Gateway V2.3.0.2.0 installation on top of an Oracle Utilities Application Framework V4.4.0.2.0 installation, or Oracle Utilities Smart Grid Gateway V2.4.0.0.0 installation on top of an Oracle Utilities Application Framework V4.4.0.3.0 installation.

Copying and Decompressing Install Media

To copy and decompress the Oracle Utilities Application Framework and Oracle Utilities Smart Grid Gateway databases:

1. Download the Oracle Utilities Application Framework V4.4.0.3.0 Multiplatform and Oracle Utilities Meter Data Management V2.4.0.0.0 Multiplatform from Oracle Software Delivery Cloud to install any of the following adapters:
 - Oracle Utilities Smart Grid Gateway Adapter Development Kit
 - Oracle Utilities Smart Grid Gateway Adapter MV-90 Adapter for Itron
 - Oracle Utilities Smart Grid Gateway Adapter for Networked Energy Services
 - Oracle Utilities Smart Grid Gateway Adapter for Itron OpenWay
 - Oracle Utilities Smart Grid Gateway Adapter for Sensus
 - Oracle Utilities Smart Grid Gateway Adapter for Landis+Gyr
 - Oracle Utilities Smart Grid Gateway Adapter for Silver Spring Networks

2. Create a temporary directory, such as C:\OUMSM\temp or /OUMSM/temp (referred to below as <TEMPDIR>).

This directory must be located outside any current working Oracle Utilities application environment. All the files that are placed in this directory as a part of the installation can be deleted after a successful installation.

3. Copy the FW_V4.4.0.3.0.zip and MDM_V2.4.0.0.0.zip files from the downloaded package to the <TEMPDIR>\DB directory.
4. Unzip the FW_V4.4.0.3.0.zip and MDM_V2.4.0.0.0.zip files to a temporary folder named DB. These files contain the database components required to install Oracle Utilities Application Framework and Oracle Utilities Meter Data Management databases.

Installing the CISADM Schema

Make sure to take a backup of your database before carrying out the upgrade process.

You must install Oracle Utilities Application Framework V4.4.0.3.0 prior to installing Oracle Utilities Smart Grid Gateway. The Oracle Utilities Application Framework installation files are located in the FW_V4.4.0.3.0 folder.

This section includes the following:

- [Installing the Oracle Utilities Application Framework Database Component using OraDBI.jar](#)
- [Installing the Oracle Utilities Smart Grid Gateway Database Component](#)

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

Prepare the following parameters before installation:

- The name of the database server in which the database is configured - DB_SERVER
- The listener port number of the database - PORT
- The target database name in which the product is to be installed - SERVICE_NAME
- A database user that will own the application schema (for example: CISADM) - DBUSER
- Password of the database user that will own the application schema - DBPASS
- A database user that has read-write (select/update/insert/delete) privileges to the objects in the application schema (for example: CISUSER). The application will access the database as this user - RWUSER
- A database user with read-only privileges to the objects in the application schema. (for example: CISREAD) - RUSER
- A database role that has read-write (select/update/insert/delete) privileges to the objects in the application schema. (for example: CIS_USER) - RW_USER_ROLE

- A database role with read-only privileges to the objects in the application schema. (for example: CIS_READ)- R_USER_ROLE
- Location for jar files. (The Jar files are bundled with the database package) - CLASSPATH
- Java Home (for example: <Java Install location>) - JAVA_HOME

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

- [Using the Interactive Mode](#)
- [Using the Command Line Mode](#)

Using the Interactive Mode

The following procedure lists the steps to install the Oracle Utilities Application Framework V4.4.0.3.0 component using OraDBI.

Run the following command with the defined parameters on the command prompt from the ../DB/FW_V4.4.0.3.0/Install-Upgrade/ directory.

1. Open a command line prompt.

2. Set Java Home.

```
export JAVA_HOME=<Java Install location>/jdk1.8.0
```

3. Set the class path.

```
export CLASSPATH=<TEMPDIR>/DB/FW_V4.4.0.3.0/jarfiles/*
```

4. Run the following command:

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

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

- Name of the database server: <DB SERVER>
- Port no: <PORT>
- Name of the target database: <SERVICE_NAME>
- Name of the owner of the database schema: <DBUSER>
- Password of the user name: <DBPASS>
- Location of Java Home: (example: <Java Install location>): <Java Home>
- Oracle user with read-write privileges to the Database Schema: <CISUSER>
- Oracle user with read-only privileges to the Database Schema: <CISREAD>
- Oracle database role with read-write privileges to the Database Schema: <CIS_USER>
- Oracle database role with read-only privileges to the Database Schema: <CIS_READ>
- Enter the name of the target schema where you want to install or upgrade: <CISADM>
- Enter the password for the target schema: <CISADM password>

This process generates log files in the directory Install-Upgrade/logs. Make sure to check log files for any errors.

Note: For OraDBI jar, 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)
```

Using the Command Line Mode

Run the following command with the defined parameters on the command prompt from ../DB/FW_V4.4.0.3.0/Install-Upgrade directory

```
java com.oracle.ouaf.oem.install.OraDBI -d
jdbc:oracle:thin:@<DB_Server>:1521/
<SERVICE_NAME>,<DBUSER>,<DBPASS>,<RWUSER>,<RUSER>,<RW_USER_ROLE>,<
R_USER_ROLE>,<DBUSER> -l 1,2 - q true -p <RWUSER_PASS>,<RUSER_PASS>
```

This process generates log files in the directory Install-Upgrade/logs. Make sure to check log files for any errors.

Note: For OraDBI jar, 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)
```

Dropping Column from Database

Connect to CISADM schema through SQL and the drop the following column.

```
ALTER TABLE CI_XAI_RCVR_CTX DROP COLUMN CTXT_VAL;
```

If you are running Oracle GoldenGate, you may need to drop the log group. Use following script to drop the column.

```
select LOG_GROUP_NAME from user_log_groups where
table_name='CI_XAI_RCVR_CTX'
/
ALTER TABLE CI_XAI_RCVR_CTX DROP supplemental log group
GGS_CI_XAI_RCVR_CTX_111254
/
```

Installing the Oracle Utilities Smart Grid Gateway Database Component

The Oracle Utilities Meter Data Management Database Component is a cumulative package comprising of data used by all the Oracle Utilities Smart Grid Gateway and Oracle Utilities Service Order Management adapters.

Before installing the Oracle Utilities Smart Grid Gateway database component, prepare the following parameters:

- The name of the database server in which the database is configured - DB_SERVER
- The listener port number of the database - PORT
- The target database name in which the product is to be installed - SERVICE_NAME
- A database user that will own the application schema (for example: CISADM) - DBUSER
- Password of the database user that will own the application schema - DBPASS
- A database user that has read-write (select/update/insert/delete) privileges to the objects in the application schema (for example: CISUSER). The application will access the database as this user - RWUSER
- A database user with read-only privileges to the objects in the application schema. (for example: CISREAD) - RUSER
- A database role that has read-write (select/update/insert/delete) privileges to the objects in the application schema. (for example: CIS_USER) - RW_USER_ROLE
- A database role with read-only privileges to the objects in the application schema. (for example: CIS_READ)- R_USER_ROLE
- Location for jar files. (The Jar files are bundled with the database package) - CLASSPATH
- Java Home (for example: <Java Install location>) - JAVA_HOME

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

- [Using Interactive Mode](#)
- [Using the Command Line Mode](#)

Using the Interactive Mode

The following procedure lists the steps to install the Oracle Utilities Smart Grid Gateway component using OraDBI.

Run the following command with the defined parameters on the command prompt from the ../DB/MDM_V2.4.0.0.0/Install-Upgrade/ directory.

1. Open a command line prompt.

2. Set Java Home.

```
export JAVA_HOME=<Java Install location>/jdk1.8.0
```

3. Set the class path.

```
export CLASSPATH=<TEMPDIR>/DB/FW_V4.4.0.3.0/jarfiles/*
```

4. Run the following command:

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

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

- Name of the database server: <DB SERVER>
- Port no: <PORT>
- Name of the target database: <SERVICE_NAME>
- Name of the owner of the database schema: <DBUSER>
- Password of the user name: <DBPASS>
- Location of Java Home: (example: <Java Install location>): <Java Home>
- Oracle user with read-write privileges to the Database Schema: <CISUSER>
- Oracle user with read-only privileges to the Database Schema: <CISREAD>
- Oracle database role with read-write privileges to the Database Schema: <CIS_USER>
- Oracle database role with read-only privileges to the Database Schema: <CIS_READ>
- Enter the name of the target schema where you want to install or upgrade: <CISADM>
- Enter the password for the target schema: <CISADM password>

This process generates log files in the directory Install-Upgrade/logs. Make sure to check log files for any errors.

Note: For OraDBI jar, 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)
```

Using the Command Line Mode

Run the following command with the defined parameters on the command prompt from ../DB/MDM_V2.4.0.0/Install-Upgrade/ directory.

```
java com.oracle.ouaf.oem.install.OraDBI -d
jdbc:oracle:thin:@<DB_Server>:1521/
<SERVICE_NAME>,<DBUSER>,<DBPASS>,<RWUSER>,<RUSER>,<RW_USER_ROLE>,<
R_USER_ROLE>,<DBUSER> -l 1,2 -q true -p <RWUSER_PASS>,<RUSER_PASS>
```

This process generates log files in the directory Install-Upgrade/logs. Make sure to check log files for any errors.

Note: For OraDBI jar, 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)
If you chose to continue, OraDBI first checks for the existence of each of the users specified
and prompts for their password, default tablespace, and temporary tablespace.
```

OraDBI performs the following tasks

- Interacts with the user to collect information about the name of Oracle account that will own the application schema (for example: CISADM), password of this account and the name of the Oracle account that the application user will use (for example: CISUSER), and the name of the Oracle account that will be assigned read-only privileges to the application schema (for example: CISREAD).
- Connects to the database as CISADM account, checks whether the user already has the application schema installed to verify whether this is an initial installation.
- Verifies whether the upgrade path from the current release id to the target release id is supported by the upgrade.
- Verifies whether the tablespace names already exist in Storage.xml file (the process aborts, if not).
- Upgrades the schema, upgrades 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 necessarily mean a problem exists.

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

Post-installation Tasks

- [Generating Database Statistics](#)
- [Enabling USER_LOCK Package](#)
- [Creating Activity Statistics Materialized view](#)

- [Configuring Security](#)
- [Populating the CI_INSTALL_PROD Table](#)
- [Dropping Index D1T304S3 for Payload Statistic Functionality \(Optional\)](#)

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 using the DBMS_STATS package.

Enabling USER_LOCK Package

For In-bound 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 using the following steps.

1. Login as SYS user.
2. On SQL prompt run:


```
@?/rdbms/admin/userlock.sql
```
3. Grant permission by running following SQL:


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

Note that grant can also be made to the database user which the Application connects to only instead of to public. For example, cisuser.

Creating Activity Statistics Materialized view

To improve the performance of drill down queries, use the following procedure to create the materialized view and then refresh the materialized view.

1. Navigate to ../DB/MDM_V2.4.0.0.0/Post-Upgrade.
2. Login as CISADM user.
3. At the SQL prompt, run the following:


```
@Materialized_View_Creation.sql
```

Configuring Security

To configure the security:

1. Set PATH.


```
export JAVA_HOME=<Java Install location>/jdk1.8.0
export PATH=$JAVA_HOME/bin:$PATH
```
2. Set CLASSPATH.


```
export CLASSPATH=<TEMPDIR>/DB/FW_V4.4.0.3.0/jarfiles/*
```
3. Run the following command with the defined parameters on the command prompt.


```
java com.oracle.ouaf.oem.install.OraGenSec -d
<DBUSER>,<DBPASS>,jdbc:oracle:thin:@<DB_Server>:1521/
<SERVICE_NAME> -a A -r <RW_USER_ROLE>,<R_USER_ROLE> -u
<RWUSER>,<RUSER> -p <RWUSER_PASS>,<RUSER_PASS>
```

Populating the CI_INSTALL_PROD Table

After completing the Oracle Utilities Smart Grid Gateway database installation, run the scripts based on the adapter being upgraded.

Navigate to ..\MDM.V2.4.0.0.0\Post-Upgrade, connect to the CISADM schema, and run the relevant script.

Adapter	Script
Oracle Utilities Smart Grid Gateway Adapter for Landis+Gyr	SGG_D3_Update.sql
Oracle Utilities Smart Grid Gateway Adapter for Networked Energy Services	SGG_D4_Update.sql
Oracle Utilities Smart Grid Gateway Adapter for Itron	SGG_D5_Update.sql
Oracle Utilities Smart Grid Gateway Adapter for Sensus RNI	SGG_D6_Update.sql
Oracle Utilities Smart Grid Gateway Adapter for Silver Spring Networks	SGG_D7_Update.sql
Oracle Utilities Smart Grid Gateway Adapter for Itron OpenWay	SGG_D8_Update.sql
Oracle Utilities Smart Grid Gateway Adapter Development Kit	SGG_DG_Update.sql

Dropping Index D1T304S3 for Payload Statistic Functionality (Optional)

For an upgrade installation, this index already exists. If you are not using the Payload statistic functionality, or if you have no other SQL scripts referencing these fields, you may drop the index using the following SQL statement.

Connect to CISADM schema and run the following command.

```
DROP INDEX D1T304S3;
```

Demo Install

This section describes how to install the demo database components for Oracle Utilities Smart Grid Gateway, including:

- [Copying and Decompressing Install Media](#)
- [Creating the Database and Importing the Dump File](#)
- [Post-installation Tasks](#)

Note that the Oracle Utilities Meter Data Management Database Component is a cumulative package comprising of data used by all the Oracle Utilities Smart Grid Gateway and Oracle Utilities Service Order Management adapters.

Copying and Decompressing Install Media

To copy and decompress the Oracle Utilities Smart Grid Gateway database:

1. Download the Oracle Utilities Meter Data Management V2.4.0.0.0 Demo from Oracle Software Delivery Cloud for the following adapters:
 - Oracle Utilities Smart Grid Gateway Adapter Development Kit
 - Oracle Utilities Smart Grid Gateway Adapter for Networked Energy Services
 - Oracle Utilities Smart Grid Gateway Adapter for Itron OpenWay
 - Oracle Utilities Smart Grid Gateway Adapter for Landis+Gyr
 - Oracle Utilities Smart Grid Gateway Adapter for Sensus
 - Oracle Utilities Smart Grid Gateway Adapter for Silver Spring Networks
 - Oracle Utilities Smart Grid Gateway Adapter MV-90 Adapter for Itron
2. Unzip the downloaded file. It is extracted to the Demo directory.

Creating the Database and Importing the Dump File

This section describes steps to create the database and import the demo data dump file. It includes the following:

- [Creating the Demo Database on Unix or Windows](#)
- [Importing the Demo Dump File](#)

Note: When implementing Oracle Utilities Smart Grid Gateway with Oracle Utilities Meter Data Management, both the Smart Grid Gateway and Meter Data Management database components should be installed in the same database.

Creating the Demo Database on Unix or Windows

Follow the steps below to create the demo database:

1. Create the database using the Database Configuration Assistant (DBCA). Refer to [Creating the Database](#) for steps to create the database.
2. Make sure to set character set for database as AL32UTF8.
3. Create a directory in the database with the name 'data_pump_dir'.
4. Copy the demo dump file from ../Demo/ folder to the physical location on the disk that is mapped to data_pump_dir.
5. Unzip the demo dump file.

Importing the Demo Dump File

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

1. Set the correct ORACLE_SID and ORACLE_HOME.
2. Run the following command to import the demo dump:

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

Post-installation Tasks

This section describes the following post installation tasks:

- [Configuring Security](#)
- [Populating Language Data](#)
- [Populating the CI_INSTALL_PROD Table](#)

Configuring Security

To configure the security, follow these instructions:

1. Set PATH.

```
export JAVA_HOME=<Java Install location>/jdk1.8.0
export PATH=$JAVA_HOME/bin:$PATH
```

2. Set CLASSPATH.

```
export CLASSPATH=<TEMPDIR>/DB/FW_V4.4.0.3.0/jarfiles/*
```

3. Run the following command with the defined parameters at the command prompt.

```
java com.oracle.ouaf.oem.install.OraGenSec -d
<DBUSER>,<DBPASS>,jdbc:oracle:thin:@<DB_Server>:1521/
<SERVICE_NAME> -a A -r <RW_USER_ROLE>,<R_USER_ROLE> -u
<RWUSER>,<RUSER> -p <RWUSER_PASS>,<RUSER_PASS>
```

Populating Language Data

Please note that this database contains data in the ENGLISH language only. If you use any other supported language, you can run the F1-LANG batch program to duplicate the entries for new language records. For more information on running this batch program, refer to the user documentation section “Defining Background Processes.” You can also install the language specific demo data packages (if available) into the database. Please contact your Oracle representative to receive information on these packages.

Populating the CI_INSTALL_PROD Table

After completing the Oracle Utilities Smart Grid Gateway database installation, run the scripts based on the adapter being upgraded.

Navigate to ..\MDM.V2.4.0.0.0\Post-Upgrade, connect to the CISADM schema, and run the relevant script.

Adapter	Script
Oracle Utilities Smart Grid Gateway Adapter for Landis+Gyr	SGG_D3_Update.sql
Oracle Utilities Smart Grid Gateway Adapter for Networked Energy Services	SGG_D4_Update.sql
Oracle Utilities Smart Grid Gateway Adapter for Itron	SGG_D5_Update.sql
Oracle Utilities Smart Grid Gateway Adapter for Sensus RNI	SGG_D6_Update.sql
Oracle Utilities Smart Grid Gateway Adapter for Silver Spring Networks	SGG_D7_Update.sql

Adapter	Script
Oracle Utilities Smart Grid Gateway Adapter for Itron OpenWay	SGG_D8_Update.sql
Oracle Utilities Smart Grid Gateway Adapter Development Kit	SGG_DG_Update.sql

Chapter 3

Database Design

This chapter provides a standard for database objects such as tables, columns, and indexes, for products using the Oracle Utilities Application Framework. This standard helps 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.

The chapter includes:

- [Database Object Standards](#)
- [Column Data Type and Constraints](#)
- [Standard Columns](#)

Database Object Standards

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 the [Indexes](#) section 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 (LANGUAGE_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. 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. Prior to Version 4.1.0 of the framework the leading character of the base Owner Flag was used. From 4.1.0 on the first two characters of product's owner flag value should be used. For client specific implementation of index, use CM for Owner Flag.
- 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.

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, Framework its F1. Other examples are D1,D2, etc.
- 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

- OF stands for Owner Flag. For example, the framework is F1. Other examples are D1,D2, etc.
- 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 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

Oracle Utilities Application Framework 4.1.0 Group Fix 2 and later versions support 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 added to FW410 Group Fix 2 fixes the metadata to make sure that the existing tables will not be affected.

This new feature only supports tables maintained by Java. 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 COBOL.

XML Type Support

Oracle Utilities Application Framework v4.2.0.0 onwards supports XML Type. XML Type provides following advantages

1. The ability to use XQuery for querying nodes in the XML document stored within a column defined as XMLType.
2. The option to use the XML engine, which is built into the Oracle Database, to create indexes using nodes within the XML document stored in the XMLType column.

Cache and Key Validation Flags

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

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

Table Classification and Table Volume Flags

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

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

Default Value Setting

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

- When a predefined default value is not available, set the default value of Non-null CHAR or VARCHAR columns to blank except the primary key columns.
- When a predefined default value is not available, set the default value Non-null Number columns to 0 (zero) except the primary key columns.
- No database default values should be assigned to the Non Null Date, Time, and Timestamp columns.

Foreign Key Constraints

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

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 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

This chapter outlines the general implementation guidelines for the database components, including:

- [Configuration Guidelines](#)
- [Oracle Database Implementation Guidelines](#)

Configuration Guidelines

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

- [Index](#)
- [Table Partitioning Recommendations](#)
- [Transparent Data Encryption Recommendations](#)
- [Data Compression Recommendations](#)
- [Database Vault Recommendations](#)
- [Oracle Fuzzy Search Support](#)
- [Information Lifecycle Management \(ILM\) and Data Archiving Support](#)
- [Storage Recommendations](#)
- [Database Configuration Recommendations](#)
- [Database Syntax](#)
- [Database Initialization Parameters](#)

Index

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

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

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

Table Partitioning Recommendations

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

Transparent Data Encryption Recommendations

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

Please consider the following when implementing TDE:

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

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

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

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

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

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

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

- AES128 is the default encryption algorithm.
 - Tablespace level encryption is also supported using the following command:
- ```
Create tablespace <tablespace_name> logging datafile '<datafile
location>' size <initial size> reuse autoextend on next <next
size>
maxsize unlimited extent management local uniform size
<uniform size> encryption using 'AES128' default
storage(encrypt);
```
- Indexed columns can only be encrypted using the NO SALT Option. Salt is a way to strengthen the security of encrypted data. It is a random string added to the data before it is encrypted, causing repetition of text in the clear to appear different when encrypted.

## Data Compression Recommendations

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

### Exadata Hardware

For Exadata hardware the compression recommendations are:

- For the Final Measurement table (D1\_MSMT), keep the current table partition uncompressed. All of the older partitions will be compressed based on QUERY HIGH compression.
- For the Initial Measurement Data table (D1\_INIT\_MSMRT\_DATA), always keep CLOBs in securefiles and with MEDIUM compression. Also keep the current table partition uncompressed. All of the older partitions will be compressed based on QUERY HIGH compression.
- Load data into the uncompressed table partitions using a conventional load and then, once data is loaded using a CTAS operation, load into a temporary heap table. Then truncate the original partition. Alter the original partition into HCC compressed and then partition exchange this with the temporary heap table.

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

### Non- Exadata Hardware

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

### CLOB Fields

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

### Compression Guidelines

- Admin and Metadata tables and their indexes will NOT be compressed.
- All Transactional Tables will be compressed. This includes ILM enabled MOs where applicable.
- Compression will be done at the tablespace level.
  - Different MOs will have different tablespaces.
  - Partitioned MOs will have one tablespace per partition.
  - Child tables will use reference partitioning with parent + children sharing the same tablespace. (parent and child will always be managed/archived together).
- All multicolumn indexes on transactional/ILM 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
/
```

## Information Lifecycle Management (ILM) and Data Archiving Support

The product supports Data Archiving based on Information Lifecycle Management (ILM). If Information Lifecycle Management is part of your implementation, refer to the [Information Lifecycle Management and Data Archiving in SGG](#) chapter for instructions on partitioning objects when using ILM.

## 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.

## ILM Enabled Tablespace Requirements

- One tablespace for each partition of Parent table (Child table is referenced partitioned and would inherit the tablespace from Parent partition)
- One tablespace for each MO's global indexes (including child tables indexes) and \_K table

## 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)
 CONSTRAINT <> PRIMARY KEY(...)
)
ORGANIZATION INDEX PCTTHRESHOLD 50 OVERFLOW
LOB(COLUMN2) STORE AS SECUREFILE (ENABLE STORAGE IN ROW CHUNK CACHE
COMPRESS);
```

## Database Initialization Parameters

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

db\_block\_size=8192

log\_checkpoint\_interval=0

db\_file\_multiblock\_read\_count=8

transactions=3000

open\_cursors=3000

db\_writer\_processes=10

db\_files=1024

dbwr\_io\_slaves=10 (Only if Asynchronous IO is not Supported)

sessions=4500

memory\_target=0

memory\_max\_target=0

processes=3000

dml\_locks=48600

\_b\_tree\_bitmap\_plans=FALSE

# Oracle Database Implementation Guidelines

This section provides specific guidelines for implementing the Oracle database.

## Oracle Partitioning

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

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

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

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

## Database Statistic

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

A sample syntax that can be used is as following:

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

## Materialized View

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

### Prerequisites

Make sure to set up the following:

1. Set parameter QUERY\_REWRITE\_ENABLED=TRUE at database level.

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

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

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

To debug materialized views, refer the below URLs:

- Oracle 11g - [https://docs.oracle.com/cd/B28359\\_01/server.111/b28313/qrbasic.htm](https://docs.oracle.com/cd/B28359_01/server.111/b28313/qrbasic.htm)
- Oracle 12c - <https://docs.oracle.com/database/121/DWHSG/qrbasic.htm#DWHSG01813>
- Troubleshoot Materialized View - [http://docs.oracle.com/database/121/ARPLS/d\\_mview.htm#ARPLS67193](http://docs.oracle.com/database/121/ARPLS/d_mview.htm#ARPLS67193)

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

```
create synonym SPLUSR.F1_BO_LIFECYCLE_STATUS_MVW for
SPLADM.F1_BO_LIFECYCLE_STATUS_MVW;
```

```
grant select on F1_BO_LIFECYCLE_STATUS_MVW to FW_DEV;
```

---

```
grant select on F1_BO_LIFECYCLE_STATUS_MVW to SPL_USER;
grant select on F1_BO_LIFECYCLE_STATUS_MVW to SPL_READ;
```

For more information, refer to the following documents:

- Basic Query Rewrite (Oracle 11g) - [https://docs.oracle.com/cd/B28359\\_01/server.111/b28313/qrbasic.htm](https://docs.oracle.com/cd/B28359_01/server.111/b28313/qrbasic.htm)
- Basic Query Rewrite for Materialized Views (Oracle 12c) - <https://docs.oracle.com/database/121/DWHSG/qrbasic.htm#DWHSG01813>
- Troubleshooting Materialized Views - [http://docs.oracle.com/database/121/ARPLS/d\\_mview.htm#ARPLS67193](http://docs.oracle.com/database/121/ARPLS/d_mview.htm#ARPLS67193)
- Debugging materialized Views - [http://docs.oracle.com/cd/B28359\\_01/server.111/b28313/qradv.htm](http://docs.oracle.com/cd/B28359_01/server.111/b28313/qradv.htm)

### Known Issues

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

- Query Did Not Rewrite For A User Other Than The Owner Of the Materialized View (Doc ID 1594725.1) - A patch is available for bug report 14772096 for some platforms.
- Query rewrite not working as expected with SELECT DISTINCT (Doc ID 7661113.8) for Oracle version – 11.2.0.1 and 11.1.0.7 Fixed in version - 12.1.0.1 (Base Release), 11.2.0.2 (Server Patch Set)



# Chapter 5

---

## Conversion Tools

This chapter describes the following database conversion tools:

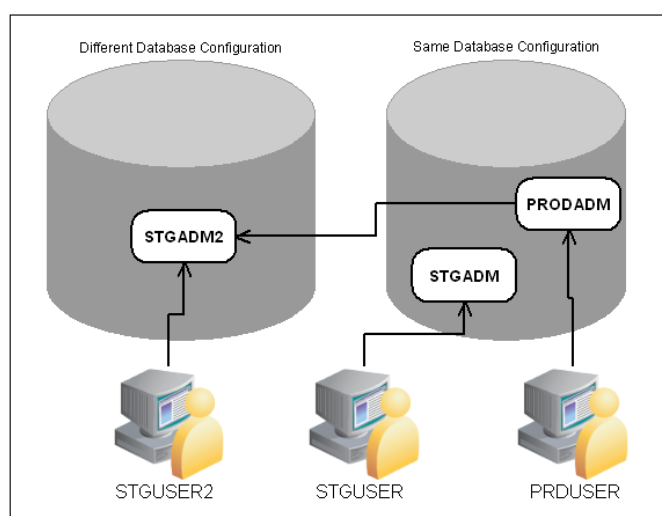
- [Database Configuration](#)
- [Installing the Script](#)
- [Preparing the Production Database](#)
- [Preparing the Staging Database](#)

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. Install Oracle 19c Client on the Windows desktop and configure SQLNet to connect to the target database. The Conversion folder includes the conversion setup utilities - ConvSetup.tar, scripts to create the users, and jarfiles.

### ConversionSetup Java Usage

|                          |                                                                                                                          |
|--------------------------|--------------------------------------------------------------------------------------------------------------------------|
| -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.                         |

|                                          |                                                                                                    |
|------------------------------------------|----------------------------------------------------------------------------------------------------|
| <code>-p,--prodSchema &lt;arg&gt;</code> | Primary/Production Schema and credentials to connect: CIS_ADMIN,CIS_PSWD. [example: CISADM,CISADM] |
| <code>-r,--rollback</code>               | Revert Conversion setup on Staging Schema. '-s' is mandatory if '-r' is passed                     |
| <code>-s,--stgSchema &lt;arg&gt;</code>  | 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).

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.

### UNIX:

```
export JAVA_HOME=/scratch/software/jdk1.8.0_102/
export PATH=$JAVA_HOME/bin:$PATH
export CLASSPATH=../FW-V4.4.0.3.0-Oracle-Database-Multiplatform/
FW/jarfiles/*
```

### WINDOWS:

```
SET JAVA_HOME=C:\Program Files\Java\jdk1.8.0_101
SET PATH=%JAVA_HOME%\bin;%PATH%
SET CLASSPATH= C:\FW-V4.4.0.3.0-Oracle-Database-
Multiplatform\FW\jarfiles*
```

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

### 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.

```
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}
```

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.

```
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}
```

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

```
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}
```

#### 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.

```
"%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}%
```

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.

```
"%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}%
```

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

```
"%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}%
```

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

|                      |                                                                           |
|----------------------|---------------------------------------------------------------------------|
| 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.

#### UNIX:

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

#### WINDOWS:

```
"%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>,< STAGING_R_USER> -r <STAGING
RW_USER_ROLE>,<STAGING_R_USER_ROLE> -a A -p
<STAGING_RW_USERPASS>,<STAGING_R_USERPASS>
```

# Chapter 6

---

## Information Lifecycle Management and Data Archiving in SGG

Oracle Utilities Smart Grid Gateway provides support for Information Lifecycle Management (ILM) and Data Archiving.

ILM is process to address data management issues, with a combination of processes, policies, software and hardware so that the appropriate technology can be used for each phase of the lifecycle of the data. The lifecycle of data typically refers to the fact that the most recent data is active in the system and as time passes the data is accessed less frequently or not at all. The costs of storing data that are accessed infrequently can be reduced by moving the data to lower cost mass storage media. Typically this involves a trade-off between cost and increased access times. Based on business needs, data may eventually be archived and purged from the database and kept offline ready to be restored if required.

This chapter includes:

- [ILM Implementation Overview](#)
- [ILM Implementation Components](#)
- [ILM Database Administrator's Tasks](#)

# ILM Implementation Overview

The implementation of ILM for products based on Oracle Utilities Application Framework includes a combination of application and database configuration and requires Oracle Partitioning.

An underlying design principle of the Oracle Utilities Application Framework ILM implementation is the concept that the age of the data may not be the only criterion used to determine when a record is able to be archived. There may be business rules that dictate that some records are still current and must not be archived yet.

ILM enabled objects have a combination of an ILM date and an ILM Archive Switch. The ILM date is used in conjunction with partitioning to group data by age. The ILM Archive Switch is set by a background process when the record meets the business rules specific to that Maintenance Object if the record is eligible to be archived. The ILM Archive Switch gives Database Administrators an easy method to check when all records in a partition meet the business criteria that make the partition eligible to be archived. If the ILM Archive Switch is set for all records, then the DBA can take the steps required to archive the partition.

Moving data between storage tiers takes advantage of the partitioning by ILM Date but does not require that the ILM Archive Switch is set. Oracle recommends using the Oracle Database ILM Assistant to assist with this process.

## ILM Implementation Components

The ILM based solution contains a number of components.

- ILM Specific Table Columns - For any Maintenance Object (MO) that has been configured to support ILM, the primary table of the MO includes two columns: ILM Date and ILM Archive Switch.
  - ILM\_DT - This date column is defaulted to an appropriate date (typically the system date) when a new record is inserted, the MO is partitioned on the ILM\_DT, so it should only be updated in exceptional circumstances as this would cause the record to be deleted from its current partition and inserted into a different partition, which is a relatively expensive operation.
  - ILM\_ARCHIVE\_SW - This field is set to N (Not yet eligible for archiving) when a new record is inserted. Subsequent reviews of "old" records may assess the data and change the value to "Y" based on business rules indicating that the record is eligible to be archived.
- Database Referential Integrity Constraints - These are required for reference partitioning of Child tables of ILM enabled MOs
- Partitioning - Partitioning is mandatory for ILM implementation. It is used to separate the data by ILM date so that data of a similar age is kept together.
- One Tablespace per Partition - The ILM implementation requires that each MO partition resides in a dedicated tablespace so that they can be easily managed.
- [Naming Convention](#) - This section covers the recommended naming convention to be used for partitions/subpartitions and tablespaces.

# ILM Database Administrator's Tasks

For a database administrator, there are two key phases involved with managing your data using ILM.

- [Preparation Phase](#) - This phase covers the database level configuration that needs to be done before the ILM solution runs in a production environment.
- [On-going Maintenance Phase](#) - This phase covers the ongoing maintenance tasks.

## Preparation Phase

Note: In order to successfully implement ILM as described here, the following DB Version and Patch are pre-requisites: database version 12.1.0.2.0 Enterprise Edition and Patch 15996848.

The steps needed to enable ILM functionality differ depending on whether ILM is enabled as part of the initial implementation of the product or enabled ILM on an existing implementation where data already exists in the respective tables.

- **Initial Install** – For an initial installation, the section [Module Specific ILM Implementation Details](#) outlines the additional steps to be performed on base delivered ILM Enabled Tables to conform to ILM requirements. In addition, [Appendix A: Sample SQL for Enabling ILM in SGG \(Initial Installation\)](#) provides sample reference DDLs using two maintenance objects as examples.
  - **Transform NON-ILM implementation to ILM Enabled Implementation:** The following steps provide a high level overview of steps that must be performed to implement ILM on enabled MOs for an existing implementation. Please refer to the [Appendix B: Sample SQL for Enabling ILM in SGG \(Existing Installation\)](#) section for detailed information using To Do Entry as an example. Also refer to [Appendix C: Sample SQL for Enabling ILM with Sub Retention in SGG \(Existing Installation\)](#) or detailed information using D1\_INIT\_MSRMT\_DATA as an example.
1. Rename the existing tables (Parent table followed by child table), and primary key index associated with ILM enabled MOs by renaming the tables.
  2. Save the DDLs for the secondary indexes as you will need to recreate them later.
  3. Drop secondary indexes on the renamed tables.
  4. Create Partitioned table with no secondary indexes for ILM enabled MOs using a CTAS operation (Create Table as Select), which will also load the data into the partitioned table structure.

**Functional Note:** ILM enabled MOs should have the ILM date (ILM\_DT) populated when data is moved into the new partitioned table. Please refer to the [Module Specific ILM Implementation Details](#) section below for initial load details on which date column to use as the basis for populating the ILM date. Often it is based on Create Date (CRE\_DTTM). ILM\_ARCH\_SW should initially be set to 'N'.

**Note:** Certain ILM enabled MOs, specifically IMD, Device Event, and Activity, support more than one retention period also known as sub retention periods. For these MOs the table will be sub-partitioned based



on the retention period. Furthermore, a more detailed approach will be required to set both the ILM date (ILM\_DT) and the retention period (<field name>). If your implementation does not wish to leverage the ability to define multiple retention periods for these MOs, this note can be ignored and the general guidelines for ILM enablement can be followed. If your implementation wishes to leverage the multiple retention period capability, refer to the [Module Specific ILM Implementation Details For Sub Retention](#) section.

5. Enable logging option.
6. Create Primary Key index.
7. Create Primary Key Constraint of parent table.
8. Create secondary indexes for the newly-created partitioned tables. This includes creating an index used specifically to benefit the ILM Crawler batch. The recommendation for this index name is to prefix it with "ILM".

**Note:** This can be created specifying parallel index create; remember to turn off parallelism after the index is created.

9. Follow a similar operation for all child tables for this MO, such as rename child table, and primary key index, generate DDL for secondary index, drop secondary index etc. Sample DDL for child tables their partitioning and indexes can be found in [Appendix B: Sample SQL for Enabling ILM in SGG \(Existing Installation\)](#). If sub retention is supported, sample DDL for child tables can be found in [Appendix C: Sample SQL for Enabling ILM with Sub Retention in SGG \(Existing Installation\)](#). Note that child table should be partitioned using reference partitioning of the parent table's partitioning key.
10. Drop the original, renamed tables after verifying the newly created partitioned tables.
11. If sub-retention is not supported, create the ILM specific indexes from section [Module Specific ILM Implementation Details](#).

| Table Name            | Index Name      |
|-----------------------|-----------------|
| CI_TD_ENTRY           | CM_ILM_XT039S8  |
| D1_ACTIVITY           | CM_ILM_D1T319S1 |
| D1_COMM_IN            | CM_ILM_D1T386S1 |
| D1_COMM_OUT           | CM_ILM_D1T380S1 |
| D1_COMPL_EVT          | CM_ILM_D1T340S1 |
| D1_DVC_EVT            | CM_ILM_D1T400S4 |
| D1_INIT_MSRMT_DATA    | CM_ILM_D1T304S4 |
| D1_USAGE              | CM_ILM_D1T281S2 |
| D1_USAGE_EXCP         | CM_ILM_D1T443S1 |
| D1_VEE_EXCP           | CM_ILM_D1T308S2 |
| D1_SNAPSHOT_DL_CTRL   | CM_ILM_D1T433S1 |
| D1_SP_SNAP_DL         | CM_ILM_D1T434S1 |
| D1_SP_UNR_USG_SNAP_DL | CM_ILM_D1T438S1 |

| Table Name             | Index Name      |
|------------------------|-----------------|
| D1_SP_USG_SNAP_DL      | CM_ILM_D1T436S1 |
| D1_SP_VEE_EXCP_SNAP_DL | CM_ILM_D1T440S1 |
| F1_BUS_FLG             | CM_ILM_F1T681S2 |
| F1_ERASURE_SCHED       | CM_ILM_F1T756S1 |
| F1_OBJ_REV             | CM_ILM_FT035S6  |
| F1_OUTMSG              | CM_ILM_FT010S2  |
| F1_PROC_STORE          | CM_ILM_F1T747S1 |
| F1_REMOTE_MSG          | CM_ILM_F1T735S7 |
| F1_STATS_SNPST         | CM_ILM_F1C706S2 |
| F1_SVC_TASK            | CM_ILM_F1C474S3 |
| F1_SYNC_REQ            | CM_ILM_F1T014S4 |
| F1_SYNC_REQ_IN         | CM_ILM_F1T191S3 |

12. If sub-retention is supported, create the following ILM specific indexes from the [Module Specific ILM Implementation Details](#) section:

| Table Name             | Index Name      |
|------------------------|-----------------|
| CI_TD_ENTRY            | CM_ILM_XT039S8  |
| D1_COMM_IN             | CM_ILM_D1T386S1 |
| D1_COMM_OUT            | CM_ILM_D1T380S1 |
| D1_COMPL_EVT           | CM_ILM_D1T340S1 |
| D1_USAGE               | CM_ILM_D1T281S2 |
| D1_USAGE_EXCP          | CM_ILM_D1T443S1 |
| D1_VEE_EXCP            | CM_ILM_D1T308S2 |
| D1_SNAPSHOT_DL_CTRL    | CM_ILM_D1T433S1 |
| D1_SP_SNAP_DL          | CM_ILM_D1T434S1 |
| D1_SP_UNR_USG_SNAP_DL  | CM_ILM_D1T438S1 |
| D1_SP_USG_SNAP_DL      | CM_ILM_D1T436S1 |
| D1_SP_VEE_EXCP_SNAP_DL | CM_ILM_D1T440S1 |
| F1_BUS_FLG             | CM_ILM_F1T681S2 |
| F1_ERASURE_SCHED       | CM_ILM_F1T756S1 |
| F1_OBJ_REV             | CM_ILM_FT035S6  |
| F1_OUTMSG              | CM_ILM_FT010S2  |
| F1_PROC_STORE          | CM_ILM_F1T747S1 |

| Table Name      | Index Name      |
|-----------------|-----------------|
| F1_REMOTE_MSG   | CM_ILM_F1T735S7 |
| F1_STATS_SNPSHT | CM_ILM_F1C706S2 |
| F1_SVC_TASK     | CM_ILM_F1C474S3 |
| F1_SYNC_REQ     | CM_ILM_F1T014S4 |
| F1_SYNC_REQ_IN  | CM_ILM_F1T191S3 |

and the ILM subretention specific indexes from the [Module Specific ILM Implementation Details For Sub Retention](#) section:

| Table Name         | Index Name      |
|--------------------|-----------------|
| D1_ACTIVITY        | CM_ILM_D1T319S1 |
| D1_DVC_EVT         | CM_ILM_D1T400S4 |
| D1_INIT_MSRMT_DATA | CM_ILM_D1T304S4 |

## Module Specific ILM Implementation Details

This section outlines each maintenance object that has been configured to support ILM. The parent table is noted. Other tables are child tables of the parent unless otherwise noted. In each case, the partitioning strategy is indicated.

All indexes are listed with a recommendation whether the index should be global or local and whether the index should be partitioned. In addition to the base delivered indexes, each parent table includes a recommended ILM specific local index to build with the ILM\_DT, ILM\_ARCH\_SW and the primary key of the table. The recommended column that should be used to populate the ILM\_DT is also shown.

This section details the following maintenance objects:

- [To Do Entry](#)
- [Sync Request \(Outbound\)](#)
- [Inbound Sync Request](#)
- [Outbound Message](#)
- [Service Task](#)
- [Object Revision](#)
- [Business Flag](#)
- [Remote Message](#)
- [Statistics Snapshot](#)
- [Object Erasure](#)
- [Process Flow](#)
- [Activity](#)
- [Communication In](#)
- [Communication Out](#)

- [Device Event](#)
- [Completion Event](#)
- [Initial Measurement Data](#)
- [Usage Transaction](#)
- [Usage Transaction Exception](#)
- [VEE Exception](#)
- [Snapshot Tables](#)

### To Do Entry

This table describes the To Do Entry maintenance object.

| Table Name           | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name     | Index Columns                                          | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load   |
|----------------------|--------------------------------------------------------------|----------------|--------------------------------------------------------|----------------------------|-----------------------------------------|-----------------------|
| CI_TD_ENTRY (Parent) | RANGE (ILM_DT, TD_ENTRY_ID)                                  |                |                                                        |                            |                                         | CI_TD_ENTRY, CRE_DTTM |
|                      |                                                              | XT039P0        | TD_ENTRY_ID                                            | Global Partitioned         | RANGE (TD_ENTRY_ID)                     |                       |
|                      |                                                              | XT039S2        | ASSIGNED_TO, TD_ENTRY_ID                               | Global                     |                                         |                       |
|                      |                                                              | XT039S3        | ENTRY_STATUS_FLG, ASSIGNED_TO                          | Global                     |                                         |                       |
|                      |                                                              | XT039S4        | ROLE_ID, TD_TYPE_CD, ENTRY_STATUS_FLG, TD_PRIORITY_FLG | Global                     |                                         |                       |
|                      |                                                              | XT039S5        | BATCH_CD, BATCH_NBR, ENTRY_STATUS_FLG                  | Global                     |                                         |                       |
|                      |                                                              | XT039S6        | TD_ENTRY_ID, ASSIGNED_TO, ENTRY_STATUS_FLG             | Global                     |                                         |                       |
|                      |                                                              | XT039S7        | COMPLETE_USER_ID, COMPLETE_DTTM, TD_ENTRY_ID           | Global                     |                                         |                       |
|                      |                                                              | CM_ILM_XT039S8 | ILM_DT, ILM_ARCH_SW, TD_ENTRY_ID                       | Local Partitioned          |                                         |                       |
| CI_TD_ENTRY_CHA      | Reference Partitioning                                       | XT701P0        | TD_ENTRY_ID, CHAR_TYPE_CD, SEQ_NUM                     | Global Partitioned         |                                         |                       |
|                      |                                                              | XT701S1        | SRCH_CHAR_VAL, CHAR_TYPE_CD, TD_ENTRY_ID               | Global                     |                                         |                       |

| Table Name     | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name | Index Columns                               | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load |
|----------------|--------------------------------------------------------------|------------|---------------------------------------------|----------------------------|-----------------------------------------|---------------------|
|                |                                                              | XT701S2    | CHAR_VAL_FK1                                | Global                     |                                         |                     |
| CI_TD_DRLKEY   | Reference Partitioning                                       | XT037P0    | TD_ENTRY_ID, SEQ_NUM                        | Global Partitioned         |                                         |                     |
|                |                                                              | XT037S1    | KEY_VALUE, TD_ENTRY_ID                      | Global                     |                                         |                     |
| CI_TD_LOG      | Reference Partitioning                                       | XT721P0    | TD_ENTRY_ID, SEQ_NUM                        | Global Partitioned         |                                         |                     |
|                |                                                              | XT721S1    | LOG_DTTM,USER_ID, LOG_TYPE_FLG, TD_ENTRY_ID | Global                     |                                         |                     |
| CI_TD_MSG_PARM | Reference Partitioning                                       | XT040P0    | TD_ENTRY_ID, SEQ_NUM                        | Global                     |                                         |                     |
| CI_TD_SRTKEY   | Reference Partitioning                                       | XT041P0    | TD_ENTRY_ID, SEQ_NUM                        | Global Partitioned         |                                         |                     |
|                |                                                              | XT041S1    | KEY_VALUE, TD_ENTRY_ID                      | Global                     |                                         |                     |

### Sync Request (Outbound)

This table describes the Sync Request (Outbound) maintenance object.

| Table Name           | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name | Index Columns                                      | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load   |
|----------------------|--------------------------------------------------------------|------------|----------------------------------------------------|----------------------------|-----------------------------------------|-----------------------|
| F1_SYNC_REQ (Parent) | RANGE (ILM_DT, F1_SYNC_REQ_ID)                               |            |                                                    |                            | RANGE (F1_SYNC_REQ_ID)                  | F1_SYNC_REQ.C RE_DTTM |
|                      |                                                              | F1T014P0   | F1_SYNC_REQ_ID                                     | Global Partitioned         |                                         |                       |
|                      |                                                              | F1T014S1   | BO_STATUS_CD, BUS_OBJ_CD, F1_SYNC_REQ_ID           | Global                     |                                         |                       |
|                      |                                                              | F1T014S2   | BO_STATUS_REASON_CD                                | Global                     |                                         |                       |
|                      |                                                              | F1T014S3   | MAINT_OBJ_CD, PK_VALUE1, PK_VALUE2, F1_SYNC_REQ_ID | Global                     |                                         |                       |

| Table Name               | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name          | Index Columns                               | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load |
|--------------------------|--------------------------------------------------------------|---------------------|---------------------------------------------|----------------------------|-----------------------------------------|---------------------|
|                          |                                                              | CM_ILM_<br>F1T014S4 | ILM_DT,<br>ILM_ARC_SW,<br>F1_SYNC_REQ_ID    | Local<br>Partitioned       |                                         |                     |
| F1_SYNC_REQ_<br>CHAR     | Reference<br>Partitioning                                    | F1T017P0            | F1_SYNC_REQ_ID,<br>CHAR_TYPE_CD,<br>SEQ_NUM | Global<br>Partitioned      |                                         |                     |
|                          |                                                              | F1T017S1            | SRCH_CHAR_VAL                               | Global                     |                                         |                     |
| F1_SYNC_REQ_<br>EXTRACT  | Reference<br>Partitioning                                    | F1T019P0            | F1_SYNC_REQ_ID,<br>SEQ_NUM                  | Global<br>Partitioned      |                                         |                     |
| F1_SYNC_REQ_<br>LOG      | Reference<br>Partitioning                                    | F1T015P0            | F1_SYNC_REQ_ID,<br>SEQNO                    | Global<br>Partitioned      |                                         |                     |
|                          |                                                              | F1T015S1            | CHAR_TYPE_CD,<br>CHAR_VAL_FK1               | Global                     |                                         |                     |
|                          |                                                              | F1T015S2            | CHAR_TYPE_CD,<br>CHAR_VAL                   | Global                     |                                         |                     |
|                          |                                                              | F1T015S3            | BO_STATUS_REAS<br>ON_CD                     | Global                     |                                         |                     |
| F1_SYNC_REQ_<br>LOG_PARM | Reference<br>Partitioning                                    | F1T016P0            | F1_SYNC_REQ_ID,<br>SEQNO,<br>PARM_SEQ       | Global<br>Partitioned      |                                         |                     |

**Note:** It is recommended that data retention policies and rules for this object match the policies and rules implemented for the Inbound Sync Request on the target system to avoid data inconsistencies when auditing.

### Inbound Sync Request

This table describes the Inbound Sync Request maintenance object.

| Table Name                  | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name | Index Columns | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load         |
|-----------------------------|--------------------------------------------------------------|------------|---------------|----------------------------|-----------------------------------------|-----------------------------|
| F1_SYNC_REQ_<br>IN (Parent) | RANGE(ILM_DT,<br>F1_SYNC_REQ_I<br>N_ID)                      |            |               |                            | RANGE<br>(F1_SYNC_REQ_<br>IN_ID)        | F1_SYNC_REQ_I<br>N.CRE_DTTM |

| Table Name               | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name      | Index Columns                                     | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load |
|--------------------------|--------------------------------------------------------------|-----------------|---------------------------------------------------|----------------------------|-----------------------------------------|---------------------|
|                          |                                                              | F1T191P0        | F1_SYNC_REQ_IN_ID                                 | Global Partitioned         |                                         |                     |
|                          |                                                              | F1T191S1        | BO_STATUS_CD, BUS_OBJ_CD, F1_SYNC_REQ_IN_ID       | Global                     |                                         |                     |
|                          |                                                              | F1T191S2        | MAINT_OBJ_CD, EXT_PK_VALUE1, NT_XID_CD, PK_VALUE1 | Global                     |                                         |                     |
|                          |                                                              | CM_ILM_F1T191S3 | ILM_DT, ILM_ARCH_SW, F1_SYNC_REQ_IN_ID            | Local Partitioned          |                                         |                     |
| F1_SYNC_REQ_IN_CHAR      | Reference Partitioning                                       | F1T193P0        | F1_SYNC_REQ_IN_ID, CHAR_TYPE_CD, SEQ_NUM          | Global Partitioned         |                                         |                     |
|                          |                                                              | F1T193S1        | SRCH_CHAR_VAL                                     | Global                     |                                         |                     |
| F1_SYNC_REQ_IN_EXCP      | Reference Partitioning                                       | F1T197P0        | F1_SYNC_REQ_IN_ID, SEQNO                          | Global Partitioned         |                                         |                     |
| F1_SYNC_REQ_IN_EXCP_PARM | Reference Partitioning                                       | F1T198P0        | F1_SYNC_REQ_IN_ID, SEQNO, PARM_SEQ                | Global Partitioned         |                                         |                     |
| F1_SYNC_REQ_IN_LOG       | Reference Partitioning                                       | F1T194P0        | F1_SYNC_REQ_IN_ID, SEQNO                          | Global Partitioned         |                                         |                     |
|                          |                                                              | F1T194S1        | CHAR_TYPE_CD, CHAR_VAL_FK1                        | Global                     |                                         |                     |
|                          |                                                              | F1T194S2        | CHAR_TYPE_CD, CHAR_VAL                            | Global                     |                                         |                     |
| F1_SYNC_REQ_IN_LOG_PARM  | Reference Partitioning                                       | F1T195P0        | F1_SYNC_REQ_IN_ID, SEQNO, PARM_SEQ                | Global Partitioned         |                                         |                     |
| F1_SYNC_REQ_IN_REL_OBJ   | Reference Partitioning                                       | F1T192P0        | F1_SYNC_REQ_IN_ID, MAINT_OBJ_CD, REL_OBJ_TYPE_FLG | Global Partitioned         |                                         |                     |

**Note:** It is recommended that data retention policies and rules for this object match the policies and rules implemented for the Outbound Sync

Request on the source system to avoid data inconsistencies when auditing.

### Outbound Message

This table describes the Outbound Message maintenance object.

| Table Name         | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name         | Index Columns                                | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load    |
|--------------------|--------------------------------------------------------------|--------------------|----------------------------------------------|----------------------------|-----------------------------------------|------------------------|
| F1_OUTMSG (Parent) | RANGE (ILM_DT, OUTMSG_ID)                                    |                    |                                              |                            | RANGE (OUMSG_ID)                        | F1_OUTMSG.<br>CRE_DTTM |
|                    |                                                              | FT010P0            | OUTMSG_ID                                    | Global Partitioned         |                                         |                        |
|                    |                                                              | FT010S1            | OUTMSG_STAT<br>US_FLG,<br>OUTMSG_TYPE<br>_CD | Global                     |                                         |                        |
|                    |                                                              | CM_ILM_<br>FT010S2 | ILM_DT,<br>ILM_ARC_SW,<br>OUTMSG_ID          | Local Partitioned          |                                         |                        |
| F1_OUTMSG_ERRPARM  | Reference Partitioning                                       | FT011P0            | OUTMSG_ID,<br>PARM_SEQ                       | Global Partitioned         |                                         |                        |

### Service Task

This table describes the Service Task maintenance object.

| Table Name           | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name          | Index Columns                            | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load      |
|----------------------|--------------------------------------------------------------|---------------------|------------------------------------------|----------------------------|-----------------------------------------|--------------------------|
| F1_SVC_TASK (Parent) | RANGE (ILM_DT, F1_SVC_TASK_ID)                               |                     |                                          |                            | RANGE (F1_SVC_TASK_ID)                  | F1_SVC_TASK.<br>CRE_DTTM |
|                      |                                                              | F1C474P0            | F1_SVC_TASK_ID                           | Global Partitioned         |                                         |                          |
|                      |                                                              | F1C474S1            | F1_STASK_TYPE_CD                         | Global                     |                                         |                          |
|                      |                                                              | F1C474S2            | BUS_OBJ_CD                               | Global                     |                                         |                          |
|                      |                                                              | CM_ILM_<br>F1C474S2 | ILM_DT,<br>ILM_ARC_SW,<br>F1_SVC_TASK_ID | Local Partitioned          |                                         |                          |



| Table Name           | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name | Index Columns                                 | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load |
|----------------------|--------------------------------------------------------------|------------|-----------------------------------------------|----------------------------|-----------------------------------------|---------------------|
| F1_SVC_TASK_CHAR     | Reference Partitioning                                       | F1C476P0   | F1_SVC_TASK_ID, CHAR_TYPE_CD, SEQ_NUM         | Global Partitioned         |                                         |                     |
|                      |                                                              | F1C476S1   | SRCH_CHAR_VAL                                 | Global                     |                                         |                     |
| F1_SVC_TASK_LOG      | Reference Partitioning                                       | F1C477P0   | F1_SVC_TASK_ID, SEQNO                         | Global Partitioned         |                                         |                     |
|                      |                                                              | F1C477S1   | CHAR_TYPE_CD, CHAR_VAL_FK1                    | Global                     |                                         |                     |
|                      |                                                              | F1C477S2   | CHAR_TYPE_CD, CHAR_VAL                        | Global                     |                                         |                     |
| F1_SVC_TASK_LOG_PARM | Reference Partitioning                                       | F1C478P0   | F1_SVC_TASK_ID, SEQNO, PARM_SEQ               | Global Partitioned         |                                         |                     |
| F1_SVC_TASK_REL_OBJ  | Reference Partitioning                                       | F1C479P0   | F1_SVC_TASK_ID, MAINT_OBJ_CD, SEQ_NUM         | Global Partitioned         |                                         |                     |
|                      |                                                              | F1C479S1   | MAINT_OBJ_CD, PK_VALUE1, PK_VALUE2, PK_VALUE3 | Global                     |                                         |                     |

### Object Revision

This table describes the Object Revision maintenance object.

| Table Name          | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name | Index Columns | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load           |
|---------------------|--------------------------------------------------------------|------------|---------------|----------------------------|-----------------------------------------|-------------------------------|
| F1_OBJ_REV (Parent) | RANGE (ILM_DT, REV_ID)                                       |            |               |                            | RANGE (REV_ID)                          | F1_OBJ_REV, STATUS_UPD_D, TTM |
|                     |                                                              | FT035P0    | REV_ID        | Global Partitioned         |                                         |                               |

| Table Name              | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name         | Index Columns                          | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load |
|-------------------------|--------------------------------------------------------------|--------------------|----------------------------------------|----------------------------|-----------------------------------------|---------------------|
|                         |                                                              | FT035S1            | BO_STATUS_CD,<br>BUS_OBJ_CD,<br>REV_ID | Global                     |                                         |                     |
|                         |                                                              | FT035S2            | MAINT_OBJ_CD,<br>PK_VALUE1             | Global                     |                                         |                     |
|                         |                                                              | FT035S3            | EXT_REFERENCE_ID,<br>MAINT_OBJ_CD      | Global                     |                                         |                     |
|                         |                                                              | FT035S4            | USER_ID,<br>MAINT_OBJ_CD               | Global                     |                                         |                     |
|                         |                                                              | FT035S5            | PK_VALUE1                              | Global                     |                                         |                     |
|                         |                                                              | CM_ILM_<br>FT035S6 | ILM_DT,<br>ILM_ARC_SW,<br>REV_ID       | Local<br>Partitioned       |                                         |                     |
| F1_OBJ_REV_<br>CHAR     | Reference<br>Partitioning                                    | FT037P0            | REV_ID,<br>CHAR_TYPE_CD,<br>SEQ_NUM    | Global<br>Partitioned      |                                         |                     |
|                         |                                                              | FT037S1            | SRCH_CHAR_VAL                          | Global                     |                                         |                     |
| F1_OBJ_REV_<br>LOG      | Reference<br>Partitioning                                    | FT039P0            | REV_ID, SEQNO                          | Global<br>Partitioned      |                                         |                     |
| F1_OBJ_REV_<br>LOG_PARM | Reference<br>Partitioning                                    | FT040P0            | REV_ID, SEQNO,<br>PARM_SEQ             | Global<br>Partitioned      |                                         |                     |

**Note:** This maintenance object is enabled for ILM, however it is not used in a production environment. It is typically used in a development or configuration environment. Your implementation should review its use of this functionality and consider whether or not it is a candidate for ILM and in which region.

**Business Flag**

This table describes the Business Flag maintenance object.

| Table Name          | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name          | Index Columns                                 | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load  |
|---------------------|--------------------------------------------------------------|---------------------|-----------------------------------------------|----------------------------|-----------------------------------------|----------------------|
| F1_BUS_FLG (Parent) | RANGE (ILM_DT,BUS_FLG_ID)                                    |                     |                                               |                            | RANGE(BUS_FLG_ID)                       | F1_BUS_FLG, CRE_DTTM |
|                     |                                                              | F1T681P0            | BUS_FLG_ID                                    | Global Partitioned         |                                         |                      |
|                     |                                                              | F1T681S1            | BUS_OBJ_CD, BO_STATUS_CD, BUS_FLG_ID          | Global                     |                                         |                      |
|                     |                                                              | CM_ILM_<br>F1T681S2 | ILM_DT, ILM_ARCH_SW, BUS_FLG_ID               | Local Partitioned          |                                         |                      |
| F1_BUS_FLG_CHAR     | Reference Partitioning                                       | F1T684P0            | BUS_FLG_ID, CHAR_TYPE_CD, SEQ_NUM             | Global Partitioned         |                                         |                      |
|                     |                                                              | F1T684S0            | SRCH_CHAR_VAL                                 | Global                     |                                         |                      |
| F1_BUS_FLG_LOG      | Reference Partitioning                                       | F1T685P0            | BUS_FLG_ID, SEQNO                             | Global Partitioned         |                                         |                      |
|                     |                                                              | F1T685S1            | CHAR_TYPE_CD, CHAR_VAL_FK1                    | Global                     |                                         |                      |
|                     |                                                              | F1T685S2            | CHAR_TYPE_CD, CHAR_VAL                        | Global                     |                                         |                      |
| F1_BUS_FLG_LOG_PARM | Reference Partitioning                                       | F1T686P0            | BUS_FLG_ID, SEQNO, PARM_SEQ                   | Global Partitioned         |                                         |                      |
| F1_BUS_FLG_REL      | Reference Partitioning                                       | F1T682P0            | BUS_FLG_ID, BUS_FLG_REL_TYPE_FLG, SEQ_NUM     | Global Partitioned         |                                         |                      |
| F1_BUS_FLG_REL_OBJ  | Reference Partitioning                                       | F1T683P0            | BUS_FLG_ID, BUS_FLG_REL_OBJ_TYPE_FLG, SEQ_NUM | Global Partitioned         |                                         |                      |

## Remote Message

This table describes the Remote Message maintenance object.

| Table Name             | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name      | Index Columns                                      | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load    |
|------------------------|--------------------------------------------------------------|-----------------|----------------------------------------------------|----------------------------|-----------------------------------------|------------------------|
| F1_REMOTE_MSG (Parent) | RANGE (ILM_DT,F1_REMOTE_MSG_ID)                              |                 |                                                    |                            | RANGE(F1_REMOTE_MSG_ID)                 | F1_REMOTE_MSG.CRE_DTTM |
|                        |                                                              | F1T735P0        | F1_REMOTE_MSG_ID                                   | Global Partitioned         |                                         |                        |
|                        |                                                              | F1T735S1        | CRE_DTTM                                           | Global                     |                                         |                        |
|                        |                                                              | F1T735S2        | F1_MDT_ID                                          | Global                     |                                         |                        |
|                        |                                                              | F1T735S3        | MAINT_OBJ_CD                                       | Global                     |                                         |                        |
|                        |                                                              | F1T735S4        | PK_VALUE1                                          | Global                     |                                         |                        |
|                        |                                                              | F1T735S5        | F1_DEVICE_MSG_ID                                   | Global                     |                                         |                        |
|                        |                                                              | F1T735S6        | F1_MDT_ID, F1_MSG_CLASS_FLG, F1_DELIVERY_STATE_FLG | Global                     |                                         |                        |
|                        |                                                              | CM_ILM_F1T735S7 | ILM_DT, ILM_ARCH_SW, F1_REMOTE_MSG_ID              | Local Partitioned          |                                         |                        |
| F1_REMOTE_MSG_CHAR     | Reference Partitioning                                       | F1T736P0        | F1_REMOTE_MSG_ID, CHAR_TYPE_CD, SEQ_NUM            | Global Partitioned         |                                         |                        |
|                        |                                                              | F1T736S1        | SRCH_CHAR_VAL                                      | Global                     |                                         |                        |
| F1_REMOTE_MSG_LOG      | Reference Partitioning                                       | F1T737P0        | F1_REMOTE_MSG_ID, SEQNO                            | Global Partitioned         |                                         |                        |
|                        |                                                              | F1T737S1        | CHAR_TYPE_CD, CHAR_VAL_FK1                         | Global                     |                                         |                        |
|                        |                                                              | F1T737S2        | CHAR_TYPE_CD, CHAR_VAL                             | Global                     |                                         |                        |
| F1_REMOTE_MSG_LOG_PARM | Reference Partitioning                                       | F1T738P0        | F1_REMOTE_MSG_ID, SEQNO, PARM_SEQ                  | Global Partitioned         |                                         |                        |

## Statistics Snapshot

This table describes the Statistics Snapshot maintenance object.

| Table Name              | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name      | Index Columns                                       | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load      |
|-------------------------|--------------------------------------------------------------|-----------------|-----------------------------------------------------|----------------------------|-----------------------------------------|--------------------------|
| F1_STATS_SNPSHT(Parent) | RANGE (ILM_DT, SNAPSHOT_ID)                                  |                 |                                                     |                            | RANGE (SNAPSHOT_ID)                     | F1_STATS_SNPSHT.CRE_DTTM |
|                         |                                                              | F1C706P0        | SNAPSHOT_ID                                         | Global Partitioned         |                                         |                          |
|                         |                                                              | F1C706S1        | BUS_OBJ_CD, BO_STATUS_CD, SNAPSHOT_ID               | Global                     |                                         |                          |
|                         |                                                              | CM_ILM_F1C706S2 | ILM_DT, ILM_ARCH_SW, SNAPSHOT_ID                    | Local Partitioned          |                                         |                          |
| F1_STATS_SNPSHT_CHAR    | Reference Partitioning                                       | F1C707P0        | SNAPSHOT_ID, CHAR_TYPE_CD, SEQ_NUM                  | Global Partitioned         |                                         |                          |
|                         |                                                              | F1C707S1        | SRCH_CHAR_VAL                                       | Global                     |                                         |                          |
| F1_STATS_SNPSHT_LOG     | Reference Partitioning                                       | F1C708P0        | SNAPSHOT_ID, SEQNO                                  | Global Partitioned         |                                         |                          |
|                         |                                                              | F1C708S1        | CHAR_TYPE_CD, CHAR_VAL_FK1                          | Global                     |                                         |                          |
|                         |                                                              | F1C708S2        | SNAPSHOT_ID, SEQNO, PARM_SEQ                        | Global Partitioned         |                                         |                          |
| F1_STATS_SNPSHT_REL_OBJ | Reference Partitioning                                       | F1C710P0        | SNAPSHOT_ID, STATS_SNPSHT_REL_OBJ_TYPE_FLG, SEQ_NUM | Global Partitioned         |                                         |                          |

## Object Erasure

This table describes the Object Erasure maintenance object.

| Table Name                        | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name          | Index Columns                                   | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load                      |
|-----------------------------------|--------------------------------------------------------------|---------------------|-------------------------------------------------|----------------------------|-----------------------------------------|------------------------------------------|
| F1_ERASURE_S<br>CHED (Parent)     | RANGE<br>(ILM_DT,<br>ERASURE_SCHE<br>D_ID)                   |                     |                                                 |                            |                                         | F1_ERASURE_S<br>CHED.STATUS_<br>UPD_DTTM |
|                                   |                                                              | F1T756P0            | ERASURE_SCHED<br>_ID                            | GLOBAL<br>Partitioned      | RANGE<br>(ERASURE_SCH<br>ED_ID)         |                                          |
|                                   |                                                              | CM_ILM_F1T756<br>S1 | ILM_DT,<br>ILM_ARCH_SW,<br>ERASURE_SCHED<br>_ID | LOCAL                      |                                         |                                          |
| F1_ERASURE_S<br>CHED_LOG          | Reference<br>partitioning                                    | F1T757P0            | ERASURE_SCHED<br>_ID,<br>SEQNO                  | GLOBAL<br>Partitioned      |                                         |                                          |
| F1_ERASURE_S<br>CHED_LOG_PA<br>RM | Reference<br>partitioning                                    | F1T758P0            | ERASURE_SCHED<br>_ID,<br>SEQNO,<br>PARM_SEQ     | GLOBAL<br>Partitioned      |                                         |                                          |

## Process Flow

This table describes the Process Flow maintenance object.

| Table Name                         | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name          | Index Columns                              | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load                   |
|------------------------------------|--------------------------------------------------------------|---------------------|--------------------------------------------|----------------------------|-----------------------------------------|---------------------------------------|
| F1_PROC_STOR<br>E (Parent)         | RANGE(ILM_<br>DT, PROC_<br>STORE_ID)                         |                     |                                            |                            |                                         | F1_PROC_STOR<br>E.STATUS_UPD_<br>DTTM |
|                                    |                                                              | F1T747P0            | PROC_STORE_ID                              | GLOBAL<br>Partitioned      | RANGE(PROC_<br>STORE_ID)                |                                       |
|                                    |                                                              | CM_ILM_F1T747<br>S1 | ILM_DT,<br>ILM_ARCH_SW,<br>PROC_STORE_ID   | LOCAL                      |                                         |                                       |
| F1_PROC_STOR<br>E_DTL_ELEME<br>NTS | Reference<br>partitioning                                    | F1T748P0            | PROC_STORE_ID,<br>CHAR_TYPE_CD,<br>SEQ_NUM | GLOBAL<br>Partitioned      |                                         |                                       |
| F1_PROC_<br>STORE_LOG              | Reference<br>partitioning                                    | F1T749P0            | PROC_STORE_ID,<br>SEQNO                    | GLOBAL<br>Partitioned      |                                         |                                       |

| Table Name                 | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name | Index Columns                       | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load |
|----------------------------|--------------------------------------------------------------|------------|-------------------------------------|----------------------------|-----------------------------------------|---------------------|
| F1_PROC_STOR<br>E_LOG_PARM | Reference partitioning                                       | F1T750P0   | PROC_STORE_ID<br>SEQNO,<br>PARM_SEQ | GLOBAL<br>Partitioned      |                                         |                     |

### Activity

If sub retention periods will be defined for this MO, then please follow the guidelines set forth in section [Module Specific ILM Implementation Details For Sub Retention](#).

This table describes the Activity maintenance object.

| Table Name                 | Table Partitioning Type (Partitioning, Sub-Partitioning Key)                                                                                            | Index Name          | Index Columns                                  | Index Type Global or Local | Index Partitioning Sub-Partitioning Key              | ILM_DT Initial Load      |
|----------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------|------------------------------------------------|----------------------------|------------------------------------------------------|--------------------------|
| D1_ACTIVITY<br>(Parent)    | RANGE (ILM_DT,<br>D1_ACTIVITY_ID)<br><br><b>Note:</b> Default is to use sub-retention or use RANGE (ILM_DT, D1_ACTIVITY_ID) if not using sub-retention. |                     |                                                |                            |                                                      | D1_ACTIVITY.<br>CRE_DTTM |
|                            |                                                                                                                                                         | D1T319P0            | D1_ACTIVITY_ID                                 | Global<br>Partitioned      | RANGE<br>(D1_ACTIVITY_ID)                            |                          |
|                            |                                                                                                                                                         | D1T319S0            | BUS_OBJ_CD,<br>BO_STATUS_CD,<br>D1_ACTIVITY_ID | Global<br>Partitioned      | HASH(BUS_OBJ_CD,<br>BO_STATUS_CD,<br>D1_ACTIVITY_ID) |                          |
|                            |                                                                                                                                                         | CM_ILM_<br>D1T319S1 | ILM_DT,<br>ILM_ARCH_SW,<br>D1_ACTIVITY_ID      | Local                      |                                                      |                          |
| D1_ACTIVITY_<br>CHAR       | REFERENCE<br>(D1_ACTIVITY_<br>CHAR_FK)                                                                                                                  |                     |                                                |                            |                                                      |                          |
|                            |                                                                                                                                                         | D1T320P0            | D1_ACTIVITY_ID,<br>CHAR_TYPE_CD,<br>SEQ_NUM    | Global<br>Partitioned      | RANGE(D1_<br>ACTIVITY_ID)                            |                          |
|                            |                                                                                                                                                         | D1T320S0            | SRCH_CHAR_VAL                                  | Global<br>Partitioned      | HASH(SRCH_CHAR<br>_VAL)                              |                          |
| D1_ACTIVITY_<br>IDENTIFIER | REFERENCE<br>(D1_ACTIVITY_ID<br>ENTIFIER_FK)                                                                                                            |                     |                                                |                            |                                                      |                          |

| Table Name           | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name | Index Columns                                                       | Index Type Global or Local | Index Partitioning Sub-Partitioning Key          | ILM_DT Initial Load |
|----------------------|--------------------------------------------------------------|------------|---------------------------------------------------------------------|----------------------------|--------------------------------------------------|---------------------|
|                      |                                                              | D1T330P0   | D1_ACTIVITY_ID, ACTIVITY_ID_TYPE_FLG                                | Global Partitioned         | RANGE(D1_ACTIVITY_ID)                            |                     |
|                      |                                                              | D1T330S0   | ACTIVITY_ID_TYPE_FLG, ID_VALUE                                      | Global Partitioned         | HASH(ACTIVITY_ID_TYPE_FLG, ID_VALUE)             |                     |
|                      |                                                              | D1T330S1   | ACTIVITY_ID_TYPE_FLG, UPPER(ID_VALUE)                               |                            |                                                  |                     |
| D1_ACTIVITY_LOG      | REFERENCE (D1_ACTIVITY_LOG_FK)                               |            |                                                                     |                            |                                                  |                     |
|                      |                                                              | D1T321P0   | D1_ACTIVITY_ID, SEQNO                                               | Global Partitioned         | RANGE(D1_ACTIVITY_ID)                            |                     |
|                      |                                                              | D1T321S1   | CHAR_TYPE_CD, CHAR_VAL_FK1                                          | Global Partitioned         | HASH(CHAR_TYPE_CD, CHAR_VAL_FK1)                 |                     |
|                      |                                                              | D1T321S2   | CHAR_TYPE_CD, CHAR_VAL                                              | Global Partitioned         | HASH(CHAR_TYPE_CD, CHAR_VAL)                     |                     |
| D1_ACTIVITY_LOG_PARM | REFERENCE (D1_ACTIVITY_LOG_PARM_FK)                          |            |                                                                     |                            |                                                  |                     |
|                      |                                                              | D1T322P0   | D1_ACTIVITY_ID, SEQNO, PARM_SEQ                                     | Global Partitioned         | RANGE(D1_ACTIVITY_ID)                            |                     |
| D1_ACTIVITY_REL      | REFERENCE (D1_ACTIVITY_REL_FK)                               |            |                                                                     |                            |                                                  |                     |
|                      |                                                              | D1T323P0   | D1_ACTIVITY_ID, ACTIVITY_REL_TYPE_FLG                               | Global Partitioned         | RANGE(D1_ACTIVITY_ID)                            |                     |
|                      |                                                              | D1T323S0   | REL_ACTIVITY_ID                                                     | Global Partitioned         | HASH(REL_ACTIVITY_ID)                            |                     |
| D1_ACTIVITY_REL_OBJ  | REFERENCE (D1_ACTIVITY_REL_OBJ_FK)                           |            |                                                                     |                            |                                                  |                     |
|                      |                                                              | D1T324P0   | D1_ACTIVITY_ID, MAINT_OBJ_CD, ACTIVITY_REL_OBJ_TYPE_FLG             | Global Partitioned         | RANGE(D1_ACTIVITY_ID)                            |                     |
|                      |                                                              | D1T324S0   | PK_VALUE1, PK_VALUE2, PK_VALUE3, PK_VALUE4, PK_VALUE5, MAINT_OBJ_CD | Global Partitioned         | HASH(PK_VALUE1, PK_VALUE2, PK_VALUE3, PK_VALUE4) |                     |



**Communication In**

This table describes the Communication In maintenance object.

| Table Name            | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name          | Index Columns                              | Index Type Global or Local | Index Partitioning Sub-Partitioning Key          | ILM_DT Initial Load      |
|-----------------------|--------------------------------------------------------------|---------------------|--------------------------------------------|----------------------------|--------------------------------------------------|--------------------------|
| D1_COMM_IN (Parent)   | RANGE(ILM_DT, D1_COMM_ID)                                    |                     |                                            |                            |                                                  | D1_COMM_IN.<br>CRE_DT_TM |
|                       |                                                              | D1T386P0            | D1_COMM_ID                                 | Global Partitioned         | RANGE (D1_COMM_ID)                               |                          |
|                       |                                                              | D1T386S1            | BUS_OBJ_CD,<br>BO_STATUS_CD,<br>D1_COMM_ID | Global Partitioned         | HASH(BUS_OBJ_CD,<br>BO_STATUS_CD,<br>D1_COMM_ID) |                          |
|                       |                                                              | CM_ILM_<br>D1T386S1 | ILM_DT,<br>ILM_ARCH_SW,<br>D1_COMM_ID      | Local                      |                                                  |                          |
| D1_COMM_IN_CHAR       | REFERENCE (D1_COMM_IN_CHAR_FK)                               |                     |                                            |                            |                                                  |                          |
|                       |                                                              | D1T387P0            | D1_COMM_ID,<br>CHAR_TYPE_CD,<br>SEQ_NUM    | Global Partitioned         | RANGE (D1_COMM_ID)                               |                          |
|                       |                                                              | D1T387S0            | SRCH_CHAR_VAL                              | Global Partitioned         | HASH(SRCH_CHAR_VAL)                              |                          |
| D1_COMM_IN_IDENTIFIER | REFERENCE (D1_COMM_IN_IDENTIFIER_FK)                         |                     |                                            |                            |                                                  |                          |
|                       |                                                              | D1T391P0            | D1_COMM_ID,<br>COMM_ID_TYPE_FLG            | Global Partitioned         | RANGE(D1_COMM_ID)                                |                          |
|                       |                                                              | D1T391S0            | COMM_ID_TYPE_FLG,<br>ID_VALUE              | Global Partitioned         | HASH(COMM_ID_TYPE_FLG,<br>ID_VALUE)              |                          |
|                       |                                                              | D1T391S1            | COMM_ID_TYPE_FLG,<br>UPPER(ID_VALUE)       |                            |                                                  |                          |
| D1_COMM_IN_LOG        | REFERENCE (D1_COMM_IN_LOG_FK)                                |                     |                                            |                            |                                                  |                          |
|                       |                                                              | D1T388P0            | D1_COMM_ID,<br>SEQNO                       | Global Partitioned         | RANGE(D1_COMM_ID)                                |                          |
|                       |                                                              | D1T388S1            | CHAR_TYPE_CD,<br>CHAR_VAL_FK1              | Global Partitioned         | HASH(CHAR_TYPE_CD,<br>CHAR_VAL_FK1)              |                          |
|                       |                                                              | D1T388S2            | CHAR_TYPE_CD,<br>CHAR_VAL                  | Global Partitioned         | HASH(CHAR_TYPE_CD,<br>CHAR_VAL)                  |                          |
| D1_COMM_IN_LOG_PARM   | REFERENCE (D1_COMM_IN_LOG_PARM_FK)                           |                     |                                            |                            |                                                  |                          |

| Table Name         | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name | Index Columns                                                                      | Index Type Global or Local | Index Partitioning Sub-Partitioning Key                   | ILM_DT Initial Load |
|--------------------|--------------------------------------------------------------|------------|------------------------------------------------------------------------------------|----------------------------|-----------------------------------------------------------|---------------------|
|                    |                                                              | D1T389P0   | D1_COMM_ID,<br>SEQNO<br>PARM_SEQ                                                   | Global<br>Partitioned      | RANGE(D1_COMM_ID)                                         |                     |
| D1_COMM_IN_REL_OBJ | REFERENCE (D1_COMM_IN_REL_OBJ_FK)                            |            |                                                                                    |                            |                                                           |                     |
|                    |                                                              | D1T390P0   | D1_COMM_ID,<br>MAINT_OBJ_CD,<br>COMM_REL_OBJ_TYPE_FLG                              | Global<br>Partitioned      | RANGE(D1_COMM_ID)                                         |                     |
|                    |                                                              | D1T390S0   | PK_VALUE1,<br>PK_VALUE2,<br>PK_VALUE3,<br>PK_VALUE4,<br>PK_VALUE5,<br>MAINT_OBJ_CD | Global<br>Partitioned      | HASH(PK_VALUE1,<br>PK_VALUE2,<br>PK_VALUE3,<br>PK_VALUE4) |                     |

### Communication Out

This table describes the Communication Out maintenance object.

| Table Name           | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name      | Index Columns                              | Index Type Global or Local | Index Partitioning Sub-Partitioning Key          | ILM_DT Initial Load      |
|----------------------|--------------------------------------------------------------|-----------------|--------------------------------------------|----------------------------|--------------------------------------------------|--------------------------|
| D1_COMM_OUT (Parent) | RANGE(ILM_DT, D1_COMM_ID)                                    |                 |                                            |                            |                                                  | D1_COMM_OUT.<br>CRE_DTTM |
|                      |                                                              | D1T380P0        | D1_COMM_ID                                 | Global<br>Partitioned      | RANGE (D1_COMM_ID)                               |                          |
|                      |                                                              | D1T380S1        | BUS_OBJ_CD,<br>BO_STATUS_CD,<br>D1_COMM_ID | Global<br>Partitioned      | HASH(BUS_OBJ_CD,<br>BO_STATUS_CD,<br>D1_COMM_ID) |                          |
|                      |                                                              | CM_ILM_D1T380S1 | ILM_DT,<br>ILM_ARCH_SW,<br>D1_COMM_ID      | Local                      |                                                  |                          |
| D1_COMM_OUT_CHAR     | REFERENCE (D1_COMM_OUT_CHAR_FK)                              |                 |                                            |                            |                                                  |                          |
|                      |                                                              | D1T381P0        | D1_COMM_ID,<br>CHAR_TYPE_CD,<br>SEQ_NUM    | Global<br>Partitioned      | RANGE (D1_COMM_ID)                               |                          |
|                      |                                                              | D1T381S0        | SRCH_CHAR_VAL                              | Global<br>Partitioned      | HASH(SRCH_CHAR_VAL)                              |                          |

| Table Name             | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name | Index Columns                                                       | Index Type Global or Local | Index Partitioning Sub-Partitioning Key          | ILM_DT Initial Load |
|------------------------|--------------------------------------------------------------|------------|---------------------------------------------------------------------|----------------------------|--------------------------------------------------|---------------------|
| D1_COMM_OUT_IDENTIFIER | REFERENCE (D1_COMM_OUT_IDENTIFIER_FK)                        |            |                                                                     |                            |                                                  |                     |
|                        |                                                              | D1T385P0   | D1_COMM_ID, COMM_ID_TYPE_FLG                                        | Global Partitioned         | RANGE(D1_COMM_ID)                                |                     |
|                        |                                                              | D1T385S0   | COMM_ID_TYPE_FLG, ID_VALUE                                          | Global Partitioned         | HASH(COMM_ID_TYPE_FLG, ID_VALUE)                 |                     |
|                        |                                                              | D1T385S1   | COMM_ID_TYPE_FLG, UPPER(ID_VALUE)                                   |                            |                                                  |                     |
| D1_COMM_OUT_LOG        | REFERENCE (D1_COMM_OUT_LOG_FK)                               |            |                                                                     |                            |                                                  |                     |
|                        |                                                              | D1T382P0   | D1_COMM_ID, SEQNO                                                   | Global Partitioned         | RANGE(D1_COMM_ID)                                |                     |
|                        |                                                              | D1T382S1   | CHAR_TYPE_CD, CHAR_VAL_FK1                                          | Global Partitioned         | HASH(CHAR_TYPE_CD, CHAR_VAL_FK1)                 |                     |
|                        |                                                              | D1T382S2   | CHAR_TYPE_CD, CHAR_VAL                                              | Global Partitioned         | HASH(CHAR_TYPE_CD, CHAR_VAL)                     |                     |
| D1_COMM_OUT_LOG_PARM   | REFERENCE (D1_COMM_OUT_LOG_PARM_FK)                          |            |                                                                     |                            |                                                  |                     |
|                        |                                                              | D1T383P0   | D1_COMM_ID, SEQNO, PARM_SEQ                                         | Global Partitioned         | RANGE(D1_COMM_ID)                                |                     |
| D1_COMM_OUT_REL_OBJ    | REFERENCE (D1_COMM_OUT_REL_OBJ_FK)                           |            |                                                                     |                            |                                                  |                     |
|                        |                                                              | D1T384P0   | D1_COMM_ID, MAINT_OBJ_CD, COMM_REL_OBJ_TYPE_FLG                     | Global Partitioned         | RANGE(D1_COMM_ID)                                |                     |
|                        |                                                              | D1T384S0   | PK_VALUE1, PK_VALUE2, PK_VALUE3, PK_VALUE4, PK_VALUE5, MAINT_OBJ_CD | Global Partitioned         | HASH(PK_VALUE1, PK_VALUE2, PK_VALUE3, PK_VALUE4) |                     |

## Device Event

If sub retention periods will be defined for this MO, then please follow the guidelines set forth in section [Module Specific ILM Implementation Details For Sub Retention](#).

This table describes the Device Event maintenance object.

| Table Name            | Table Partitioning Type (Partitioning, Sub-Partitioning Key)                                              | Index Name      | Index Columns                                      | Index Type Global or Local | Index Partitioning Sub-Partitioning Key                  | ILM_DT Initial Load  |
|-----------------------|-----------------------------------------------------------------------------------------------------------|-----------------|----------------------------------------------------|----------------------------|----------------------------------------------------------|----------------------|
| D1_DVC_EVT (Parent)   | RANGE(ILM_DT, DVC_EVT_ID)                                                                                 |                 |                                                    |                            |                                                          | D1_DVC_EVT, CRE_DTTM |
|                       | <b>Note:</b> Default is to use sub-retention or use RANGE (ILM_DT,DVC_EVT_ID) if not using sub-retention. |                 |                                                    |                            |                                                          |                      |
|                       |                                                                                                           | D1T400P0        | DVC_EVT_ID                                         | Global Partitioned         | RANGE (DVC_EVT_ID)                                       |                      |
|                       |                                                                                                           | D1T400S1        | BUS_OBJ_CD, BO_STATUS_CD, DVC_EVT_ID               | Global Partitioned         | HASH(BUS_OBJ_CD, BO_STATUS_CD, DVC_EVT_ID)               |                      |
|                       |                                                                                                           | D1T400S2        | D1_DEVICE_ID, DVC_EVT_DTTM                         | Global Partitioned         | HASH(D1_DEVICE_ID, DVC_EVT_DTTM)                         |                      |
|                       |                                                                                                           | D1T400S3        | BUS_OBJ_CD, BO_STATUS_CD, D1_DEVICE_ID, DVC_EVT_ID | Global Partitioned         | HASH(BUS_OBJ_CD, BO_STATUS_CD, D1_DEVICE_ID, DVC_EVT_ID) |                      |
|                       |                                                                                                           | CM_ILM_D1T400S4 | ILM_DT, ILM_ARCH_SW, DVC_EVT_ID                    | Local                      |                                                          |                      |
| D1_DVC_EVT_CHAR       | REFERENCE (D1_DVC_EVT_CHAR_FK)                                                                            |                 |                                                    |                            |                                                          |                      |
|                       |                                                                                                           | D1T401P0        | DVC_EVT_ID, CHAR_TYPE_CD, SEQ_NUM                  | Global Partitioned         | RANGE(DVC_EVT_ID)                                        |                      |
|                       |                                                                                                           | D1T401S0        | SRCH_CHAR_VAL                                      | Global Partitioned         | HASH(SRCH_CHAR_VAL)                                      |                      |
| D1_DVC_EVT_IDENTIFIER | REFERENCE (D1_DVC_EVT_IDENTIFIER_FK)                                                                      |                 |                                                    |                            |                                                          |                      |
|                       |                                                                                                           | D1T405P0        | DVC_EVT_ID, DVC_EVT_ID_TYPE_FLG                    | Global Partitioned         | RANGE(DVC_EVT_ID)                                        |                      |
|                       |                                                                                                           | D1T405S0        | DVC_EVT_ID_TYPE_FLG, ID_VALUE                      | Global Partitioned         | HASH(DVC_EVT_ID_TYPE_FLG, ID_VALUE)                      |                      |
|                       |                                                                                                           | D1T405S1        | DVC_EVT_ID_TYPE_FLG, UPPER(ID_VALUE)               |                            |                                                          |                      |

| Table Name          | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name | Index Columns                                                       | Index Type Global or Local | Index Partitioning Sub-Partitioning Key          | ILM_DT Initial Load |
|---------------------|--------------------------------------------------------------|------------|---------------------------------------------------------------------|----------------------------|--------------------------------------------------|---------------------|
| D1_DVC_EVT_LOG      | REFERENCE (D1_DVC_EVT_LOG_FK)                                |            |                                                                     |                            |                                                  |                     |
|                     |                                                              | D1T402P0   | DVC_EVT_ID, SEQNO                                                   | Global Partitioned         | RANGE(DVC_EVT_ID)                                |                     |
|                     |                                                              | D1T402S1   | CHAR_TYPE_CD, CHAR_VAL_FK1                                          | Global Partitioned         | HASH(CHAR_TYPE_CD, CHAR_VAL_FK1)                 |                     |
|                     |                                                              | D1T402S2   | CHAR_TYPE_CD, CHAR_VAL                                              | Global Partitioned         | HASH(CHAR_TYPE_CD, CHAR_VAL)                     |                     |
| D1_DVC_EVT_LOG_PARM | REFERENCE (D1_DVC_EVT_LOG_PARM_FK)                           |            |                                                                     |                            |                                                  |                     |
|                     |                                                              | D1T403P0   | DVC_EVT_ID, SEQNO, PARM_SEQ                                         | Global Partitioned         | RANGE(DVC_EVT_ID)                                |                     |
| D1_DVC_EVT_REL_OBJ  | REFERENCE (D1_DVC_EVT_REL_OBJ_FK)                            |            |                                                                     |                            |                                                  |                     |
|                     |                                                              | D1T404P0   | DVC_EVT_ID, MAINT_OBJ_CD, DVC_EVT_REL_OBJ_TYP, E_FLG                | Global Partitioned         | RANGE(DVC_EVT_ID)                                |                     |
|                     |                                                              | D1T404S0   | PK_VALUE1, PK_VALUE2, PK_VALUE3, PK_VALUE4, PK_VALUE5, MAINT_OBJ_CD | Global Partitioned         | HASH(PK_VALUE1, PK_VALUE2, PK_VALUE3, PK_VALUE4) |                     |

### Completion Event

This table describes the Completion Event maintenance object.

| Table Name            | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name | Index Columns | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load    |
|-----------------------|--------------------------------------------------------------|------------|---------------|----------------------------|-----------------------------------------|------------------------|
| D1_COMPL_EVT (Parent) | RANGE(ILM_DT, COMPL_EVT_ID)                                  |            |               |                            |                                         | D1_COMPL_EVT, CRE_DTTM |
|                       |                                                              | D1T340P0   | COMPL_EVT_ID  | Global Partitioned         | RANGE (COMPL_EVT_ID)                    |                        |

| Table Name            | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name          | Index Columns                                                                      | Index Type Global or Local | Index Partitioning Sub-Partitioning Key                   | ILM_DT Initial Load |
|-----------------------|--------------------------------------------------------------|---------------------|------------------------------------------------------------------------------------|----------------------------|-----------------------------------------------------------|---------------------|
|                       |                                                              | D1T340S0            | D1_ACTIVITY_ID                                                                     | Global Partitioned         | HASH(D1_ACTIVITY_ID)                                      |                     |
|                       |                                                              | CM_ILM_<br>D1T340S1 | ILM_DT,<br>ILM_ARCH_SW,<br>DVC_EVT_ID                                              | Local                      |                                                           |                     |
| D1_COMPL_EVT_CHAR     | REFERENCE (D1_COMPL_EVT_CHAR_FK)                             |                     |                                                                                    |                            |                                                           |                     |
|                       |                                                              | D1T341P0            | COMPL_EVT_ID,<br>CHAR_TYPE_CD,<br>SEQ_NUM                                          | Global Partitioned         | RANGE(COMPL_EVT_ID)                                       |                     |
|                       |                                                              | D1T341S1            | SRCH_CHAR_VAL                                                                      | Global Partitioned         | HASH(SRCH_CHAR_VAL)                                       |                     |
| D1_COMPL_EVT_LOG      | REFERENCE (D1_COMPL_EVT_LOG_FK)                              |                     |                                                                                    |                            |                                                           |                     |
|                       |                                                              | D1T342P0            | COMPL_EVT_ID,<br>SEQNO                                                             | Global Partitioned         | RANGE(COMPL_EVT_ID)                                       |                     |
|                       |                                                              | D1T342S1            | CHAR_TYPE_CD,<br>CHAR_VAL_FK1                                                      | Global Partitioned         | HASH(CHAR_TYPE_CD,<br>CHAR_VAL_FK1)                       |                     |
|                       |                                                              | D1T342S2            | CHAR_TYPE_CD,<br>CHAR_VAL                                                          | Global Partitioned         | HASH(CHAR_TYPE_CD,<br>CHAR_VAL)                           |                     |
| D1_COMPL_EVT_LOG_PARM | REFERENCE (D1_COMPL_EVT_LOG_PARM_FK)                         |                     |                                                                                    |                            |                                                           |                     |
|                       |                                                              | D1T343P0            | COMPL_EVT_ID,<br>SEQNO<br>PARM_SEQ                                                 | Global Partitioned         | RANGE(COMPL_EVT_ID)                                       |                     |
| D1_COMPL_EVT_REL_OBJ  | REFERENCE (D1_COMPL_EVT_REL_OBJ_FK)                          |                     |                                                                                    |                            |                                                           |                     |
|                       |                                                              | D1T344P0            | COMPL_EVT_ID,<br>MAINT_OBJ_CD,<br>COMPL_EVT_REL_OBJ_T<br>YP_FLG                    | Global Partitioned         | RANGE(COMPL_EVT_ID)                                       |                     |
|                       |                                                              | D1T344S0            | PK_VALUE1,<br>PK_VALUE2,<br>PK_VALUE3,<br>PK_VALUE4,<br>PK_VALUE5,<br>MAINT_OBJ_CD | Global Partitioned         | HASH(PK_VALUE1,<br>PK_VALUE2,<br>PK_VALUE3,<br>PK_VALUE4) |                     |

## Initial Measurement Data

If sub retention periods will be defined for this MO, then please follow the guidelines set forth in section [Module Specific ILM Implementation Details For Sub Retention](#).

This table describes the Initial Measurement Data maintenance object.

| Table Name                  | Table Partitioning Type (Partitioning, Sub-Partitioning Key)                                                                                 | Index Name      | Index Columns                                                                 | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load             |
|-----------------------------|----------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------------------------------------------------------------------------|----------------------------|-----------------------------------------|---------------------------------|
| D1_INIT_MSRMT_DATA (Parent) | RANGE<br>(ILM_DT,MEASR_COMP_ID)<br><br>Note: Default is to use sub-retention or use RANGE (ILM_DT,MEASR_COMP_ID) if not using sub-retention. |                 |                                                                               |                            |                                         | D1_INIT_MSRMT_DATA.<br>CRE_DTTM |
|                             |                                                                                                                                              | D1T304P0        | INIT_MSRMT_DATA_ID                                                            | Global Partitioned         | RANGE (INIT_MSRMT_DATA_ID)              |                                 |
|                             |                                                                                                                                              | D1T304S1        | MEASR_COMP_ID,<br>D1_TO_DTTM                                                  |                            |                                         |                                 |
|                             |                                                                                                                                              | D1T304S1        | MEASR_COMP_ID,<br>BO_STATUS_CD,<br>BUS_OBJ_CD,<br>D1_TO_DTTM,<br>D1_FROM_DTTM | Global Partitioned         | RANGE(MEASR_COMP_ID)                    |                                 |
|                             |                                                                                                                                              | CM_ILM_D1T304S4 | ILM_DT,<br>ILM_ARCH_SW,<br>INIT_MSRMT_DATA_ID                                 | Local                      |                                         |                                 |
| D1_INIT_MSRMT_DATA_CHAR     | REFERENCE<br>(D1_INIT_MSRMT_DATA_CHAR_FK)                                                                                                    |                 |                                                                               |                            |                                         |                                 |
|                             |                                                                                                                                              | D1T305P0        | INIT_MSRMT_DATA_ID,<br>CHAR_TYPE_CD,<br>SEQ_NUM                               | Global Partitioned         | RANGE(INIT_MSRMT_DATA_ID)               |                                 |
|                             |                                                                                                                                              | D1T305S1        | SRCH_CHAR_VAL                                                                 | Global Partitioned         | HASH(SRCH_CHAR_VAL)                     |                                 |
| D1_INIT_MSRMT_DATA_LOG      | REFERENCE<br>(D1_INIT_MSRMT_DATA_LOG_FK)                                                                                                     |                 |                                                                               |                            |                                         |                                 |
|                             |                                                                                                                                              | D1T306P0        | INIT_MSRMT_DATA_ID,<br>SEQNO                                                  | Global Partitioned         | RANGE (INIT_MSRMT_DATA_ID)              |                                 |
| D1_INIT_MSRMT_DATA_LOG_PARM | REFERENCE<br>(D1_INIT_MSRMT_DATA_LOG_PARM_FK)                                                                                                |                 |                                                                               |                            |                                         |                                 |

| Table Name | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name | Index Columns                       | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load |
|------------|--------------------------------------------------------------|------------|-------------------------------------|----------------------------|-----------------------------------------|---------------------|
|            |                                                              | D1T307P0   | INIT_MSRMT_DATA_ID, SEQNO, PARM_SEQ | Global Partitioned         | RANGE (INIT_MSRMT_DATA_ID)              |                     |

### Usage Transaction

This table describes the Usage Transaction maintenance object.

| Table Name        | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name      | Index Columns                         | Index Type Global or Local | Index Partitioning Sub-Partitioning Key     | ILM_DT Initial Load |
|-------------------|--------------------------------------------------------------|-----------------|---------------------------------------|----------------------------|---------------------------------------------|---------------------|
| D1_USAGE (Parent) | RANGE(ILM_DT, D1_USAGE_ID)                                   |                 |                                       |                            |                                             | D1_USAGE.CRE_DTTM   |
|                   |                                                              | D1T281P0        | D1_USAGE_ID                           | Global Partitioned         | RANGE (D1_USAGE_ID)                         |                     |
|                   |                                                              | D1T281S0        | US_ID, START_DTTM                     | Global Partitioned         | RANGE (US_ID)                               |                     |
|                   |                                                              | D1T281S1        | BUS_OBJ_CD, BO_STATUS_CD, D1_USAGE_ID | Global Partitioned         | HASH(BUS_OBJ_CD, BO_STATUS_CD, D1_USAGE_ID) |                     |
|                   |                                                              | CM_ILM_D1T281S2 | ILM_DT, ILM_ARCH_SW, D1_USAGE_ID      | Local                      |                                             |                     |
|                   |                                                              | D1T419S1        | USG_EXT_ID, D1_USAGE_ID               | Global Partitioned         | RANGE (USG_EXT_ID)                          |                     |
| D1_USAGE_CHAR     | REFERENCE (D1_USAGE_CHAR_FK)                                 |                 |                                       |                            |                                             |                     |
|                   |                                                              | D1T285P0        | D1_USAGE_ID, CHAR_TYPE_CD, SEQ_NUM    | Global Partitioned         | RANGE(D1_USAGE_ID)                          |                     |
|                   |                                                              | D1T285S1        | SRCH_CHAR_VAL                         | Global Partitioned         | HASH(SRCH_CHAR_VAL)                         |                     |
| D1_USAGE_LOG      | REFERENCE (D1_USAGE_LOG_FK)                                  |                 |                                       |                            |                                             |                     |
|                   |                                                              | D1T286P0        | D1_USAGE_ID, SEQNO                    | Global Partitioned         | RANGE(D1_USAGE_ID)                          |                     |



| Table Name               | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name | Index Columns                                                  | Index Type Global or Local | Index Partitioning Sub-Partitioning Key                   | ILM_DT Initial Load |
|--------------------------|--------------------------------------------------------------|------------|----------------------------------------------------------------|----------------------------|-----------------------------------------------------------|---------------------|
|                          |                                                              | D1T286S1   | CHAR_TYPE_CD,<br>CHAR_VAL_FK1                                  | Global Partitioned         | HASH(CHAR_TYPE_CD,<br>CHAR_VAL_FK1)                       |                     |
|                          |                                                              | D1T286S2   | CHAR_TYPE_CD,<br>CHAR_VAL                                      | Global Partitioned         | HASH(CHAR_TYPE_CD,<br>CHAR_VAL)                           |                     |
| D1_USAGE_LOG_PARM        | REFERENCE(D1_USAGE_LOG_PARM_FK)                              |            |                                                                |                            |                                                           |                     |
|                          |                                                              | D1T287P0   | D1_USAGE_ID,<br>SEQNO<br>PARM_SEQ                              | Global Partitioned         | RANGE<br>(D1_USAGE_ID)                                    |                     |
| D1_USAGE_PERIOD          | REFERENCE(D1_USAGE_PERIOD_FK)                                |            |                                                                |                            |                                                           |                     |
|                          |                                                              | D1T283P0   | D1_USAGE_ID,<br>PERIOD_SEQ_NUM                                 | Global Partitioned         | RANGE(D1_USAGE_ID)                                        |                     |
| D1_USAGE_PERIOD_ITEM_DET | REFERENCE(D1_USAGE_PERIOD_ITEM_DET_FK)                       |            |                                                                |                            |                                                           |                     |
|                          |                                                              | D1T431P0   | D1_USAGE_ID,<br>PERIOD_SEQ_NUM,<br>ITEM_SEQ_NUM                | Global Partitioned         | RANGE(D1_USAGE_ID)                                        |                     |
| D1_USAGE_PERIOD_SQ       | REFERENCE(D1_USAGE_PERIOD_SQ_FK)                             |            |                                                                |                            |                                                           |                     |
|                          |                                                              | D1T284P0   | D1_USAGE_ID,<br>PERIOD_SEQ_NUM,<br>SQ_SEQ_NUM                  | Global Partitioned         | RANGE(D1_USAGE_ID)                                        |                     |
| D1_USAGE_PERIOD_SQ_DATA  | REFERENCE(D1_USAGE_PERIOD_SQ_DATA_FK)                        |            |                                                                |                            |                                                           |                     |
|                          |                                                              | D1T497P0   | D1_USAGE_ID,<br>PERIOD_SEQ_NUM,<br>SQ_SEQ_NUM,<br>SQ_DATA_DTTM | Global Partitioned         | RANGE(D1_USAGE_ID)                                        |                     |
| D1_USAGE_REL             | REFERENCE(D1_USAGE_REL_FK)                                   |            |                                                                |                            |                                                           |                     |
|                          |                                                              | D1T316P0   | D1_USAGE_ID,<br>USAGE_REL_TYPE_FLG                             | Global Partitioned         | RANGE(D1_USAGE_ID)                                        |                     |
|                          |                                                              | D1T316S0   | REL_USAGE_ID,<br>USAGE_REL_TYPE_FLG,<br>D1_USAGE_ID            | Global Partitioned         | HASH(REL_USAGE_ID,<br>USAGE_REL_TYPE_FLG,<br>D1_USAGE_ID) |                     |

| Table Name          | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name | Index Columns                        | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load |
|---------------------|--------------------------------------------------------------|------------|--------------------------------------|----------------------------|-----------------------------------------|---------------------|
| D1_USAGE_SCALAR_DTL | REFERENCE(D1_USAGE_SCALAR_DTL_FK)                            | D1T282P0   | D1_USAGE_ID,<br>D1_SP_ID,<br>SEQ_NUM | Global<br>Partitioned      | RANGE(D1_USAGE_ID)                      |                     |

### Usage Transaction Exception

This table describes the Usage Transaction Exception maintenance object.

| Table Name             | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name      | Index Columns                              | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load        |
|------------------------|--------------------------------------------------------------|-----------------|--------------------------------------------|----------------------------|-----------------------------------------|----------------------------|
| D1_USAGE_EXCP (Parent) | RANGE (ILM_DT, USAGE_EXCP_ID)                                |                 |                                            |                            |                                         | D1_USAGE_EXCP.<br>CRE_DTTM |
|                        |                                                              | D1T443P0        | USAGE_EXCP_ID                              | Global<br>Partitioned      | RANGE (USAGE_EXCP_ID)                   |                            |
|                        |                                                              | CM_ILM_D1T443S1 | ILM_DT,<br>ILM_ARCH_SW,<br>USAGE_EXCP_ID   | Local<br>Partitioned       |                                         |                            |
| D1_USAGE_EXCP_CHAR     | REFERENCE (D1_USAGE_EXCP_CHAR_FK)                            | D1T446P0        | USAGE_EXCP_ID,<br>CHAR_TYPE_CD,<br>SEQ_NUM | Global<br>Partitioned      | RANGE (USAGE_EXCP_ID)                   |                            |
|                        |                                                              | D1T446S1        | SRCH_CHAR_VAL                              | Global<br>Partitioned      | HASH (SRCH_CHAR_VAL)                    |                            |
| D1_USAGE_EXCP_PARM     | REFERENCE (D1_USAGE_EXCP_PARM_FK)                            | D1T445P0        | USAGE_EXCP_ID,<br>PARM_SEQ                 | Global<br>Partitioned      | RANGE (USAGE_EXCP_ID)                   |                            |

**VEE Exception**

This table describes the VEE Exception maintenance object.

| Table Name           | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name          | Index Columns                            | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load      |
|----------------------|--------------------------------------------------------------|---------------------|------------------------------------------|----------------------------|-----------------------------------------|--------------------------|
| D1_VEE_EXCP (Parent) | RANGE(ILM_DT, VEE_EXCP_ID)                                   |                     |                                          |                            |                                         | D1_VEE_EXCP.<br>CRE_DTTM |
|                      |                                                              | D1T308P0            | VEE_EXCP_ID                              | Global Partitioned         | RANGE (VEE_EXCP_ID)                     |                          |
|                      |                                                              | D1T308S1            | INIT_MSRMT_DATA_ID                       | Global Partitioned         | HASH(INIT_MSRMT_DATA_ID)                |                          |
|                      |                                                              | CM_ILM_<br>D1T308S2 | ILM_DT,<br>ILM_ARCH_SW,<br>VEE_EXCP_ID   | Local                      |                                         |                          |
| D1_VEE_EXCP_CHAR     | REFERENCE (D1_VEE_EXCP_CHAR_FK)                              |                     |                                          |                            |                                         |                          |
|                      |                                                              | D1T310P0            | VEE_EXCP_ID,<br>CHAR_TYPE_CD,<br>SEQ_NUM | Global Partitioned         | RANGE(VEE_EXCP_ID)                      |                          |
|                      |                                                              | D1T310S1            | SRCH_CHAR_VAL                            | Global Partitioned         | HASH(SRCH_CHAR_VAL)                     |                          |
| D1_VEE_EXCP_PARM     | REFERENCE (D1_VEE_EXCP_PARM_FK)                              |                     |                                          |                            |                                         |                          |
|                      |                                                              | D1T309P0            | VEE_EXCP_ID,<br>PARM_SEQ                 | Global Partitioned         | RANGE(VEE_EXCP_ID))                     |                          |

**Snapshot Tables**

This table below describes the snapshot tables.

| Table Name          | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name | Index Columns         | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load               |
|---------------------|--------------------------------------------------------------|------------|-----------------------|----------------------------|-----------------------------------------|-----------------------------------|
| D1_SNAPSHOT_DL_CTRL | RANGE(ILM_DT)                                                |            |                       |                            |                                         | D1_SNAPSHOT_DL_CTRL.SNAPSHOT_DTTM |
|                     |                                                              | D1T433P0   | SNAPSHOT_FACT_NAME_CD | Global Partitioned         |                                         |                                   |

| Table Name             | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name      | Index Columns                                                                                                                        | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load                  |
|------------------------|--------------------------------------------------------------|-----------------|--------------------------------------------------------------------------------------------------------------------------------------|----------------------------|-----------------------------------------|--------------------------------------|
|                        |                                                              | CM_ILM_D1T433S1 | ILM_DT, ILM_ARCH_SW, SNAPSHOT_FACT_NAME_CD, SNAPSHOT_DTTM                                                                            | Local                      |                                         |                                      |
| D1_SP_SNAP_DL          | RANGE(ILM_DT, SP_SNAP_ID)                                    |                 |                                                                                                                                      |                            |                                         | D1_SP_SNAP_DL.SNAPSHOT_DTTM          |
|                        |                                                              | D1T434P0        | SP_SNAP_ID                                                                                                                           | Global Partitioned         | RANGE(SP_SNAP_ID)                       |                                      |
|                        |                                                              | D1T434S0        | D1_SP_ID, SNAPSHOT_DTTM, SNAPSHOT_TYPE_FLG                                                                                           | Global Partitioned         |                                         |                                      |
|                        |                                                              | CM_ILM_D1T434S1 | ILM_DT, ILM_ARCH_SW, SP_SNAP_ID                                                                                                      | Local                      |                                         |                                      |
| D1_SP_UNR_USG_SNAP_DL  | RANGE(ILM_DT, SP_UNR_USG_SNAP_ID)                            |                 |                                                                                                                                      |                            |                                         | D1_SP_UNR_USG_SNAP_DL.SNAPSHOT_DTTM  |
|                        |                                                              | D1T438P0        | SP_UNR_USG_SNAP_ID                                                                                                                   | Global Partitioned         | RANGE(SP_USG_SNAP_ID)                   |                                      |
|                        |                                                              | D1T436S0        | D1_SP_ID, SNAPSHOT_DTTM, MEASR_COMP_ID, USG_SNAPSHOT_TYPE_FLG, D1_TOU_CD, MSRMT_COND_FLG, SNAPSHOT_TYPE_FLG                          | Global Partitioned         |                                         |                                      |
|                        |                                                              | CM_ILM_D1T436S1 | ILM_DT, ILM_ARCH_SW, SP_USG_SNAP_ID                                                                                                  | Local                      |                                         |                                      |
| D1_SP_VEE_EXCP_SNAP_DL | RANGE(ILM_DT, SP_VEE_EXCP_SNAP_ID)                           |                 |                                                                                                                                      |                            |                                         | D1_SP_VEE_EXCP_SNAP_DL.SNAPSHOT_DTTM |
|                        |                                                              | D1T440P0        | SP_VEE_EXCP_SNAP_ID                                                                                                                  | Global Partitioned         | RANGE(SP_VEE_EXCP_SNAP_ID)              |                                      |
|                        |                                                              | D1T440S0        | D1_SP_ID, SNAPSHOT_DTTM, MEASR_COMP_ID, EXCP_TYPE_CD, D1_IMD_TYPE_FLG, EXCP_SEVERITY_FLG, VEE_GRP_CD, VEE_RULE_CD, SNAPSHOT_TYPE_FLG |                            |                                         |                                      |

| Table Name | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name      | Index Columns                                  | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load |
|------------|--------------------------------------------------------------|-----------------|------------------------------------------------|----------------------------|-----------------------------------------|---------------------|
|            |                                                              | CM_ILM_D1T440S1 | ILM_DT,<br>ILM_ARCH_SW,<br>SP_VEE_EXCP_SNAP_ID | Local                      |                                         |                     |

## Module Specific ILM Implementation Details For Sub Retention

This section outlines each maintenance object that has been configured to support ILM as well as sub retention periods. This differs from the standard ILM enabled tables in that the partitioning strategy is inclusive of an additional column that defines the retention period for each record. In each case, the recommendation of the initial load of the ILM\_DT and the <field name for retention period> for existing records is noted. The CTAS operation for these tables includes an extra step of generating a temporary mapping table that will allow the select for the ILM\_DT to also identify the appropriate <retention period field name> for each record.

This section details the following maintenance objects that support ILM as well as sub retention periods:

- [Activity](#)
- [Device Event](#)
- [Initial Measurement Data](#)

### Activity

If sub retention periods will not be defined for this MO, then please follow the guidelines set forth in section [Module Specific ILM Implementation Details](#).

| Table Name           | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name      | Index Columns                                                  | Index Type Global or Local | Index Partitioning Sub-Partitioning Key              | ILM_DT Initial Load   |
|----------------------|--------------------------------------------------------------|-----------------|----------------------------------------------------------------|----------------------------|------------------------------------------------------|-----------------------|
| D1_ACTIVITY (Parent) | RANGE (ILM_DT, RETENTION_PERIOD)                             |                 |                                                                |                            |                                                      | D1_ACTIVITY.CRE_DT_TM |
|                      |                                                              | D1T319P0        | D1_ACTIVITY_ID                                                 | Global Partitioned         | RANGE (D1_ACTIVITY_ID)                               |                       |
|                      |                                                              | D1T319S0        | BUS_OBJ_CD,<br>BO_STATUS_CD,<br>D1_ACTIVITY_ID                 | Global Partitioned         | HASH(BUS_OBJ_CD,<br>BO_STATUS_CD,<br>D1_ACTIVITY_ID) |                       |
|                      |                                                              | CM_ILM_D1T319S1 | ILM_DT,<br>RETENTION_PERIOD,<br>ILM_ARCH_SW,<br>D1_ACTIVITY_ID | Local                      |                                                      |                       |

| Table Name             | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name | Index Columns                         | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load |
|------------------------|--------------------------------------------------------------|------------|---------------------------------------|----------------------------|-----------------------------------------|---------------------|
| D1_ACTIVITY_CHAR       | REFERENCE (D1_ACTIVITY_CHAR_FK)                              |            |                                       |                            |                                         |                     |
|                        |                                                              | D1T320P0   | D1_ACTIVITY_ID, CHAR_TYPE_CD, SEQ_NUM | Global Partitioned         | RANGE(D1_ACTIVITY_ID)                   |                     |
|                        |                                                              | D1T320S0   | SRCH_CHAR_VAL                         | Global Partitioned         | HASH(SRCH_CHAR_VAL)                     |                     |
| D1_ACTIVITY_IDENTIFIER | REFERENCE (D1_ACTIVITY_IDENTIFIER_FK)                        |            |                                       |                            |                                         |                     |
|                        |                                                              | D1T330P0   | D1_ACTIVITY_ID, ACTIVITY_ID_TYPE_FLG  | Global Partitioned         | RANGE(D1_ACTIVITY_ID)                   |                     |
|                        |                                                              | D1T330S0   | ACTIVITY_ID_TYPE_FLG, ID_VALUE        | Global Partitioned         | HASH(ACTIVITY_ID_TYPE_FLG, ID_VALUE)    |                     |
|                        |                                                              | D1T330S1   | ACTIVITY_ID_TYPE_FLG, UPPER(ID_VALUE) |                            |                                         |                     |
| D1_ACTIVITY_LOG        | REFERENCE (D1_ACTIVITY_LOG_FK)                               |            |                                       |                            |                                         |                     |
|                        |                                                              | D1T321P0   | D1_ACTIVITY_ID, SEQNO                 | Global Partitioned         | RANGE(D1_ACTIVITY_ID)                   |                     |
|                        |                                                              | D1T321S1   | CHAR_TYPE_CD, CHAR_VAL_FK1            | Global Partitioned         | HASH(CHAR_TYPE_CD, CHAR_VAL_FK1)        |                     |
|                        |                                                              | D1T321S2   | CHAR_TYPE_CD, CHAR_VAL                | Global Partitioned         | HASH(CHAR_TYPE_CD, CHAR_VAL)            |                     |
| D1_ACTIVITY_LOG_PARM   | REFERENCE (D1_ACTIVITY_LOG_PARM_FK)                          |            |                                       |                            |                                         |                     |
|                        |                                                              | D1T322P0   | D1_ACTIVITY_ID, SEQNO, PARM_SEQ       | Global Partitioned         | RANGE(D1_ACTIVITY_ID)                   |                     |
| D1_ACTIVITY_REL        | REFERENCE (D1_ACTIVITY_REL_FK)                               |            |                                       |                            |                                         |                     |
|                        |                                                              | D1T323P0   | D1_ACTIVITY_ID, ACTIVITY_REL_TYPE_FLG | Global Partitioned         | RANGE(D1_ACTIVITY_ID)                   |                     |
|                        |                                                              | D1T323S0   | REL_ACTIVITY_ID                       | Global Partitioned         | HASH(REL_ACTIVITY_ID)                   |                     |

| Table Name          | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name | Index Columns                                                       | Index Type Global or Local | Index Partitioning Sub-Partitioning Key          | ILM_DT Initial Load |
|---------------------|--------------------------------------------------------------|------------|---------------------------------------------------------------------|----------------------------|--------------------------------------------------|---------------------|
| D1_ACTIVITY_REL_OBJ | REFERENCE (D1_ACTIVITY_REL_OBJ_FK)                           |            |                                                                     |                            |                                                  |                     |
|                     |                                                              | D1T324P0   | D1_ACTIVITY_ID, MAINT_OBJ_CD, ACTIVITY_REL_OBJ_TYPE_FLG             | Global Partitioned         | RANGE(D1_ACTIVITY_ID)                            |                     |
|                     |                                                              | D1T324S0   | PK_VALUE1, PK_VALUE2, PK_VALUE3, PK_VALUE4, PK_VALUE5, MAINT_OBJ_CD | Global Partitioned         | HASH(PK_VALUE1, PK_VALUE2, PK_VALUE3, PK_VALUE4) |                     |

#### Query for Setting the Retention Period

The following query should be used to create a temporary table to create a mapping table that will identify the retention period for each measuring component type. This table will then be used during in the CTAS operation for Activity to identify the retention period for each record.

Please refer to [Appendix C: Sample SQL for Enabling ILM with Sub Retention in SGG \(Existing Installation\)](#) for detailed information using Initial Measurement Data as an example.

**Note:** A pre-requisite to executing this query is configuring the appropriate retention periods in the ILM master configuration in the Oracle Utilities Meter Data Management application.

```

/*****ACTIVITY*****/
CREATE TABLE ILM_ACTIVITY_RETENTION_TMP
AS
select acty.activity_type_cd
/*retrieve the retention period for Activity Types in this order of
precedence:
1. The category based retention period from the MDM master
configuration
2. The MO level retention period from the MO options
3. The installation level retention period from the FW master
configuration
*/
, CAST(coalesce(catMap.retPeriod --Category level
, (select maint_obj_opt_val
from ci_md_mo_opt mmo
where maint_obj_cd = 'D1-ACTIVITY'
and maint_obj_opt_flg = 'FLRP'
and seq_num =
(select max(seq_num)
from ci_md_mo_opt mmo
where maint_obj_cd = 'D1-ACTIVITY'
and maint_obj_opt_flg = 'FLRP')) --MO level

```

```

, extractvalue(xmlparse(content fw_mcfg.mst_config_data)
,'generalMasterConfiguration/defaultRetentionPeriod') --Install
level
) as NUMBER(5)) retPeriod
from dl_activity_type acty
, (select extractvalue(value(p),
'activityTypeCategoryRetentionPeriodList/activityTypeCategory'
)ACTIVITY_TYPE_CAT_FLG
, extractvalue(value(p),
'activityTypeCategoryRetentionPeriodList/retentionPeriod'
)retPeriod
from fl_mst_config mdm_mcfg ,
table(xmlsequence(extract(xmlparse(content
mdm_mcfg.mst_config_data),
'activityRetentionPeriod/activityTypeCategoryRetentionPeriods/
activityTypeCategoryRetentionPeriodList'
))) p
where mdm_mcfg.bus_obj_cd = 'D1-ILMMSConfig') catMap
, fl_mst_config fw_mcfg
where fw_mcfg.bus_obj_cd = 'F1-ILMMSConfig'
and acty.ACTIVITY_TYPE_CAT_FLG = catMap.ACTIVITY_TYPE_CAT_FLG (+)
order by 1;

```

## Device Event

**Note:** If sub retention periods will not be defined for this MO, then please follow the guidelines set forth in section [Module Specific ILM Implementation Details](#).

| Table Name             | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name          | Index Columns                                               | Index Type Global or Local | Index Partitioning Sub-Partitioning Key                           | ILM_DT Initial Load     |
|------------------------|--------------------------------------------------------------|---------------------|-------------------------------------------------------------|----------------------------|-------------------------------------------------------------------|-------------------------|
| D1_DVC_EVT<br>(Parent) | RANGE(ILM_DT, RETENTION_PERIOD)                              |                     |                                                             |                            |                                                                   | D1_DVC_EVT.<br>CRE_DTTM |
|                        |                                                              | D1T400P0            | DVC_EVT_ID                                                  | Global<br>Partitioned      | RANGE<br>(DVC_EVT_ID)                                             |                         |
|                        |                                                              | D1T400S1            | BUS_OBJ_CD,<br>BO_STATUS_CD,<br>DVC_EVT_ID                  | Global<br>Partitioned      | HASH(BUS_OBJ_C<br>D,BO_STATUS_CD,<br>DVC_EVT_ID)                  |                         |
|                        |                                                              | D1T400S2            | D1_DEVICE_ID,<br>DVC_EVT_DTTM                               | Global<br>Partitioned      | HASH(D1_DEVICE<br>_ID,<br>DVC_EVT_DTTM)                           |                         |
|                        |                                                              | D1T400S3            | BUS_OBJ_CD,<br>BO_STATUS_CD,<br>D1_DEVICE_ID,<br>DVC_EVT_ID | Global<br>Partitioned      | HASH(BUS_OBJ_C<br>D,BO_STATUS_CD,<br>D1_DEVICE_ID,<br>DVC_EVT_ID) |                         |
|                        |                                                              | CM_ILM_<br>D1T400S4 | ILM_DT,<br>RETENTION_PERIOD,<br>ILM_ARCH_SW,<br>DVC_EVT_ID  | Local                      |                                                                   |                         |



| Table Name            | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name | Index Columns                                        | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load |
|-----------------------|--------------------------------------------------------------|------------|------------------------------------------------------|----------------------------|-----------------------------------------|---------------------|
| D1_DVC_EVT_CHAR       | REFERENCE (D1_DVC_EVT_CHAR_FK)                               |            |                                                      |                            |                                         |                     |
|                       |                                                              | D1T401P0   | DVC_EVT_ID, CHAR_TYPE_CD, SEQ_NUM                    | Global Partitioned         | RANGE(DVC_EVT_ID)                       |                     |
|                       |                                                              | D1T401S0   | SRCH_CHAR_VAL                                        | Global Partitioned         | HASH(SRCH_CHAR_VAL)                     |                     |
| D1_DVC_EVT_IDENTIFIER | REFERENCE (D1_DVC_EVT_IDENTIFIER_FK)                         |            |                                                      |                            |                                         |                     |
|                       |                                                              | D1T405P0   | DVC_EVT_ID, DVC_EVT_ID_TYPE_FLG                      | Global Partitioned         | RANGE(DVC_EVT_ID)                       |                     |
|                       |                                                              | D1T405S0   | DVC_EVT_ID_TYPE_FLG, ID_VALUE                        | Global Partitioned         | HASH(DVC_EVT_ID, ID_VALUE)              |                     |
|                       |                                                              | D1T405S1   | DVC_EVT_ID_TYPE_FLG, UPPER(ID_VALUE)                 |                            |                                         |                     |
| D1_DVC_EVT_LOG        | REFERENCE (D1_DVC_EVT_LOG_FK)                                |            |                                                      |                            |                                         |                     |
|                       |                                                              | D1T402P0   | DVC_EVT_ID, SEQNO                                    | Global Partitioned         | RANGE(DVC_EVT_ID)                       |                     |
|                       |                                                              | D1T402S1   | CHAR_TYPE_CD, CHAR_VAL_FK1                           | Global Partitioned         | HASH(CHAR_TYPE_CD, CHAR_VAL_FK1)        |                     |
|                       |                                                              | D1T402S2   | CHAR_TYPE_CD, CHAR_VAL                               | Global Partitioned         | HASH(CHAR_TYPE_CD, CHAR_VAL)            |                     |
| D1_DVC_EVT_LOG_PARM   | REFERENCE (D1_DVC_EVT_LOG_PARM_FK)                           |            |                                                      |                            |                                         |                     |
|                       |                                                              | D1T403P0   | DVC_EVT_ID, SEQNO, PARM_SEQ                          | Global Partitioned         | RANGE(DVC_EVT_ID)                       |                     |
| D1_DVC_EVT_REL_OBJ    | REFERENCE (D1_DVC_EVT_REL_OBJ_FK)                            |            |                                                      |                            |                                         |                     |
|                       |                                                              | D1T404P0   | DVC_EVT_ID, MAINT_OBJ_CD, DVC_EVT_REL_OBJ_TYP, E_FLG | Global Partitioned         | RANGE(DVC_EVT_ID)                       |                     |

| Table Name | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name | Index Columns                                                                      | Index Type Global or Local | Index Partitioning Sub-Partitioning Key                   | ILM_DT Initial Load |
|------------|--------------------------------------------------------------|------------|------------------------------------------------------------------------------------|----------------------------|-----------------------------------------------------------|---------------------|
|            |                                                              | D1T404S0   | PK_VALUE1,<br>PK_VALUE2,<br>PK_VALUE3,<br>PK_VALUE4,<br>PK_VALUE5,<br>MAINT_OBJ_CD | Global<br>Partitioned      | HASH(PK_VALUE1,<br>PK_VALUE2,<br>PK_VALUE3,<br>PK_VALUE4) |                     |

### Query for Setting the Retention Period

The following query should be used to create a temporary table to create a mapping table that will identify the retention period for each measuring component type. This table will then be used during in the CTAS operation for Device Event to identify the retention period for each record.

Please refer to [Appendix C: Sample SQL for Enabling ILM with Sub Retention in SGG \(Existing Installation\)](#) for detailed information using Initial Measurement Data as an example.

**Note:** A pre-requisite to executing this query is configuring the appropriate retention periods in the ILM master configuration in the Oracle Utilities Meter Data Management application.

```

CREATE TABLE ILM_DVC_EVT_RETENTION_TMP
AS
select det.dvc_evt_type_cd
/*retrieve the retention period for Device Event Types in this
order of precedence:
1. The category based retention period from the MDM master
configuration
2. The MO level retention period from the MO options
3. The installation level retention period from the FW master
configuration
*/
, CAST(coalesce(catMap.retPeriod --Category level
, (select maint_obj_opt_val
from ci_md_mo_opt mmo
where maint_obj_cd = 'D1-DVCEVENT'
and maint_obj_opt_flg = 'FLRP'
and seq_num = (select max(seq_num)
from ci_md_mo_opt mmo
where maint_obj_cd = 'D1-DVCEVENT'
and maint_obj_opt_flg = 'FLRP')) --MO level
, extractvalue(xmlparse(content
fw_mcfg.mst_config_data),
'generalMasterConfiguration/defaultRetentionPeriod') --Install
level
) as NUMBER(5)) retPeriod
from dl_dvc_evt_type det
, (select extractvalue(value(p),
'deviceEventCategoryRetentionPeriodList/deviceEventCategory')
dvc_evt_cat_flg
, extractvalue(value(p),

```

```

'deviceEventCategoryRetentionPeriodList/retentionPeriod')
retPeriod
from fl_mst_config mdm_mcfg ,
table(xmlsequence(extract(xmlparse(content
mdm_mcfg.mst_config_data),
'deviceEventRetentionPeriod/deviceEventCategoryRetentionPeriods/
deviceEventCategoryRetentionPeriodList'
))) p
where mdm_mcfg.bus_obj_cd = 'D1-ILMMSConfig') catMap
, fl_mst_config fw_mcfg
where fw_mcfg.bus_obj_cd = 'F1-ILMMSConfig'
and det.dvc_evt_cat_flg = catMap.dvc_evt_cat_flg (+)
order by 1;

```

### Initial Measurement Data

If sub retention periods will not be defined for this MO, then please follow the guidelines set forth in section [Module Specific ILM Implementation Details](#).

| Table Name                         | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name          | Index Columns                                                      | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load                 |
|------------------------------------|--------------------------------------------------------------|---------------------|--------------------------------------------------------------------|----------------------------|-----------------------------------------|-------------------------------------|
| D1_INIT_MSRM<br>T_DATA<br>(Parent) | RANGE (ILM_DT,<br>RETENTION_<br>PERIOD)                      |                     |                                                                    |                            |                                         | D1_INIT_MSRM<br>T_DATA.<br>CRE_DTTM |
|                                    |                                                              | D1T304P0            | INIT_MSRMT_DATA_ID                                                 | Global<br>Partitioned      | RANGE<br>(INIT_MSRMT_<br>DATA_ID)       |                                     |
|                                    |                                                              | D1T304S1            | MEASR_COMP_ID,<br>D1_TO_DTTM                                       | Global<br>Partitioned      | RANGE<br>(MEASR_COMP_<br>ID)            |                                     |
|                                    |                                                              | CM_ILM_<br>D1T304S4 | ILM_DT,<br>RETENTION_PERIOD,<br>ILM_ARCH_SW,<br>INIT_MSRMT_DATA_ID | Local                      |                                         |                                     |
| D1_INIT_MSRM<br>T_DATA_CHAR        | REFERENCE<br>(D1_INIT_MSRMT<br>_DATA_CHAR_FK<br>)            |                     |                                                                    |                            |                                         |                                     |
|                                    |                                                              | D1T305P0            | INIT_MSRMT_DATA_ID,<br>CHAR_TYPE_CD,<br>SEQ_NUM                    | Global<br>Partitioned      | RANGE(INIT_MSR<br>MT_DATA_ID)           |                                     |
|                                    |                                                              | D1T305S1            | SRCH_CHAR_VAL                                                      | Global<br>Partitioned      | HASH(SRCH_CHAR<br>_VAL)                 |                                     |
| D1_INIT_MSRM<br>T_DATA_LOG         | REFERENCE<br>(D1_INIT_MSRMT<br>_DATA_LOG_FK)                 |                     |                                                                    |                            |                                         |                                     |
|                                    |                                                              | D1T306P0            | INIT_MSRMT_DATA_ID,<br>SEQNO                                       | Global<br>Partitioned      | RANGE<br>(INIT_MSRMT_DA<br>TA_ID)       |                                     |

| Table Name                  | Table Partitioning Type (Partitioning, Sub-Partitioning Key) | Index Name | Index Columns                       | Index Type Global or Local | Index Partitioning Sub-Partitioning Key | ILM_DT Initial Load |
|-----------------------------|--------------------------------------------------------------|------------|-------------------------------------|----------------------------|-----------------------------------------|---------------------|
| D1_INIT_MSRMT_DATA_LOG_PARM | REFERENCE (D1_INIT_MSRMT_DATA_LOG_PARM_FK)                   |            |                                     |                            |                                         |                     |
|                             |                                                              | D1T307P0   | INIT_MSRMT_DATA_ID, SEQNO, PARM_SEQ | Global Partitioned         | RANGE (INIT_MSRMT_DATA_ID)              |                     |

### Query for Setting the Retention Period

The following query should be used to create a temporary table to create a mapping table that will identify the retention period for each measuring component type. This table will then be used during in the CTAS operation for Initial Measurement Data to identify the retention period for each record.

Please refer to [Appendix B: Sample SQL for Enabling ILM in SGG \(Existing Installation\)](#) for detailed information using Initial Measurement Data as an example.

**Note:** A pre-requisite to executing this query is configuring the appropriate retention periods in the ILM master configuration in the Oracle Utilities Meter Data Management application.

```

CREATE TABLE ILM_IMD_RETENTION_TMP
AS
select mct.measr_comp_type_cd
/*retrieve the retention period for MC Types in this order of
precedence:
1. The UOM based retention period from the MDM master configuration
2. The interval IMD retention period from the MDM master configuration
3. The MO level retention period from the MO options
4. The installation level retention period from the FW master
configuration
*/
, CAST(coalesce((select retPeriod
from (select 'D1IN' interval_scalar_flg
, extractvalue(value(p), 'uomRetentionPeriodList/uom') D1_UOM_CD
, extractvalue(value(p), 'uomRetentionPeriodList/retentionPeriod')
retPeriod
from fl_mst_config mdm_mcfg
, table(xmlsequence(extract(xmlparse(content
mdm_mcfg.mst_config_data),
'imdRetentionPeriod/intervalImdRetentionPeriods/uomRetentionPeriods/
uomRetentionPeriodList')) p
where mdm_mcfg.bus_obj_cd = 'D1-ILMMSConfig'
union
select 'D1SC' INTERVAL_SCALAR_FLG
, extractvalue(value(p), 'uomRetentionPeriodList/uom') D1_UOM_CD
, extractvalue(value(p), 'uomRetentionPeriodList/retentionPeriod')
retPeriod
from fl_mst_config mdm_mcfg
, table(xmlsequence(extract(xmlparse(content
mdm_mcfg.mst_config_data),

```

```

'imdRetentionPeriod/scalarImdRetentionPeriods/uomRetentionPeriods/
uomRetentionPeriodList')) p
where mdm_mcfg.bus_obj_cd = 'D1-ILMMSConfig') uomMap
where uomMap.interval_scalar_flg = mct.interval_scalar_flg
and trim(mctvi.dl_uom_cd) = trim(uomMap.dl_uom_cd) --UOM
, DECODE(mct.interval_scalar_flg
,'D1IN'
,extractvalue(xmlparse(content mdm_mcfg.mst_config_data),
'imdRetentionPeriod/intervalImdRetentionPeriods/
intervalRetentionPeriod') --interval IMD
,extractvalue(xmlparse(content mdm_mcfg.mst_config_data),
'imdRetentionPeriod/scalarImdRetentionPeriods/scalarRetentionPeriod')
--scalar IMD
)
, (select maint_obj_opt_val
from ci_md_mo_opt mmo
where maint_obj_cd = 'D1-IMD'
and maint_obj_opt_flg = 'FLRP'
and seq_num = (select max(seq_num)
from ci_md_mo_opt mmo
where maint_obj_cd = 'D1-IMD'
and maint_obj_opt_flg = 'FLRP')) --IMD
, extractvalue(xmlparse(content fw_mcfg.mst_config_data),
'generalMasterConfiguration/defaultRetentionPeriod') --Install
) as NUMBER(5)) retPeriod
from dl_measr_comp_type mct
, dl_mc_type_value_identifier mctvi
, fl_mst_config fw_mcfg
, fl_mst_config mdm_mcfg
where mct.measr_comp_type_cd = mctvi.measr_comp_type_cd
and mctvi.value_id_type_flg = 'D1MS'
and fw_mcfg.bus_obj_cd = 'F1-ILMMSConfig'
and mdm_mcfg.bus_obj_cd = 'D1-ILMMSConfig'
order by 1;

```

## On-going Maintenance Phase

The following steps provide a high level overview of what needs to be done for on-going maintenance for ILM on enabled MOs.

Please refer to the [Appendix D: Sample SQL for Periodic Maintenance](#) for detailed information using To Do Entry(Without LOB), F1\_SYNC\_REC\_IN(With LOB-Tablespace per Partition), Initial Measurement Data (With LOB-Tablespace per Subpartition), and the D1\_MSRMT table (Partition Compression) as examples.

1. Add the partition:
  - a. Create Tablespace to be used for the new parent table partition.
  - b. Since, we define MAXVALUE Partition; new partition can only be created using “SPLIT” operation. Identify and use next HIGH\_VALUE Partition for the split operation.
  - c. All the child table(s) partition(s)\LOB(s) must be altered to use the same tablespace as that of the parent table’s partition.
  - d. Enable advanced compression on all child table(s).
  - e. Copy partition level statistics from the previous partition.
2. Archive the partition/subpartition:

- a. Make the tablespace that will be archived READ ONLY.
  - b. Check that no records have ILM\_ARCH\_SW = 'N'.
    - If record count is zero, then proceed for further steps.
    - If record count is not zero, then change the tablespace back to READ WRITE MODE as Archive is not Feasible at the time.
  - c. Create an archive tablespace for the partition/subpartition that needs to be archived.
  - d. Create staging tables using the new archive tablespace. Load data for all child tables first.
  - e. Create staging table using the new archive tablespace and load data for the parent table.
  - f. Export tablespace using TRANSPORT\_TABLESPACES method.
 

Make Sure Tablespace datafile required for further import is preserved.
  - g. Drop the partition, partition the tablespace and archive the tablespace (as it is already exported).
3. Restore the partition:
    - a. Create a new tablespace to restore the partition/subpartition.
    - b. Add partition using split operation on next greater high value partition.
 

If the table contains LOBS, there will an additional statement in split partition DDL indicating tablespace where the LOBs will be stored.
    - c. Enable advanced compression on all child table(s).
    - d. Import Tablespace using TRANSPORT\_TABLESPACES method.
    - e. Load data into the parent table first from the staging table.
    - f. Load data into the child table from the staging table.
    - g. Drop the archive tablespace after import and data loading is successful.
4. Compress D1\_MSRMT table Partition:
    - a. Create new compressed tablespace.
    - b. Create a table using CTAS for each subpartition of the partition being compressed in the new compressed tablespace.
    - c. Create a unique primary index for each subpartition of the partition being compressed in the new compressed tablespace. Then alter table to create the primary key.
    - d. Exchange the subpartition of the D1\_MSRMT table with the newly created table for each subpartition.
    - e. Drop the original uncompressed tablespace.
    - f. Alter the partition level metadata to reflect the new compressed tablespace.
    - g. Rename the new compressed tablespace to the original tablespace name.
  5. Move Data between different storage tiers:

The ILM facilities can be used within the database to implement storage savings, as follows:

- Use ILM Assistant to define the data groups to be used for the individual objects. Assign those data groups to partitions and storage devices to implement the storage savings. Remember to assign transportable tablespaces for the archive/dormant data stage to allow for safe removal of the data.
- Use ILM assistant to generate the necessary commands to implement the data changes manually or use Automatic Storage Management (ASM) to automate the data storage policies.
- Optionally, use Automatic Data Optimization to provide further optimizations.

For more information about ILM Assistant and ILM refer to the following:

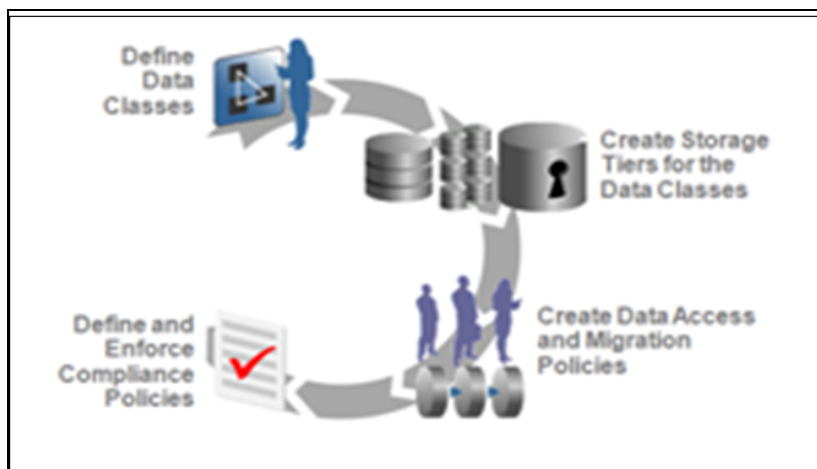
- [ILM Assistant Users Guide](#)
- Oracle Database VLDB and Partitioning Guide (11.2) available at: [http://docs.oracle.com/cd/E11882\\_01/server.112/e25523/part\\_lifecycle.htm#CACECAFB](http://docs.oracle.com/cd/E11882_01/server.112/e25523/part_lifecycle.htm#CACECAFB)
- Oracle Database VLDB and Partitioning Guide (12.1) available at: <https://docs.oracle.com/database/121/VLDBG/title.htm>

## **ILM Assistant**

The ILM Assistant can provide the following:

- Setup ILM Lifecycle definition - Here you can define different lifecycle definitions for different MOs and configure when the data is ready to be moved to a slower disk.
- Setup ILM Lifecycle tables - Here you define the tables you want to manage and assign it to a Lifecycle definition defined above. You can setup policies so that when data is moved from one partition to another it will be automatically compressed to a desired degree.
- Lifecycle Management - There is a tab called Lifecycle Management where the system admin will be alerted when partitions are eligible for archiving.

ILM Assistant can then be used to ensure the records that have ILM\_ARCH\_SW = 'Y' can be archived or purged, as deemed appropriate by the business.



**Note:** For further guidelines on ILM Assistant refer to [Implementing Information Lifecycle Management Using the ILM Assistant](#).

## Naming Convention

The naming convention for tablespace, partitions & subpartition is standardized as follows:

- Each name consists of some or all of the following parts.
- The parts of the name are organized hierarchically.
- Each part of the Name is separated with an underscore.
- The maximum name length must not exceed 30 Characters.
- For an MO, the parent table and child table share the same tablespace for the corresponding partition (or sub partition as appropriate).
- Square brackets [ ] indicate that this part of the name should be omitted if not required.

OWNERFLAG\_TABLEIDENTIFIER\_PARTITIONNAME[\_SUBPARTITIONNAME][\_ARCHIVEFLAG][\_COMPRESSFLAG]

For details on the convention, please refer to the table below:

| Convention       | Description                                                                                                                                           |
|------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| OWNERFLAG        | Owner flag for the relevant application for example “D1” for MDM                                                                                      |
| TABLE IDENTIFIER | The Index Name of the Primary Key index without the “P0” suffix. For example, if the PK index name is XT039P0, the table identifier would be “XT039”. |



---

| Convention           | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| PARTITION<br>NAME    | The Partition name should be prefixed with a P followed by a name which conforms to one of the following standards: <ul style="list-style-type: none"><li>• 4 digit year and 3 letter month abbreviation PYYYYMON corresponding to the ILM date e.g. P2011JAN</li><li>• PMAX if it is the Max Value partition</li></ul>                                                                                                                                                                                                                                                          |
| SUBPARTITION<br>NAME | If subpartitions are used, name should be prefixed with S followed by a name of not more than 5 characters which conforms to the following requirements: <ul style="list-style-type: none"><li>• SMAX if this is the Max Value sub partition</li><li>• If the sub partition holds data for a sub retention period use a number equal to that period e.g S91 if the sub retention period &lt; 91 days.</li><li>• For a range based SubPartition on Primary Key, use an integral number increasing by +1. For example, if there are 8 sub partitions use S01 through S08</li></ul> |
| ARCHIVEFLAG          | This flag is used as a suffix to the table and tablespace name for the staging tables created for the archiving operation. <ul style="list-style-type: none"><li>• ARC</li></ul>                                                                                                                                                                                                                                                                                                                                                                                                 |

---

| Convention       | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| COMPRESS<br>FLAG | <p>This flag is used as a suffix to the tablespace name for the staging tables created when compressing a partition.</p> <ul style="list-style-type: none"> <li>• C</li> </ul> <p>For compression related tasks, this is used as suffix to the tablespace name.</p> <ul style="list-style-type: none"> <li>• Partition Tablespace Name: It is formed by<br/>OWNERFLAG_TABLEIDENTIFIER_PARTTIONNAME</li> </ul> <p>For example:<br/>CM_D1T304_PMAX<br/>CM_D1T304_P2011JAN</p> <ul style="list-style-type: none"> <li>• SubPartition Tablespace Name: It is formed by<br/>OWNERFLAG_TABLEIDENTIFIER_PARTTIONNAME<br/>_SUBPARTTIONNAME</li> </ul> <p>For example:<br/>CM_D1T304_PMAX_SMAX<br/>CM_D1T304_P2011JAN_SMAX<br/>CM_D1T304_PMAX_S001<br/>CM_D1T304_P2011JAN_S181</p> <ul style="list-style-type: none"> <li>• Archive Staging Table And Its Tablespace Name (When archiving partition): It is formed by<br/>OWNERFLAG_TABLEIDENTIFIER_PARTTIONNAME<br/>_ARCHIVEFLAG.</li> </ul> <p>For example:<br/>CM_D1T304_P2011JAN_ARC</p> <ul style="list-style-type: none"> <li>• Archive Staging Table And Its Tablespace Name (When archiving subpartition): It is formed by<br/>OWNERFLAG_TABLEIDENTIFIER_PARTTIONNAME<br/>_SUBPARTTIONNAME_ARCHIVEFLAG.</li> </ul> <p>For example:<br/>CM_D1T304_P2011JAN_S181_ARC</p> <ul style="list-style-type: none"> <li>• Compressed Tablespace name (When compressing partition):<br/>For example:<br/>CM_D1T304_P2011JAN_C</li> </ul> |

# Appendix A

## Sample SQL for Enabling ILM in SGG (Initial Installation)

This section provides more detail about steps needed to fully support ILM on tables for maintenance objects that support the functionality.

Three maintenance objects are shown:

- To Do Entry - does not include a LOB field.
- Sync Request - does include a LOB field and has one tablespace per partition.
- Initial Measurement Data - includes LOB fields and has one tablespace per subpartition (shown using subretention). Other maintenance object's implementations can follow the appropriate pattern based on whether there is a LOB field or not.

The following DDL(s):

- Follows Naming convention recommendations for partitions\subpartitions\tablespaces.
- Ensures all the ILM Storage requirements are incorporated, failing which, ILM functionality will not be achieved.
  - Partitions/subpartitions are defined with respective Tablespace.
  - Child Tables are referenced partitioned.
- Ensures all Compression recommendations are incorporated.

### Maintenance Object: TO DO ENTRY

#### Parent Table: CI\_TD\_ENTRY

```
CREATE BIGFILE TABLESPACE CM_XT039_P2011JAN DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_XT039_P2011FEB DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_XT039_P2011MAR DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_XT039_P2011APR DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_XT039_P2011MAY DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_XT039_P2011JUN DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
```

```

CREATE BIGFILE TABLESPACE CM_XT039_P2011JUL DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_XT039_P2011AUG DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_XT039_P2011SEP DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_XT039_P2011OCT DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_XT039_P2011NOV DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_XT039_P2011DEC DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_XT039_PMAX DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON MAXSIZE
UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;

```

```

CREATE TABLE CI_TD_ENTRY (
 TD_ENTRY_ID CHAR(14) NOT NULL ENABLE,
 BATCH_CD CHAR(8) DEFAULT ' ' NOT NULL ENABLE,
 BATCH_NBR NUMBER(10,0) DEFAULT 0 NOT NULL ENABLE,
 MESSAGE_CAT_NBR NUMBER(5,0) DEFAULT 0 NOT NULL ENABLE,
 MESSAGE_NBR NUMBER(5,0) DEFAULT 0 NOT NULL ENABLE,
 ASSIGNED_TO CHAR(8) DEFAULT ' ' NOT NULL ENABLE,
 TD_TYPE_CD CHAR(8) DEFAULT ' ' NOT NULL ENABLE,
 ROLE_ID CHAR(10) DEFAULT ' ' NOT NULL ENABLE,
 ENTRY_STATUS_FLG CHAR(2) DEFAULT ' ' NOT NULL ENABLE,
 VERSION NUMBER(5,0) DEFAULT 1 NOT NULL ENABLE,
 CRE_DTTM DATE,
 ASSIGNED_DTTM DATE,
 COMPLETE_DTTM DATE,
 COMPLETE_USER_ID CHAR(8) DEFAULT ' ' NOT NULL ENABLE,
 COMMENTS VARCHAR2(254) DEFAULT ' ' NOT NULL ENABLE,
 ASSIGNED_USER_ID CHAR(8) DEFAULT ' ' NOT NULL ENABLE,
 TD_PRIORITY_FLG CHAR(4) DEFAULT ' ' NOT NULL ENABLE,
 ILM_DT DATE,
 ILM_ARCH_SW CHAR(1)
)
ENABLE ROW MOVEMENT
PARTITION BY RANGE (ILM_DT)
SUBPARTITION BY RANGE (TD_ENTRY_ID) SUBPARTITION TEMPLATE
(
 SUBPARTITION S01 VALUES LESS THAN ('1249999999999999'),
 SUBPARTITION S02 VALUES LESS THAN ('2499999999999999'),
 SUBPARTITION S03 VALUES LESS THAN ('3749999999999999'),
 SUBPARTITION S04 VALUES LESS THAN ('4999999999999999'),
 SUBPARTITION S05 VALUES LESS THAN ('6249999999999999'),
 SUBPARTITION S06 VALUES LESS THAN ('7499999999999999'),
 SUBPARTITION S07 VALUES LESS THAN ('8749999999999999'),
 SUBPARTITION SMAX VALUES LESS THAN (MAXVALUE)
)
(
 PARTITION "P2011JAN" VALUES LESS THAN (TO_DATE('2011-02-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
 TABLESPACE CM_XT039_P2011JAN,
 PARTITION "P2011FEB" VALUES LESS THAN (TO_DATE('2011-03-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
 TABLESPACE CM_XT039_P2011FEB,
 PARTITION "P2011MAR" VALUES LESS THAN (TO_DATE('2011-04-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
 TABLESPACE CM_XT039_P2011MAR,
 PARTITION "P2011APR" VALUES LESS THAN (TO_DATE('2011-05-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
 TABLESPACE CM_XT039_P2011APR,
 PARTITION "P2011MAY" VALUES LESS THAN (TO_DATE('2011-06-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
 TABLESPACE CM_XT039_P2011MAY,
 PARTITION "P2011JUN" VALUES LESS THAN (TO_DATE('2011-07-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
 TABLESPACE CM_XT039_P2011JUN,
 PARTITION "P2011JUL" VALUES LESS THAN (TO_DATE('2011-08-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
 TABLESPACE CM_XT039_P2011JUL,
 PARTITION "P2011AUG" VALUES LESS THAN (TO_DATE('2011-09-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
 TABLESPACE CM_XT039_P2011AUG,
 PARTITION "P2011SEP" VALUES LESS THAN (TO_DATE('2011-10-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
 TABLESPACE CM_XT039_P2011SEP,
 PARTITION "P2011OCT" VALUES LESS THAN (TO_DATE('2011-11-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
 TABLESPACE CM_XT039_P2011OCT,
 PARTITION "P2011NOV" VALUES LESS THAN (TO_DATE('2011-12-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
 TABLESPACE CM_XT039_P2011NOV,
 PARTITION "PMAX" VALUES LESS THAN (MAXVALUE)
 TABLESPACE CM_XT039_PMAX
)

```

```
);
```

## INDEX

```
CREATE BIGFILE TABLESPACE CM_XT039_IND DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON MAXSIZE
UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
```

```
CREATE UNIQUE INDEX XT039P0 ON CI_TD_ENTRY (TD_ENTRY_ID) TABLESPACE CM_XT039_IND
GLOBAL PARTITION BY RANGE (TD_ENTRY_ID)
```

```
(
PARTITION P1 VALUES LESS THAN ('12499999999999'),
PARTITION P2 VALUES LESS THAN ('24999999999999'),
PARTITION P3 VALUES LESS THAN ('37499999999999'),
PARTITION P4 VALUES LESS THAN ('49999999999999'),
PARTITION P5 VALUES LESS THAN ('62499999999999'),
PARTITION P6 VALUES LESS THAN ('74999999999999'),
PARTITION P7 VALUES LESS THAN ('87499999999999'),
PARTITION P8 VALUES LESS THAN (MAXVALUE)
);
```

```
ALTER TABLE CI_TD_ENTRY ADD CONSTRAINT XT039P0 PRIMARY KEY(TD_ENTRY_ID) USING INDEX;
```

```
CREATE UNIQUE INDEX XT039S2 ON CI_TD_ENTRY (ASSIGNED_TO, TD_ENTRY_ID) TABLESPACE
CM_XT039_IND COMPRESS ADVANCED LOW;
```

```
CREATE INDEX XT039S3 ON CI_TD_ENTRY (ENTRY_STATUS_FLG, ASSIGNED_TO) TABLESPACE
CM_XT039_IND COMPRESS ADVANCED LOW;
```

```
CREATE INDEX XT039S4 ON CI_TD_ENTRY (ROLE_ID, TD_TYPE_CD, ENTRY_STATUS_FLG,
TD_PRIORITY_FLG, ASSIGNED_TO, CRE_DTTM) TABLESPACE CM_XT039_IND COMPRESS ADVANCED LOW;
```

```
CREATE INDEX XT039S5 ON CI_TD_ENTRY (BATCH_CD, BATCH_NBR, ENTRY_STATUS_FLG) TABLESPACE
CM_XT039_IND COMPRESS ADVANCED LOW;
```

```
CREATE UNIQUE INDEX XT039S6 ON CI_TD_ENTRY (TD_ENTRY_ID, ASSIGNED_TO, ENTRY_STATUS_FLG)
TABLESPACE CM_XT039_IND COMPRESS ADVANCED LOW;
```

```
CREATE UNIQUE INDEX XT039S7 ON CI_TD_ENTRY (COMPLETE_USER_ID, COMPLETE_DTTM, TD_ENTRY_ID)
TABLESPACE CM_XT039_IND COMPRESS ADVANCED LOW;
```

```
CREATE INDEX XT039S8 ON CI_TD_ENTRY (ENTRY_STATUS_FLG, TD_TYPE_CD, MESSAGE_CAT_NBR,
MESSAGE_NBR) TABLESPACE CM_XT039_IND COMPRESS ADVANCED LOW;
```

```
CREATE UNIQUE INDEX CM_ILM_XT039S8 ON CI_TD_ENTRY (ILM_DT, ILM_ARCH_SW, TD_ENTRY_ID)
LOCAL COMPRESS ADVANCED LOW;
```

## Child Table: CI\_TD\_DRLKEY

```
CREATE TABLE CI_TD_DRLKEY
(
TD_ENTRY_ID CHAR(14) NOT NULL ENABLE,
SEQ_NUM NUMBER(3,0) NOT NULL ENABLE,
KEY_VALUE VARCHAR2(50 BYTE) DEFAULT ' ' NOT NULL ENABLE,
VERSION NUMBER(5,0) DEFAULT 1 NOT NULL ENABLE,
CONSTRAINT CI_TD_DRLKEY_FK FOREIGN KEY(TD_ENTRY_ID) REFERENCES CI_TD_ENTRY
ON DELETE CASCADE)
PARTITION BY REFERENCE (CI_TD_DRLKEY_FK)
ENABLE ROW MOVEMENT;
```

## INDEX

```
CREATE UNIQUE INDEX XT037P0 ON CI_TD_DRLKEY (TD_ENTRY_ID, SEQ_NUM) TABLESPACE
CM_XT039_IND
GLOBAL PARTITION BY RANGE (TD_ENTRY_ID)
```

```
(
PARTITION P1 VALUES LESS THAN ('12499999999999'),
PARTITION P2 VALUES LESS THAN ('24999999999999'),
PARTITION P3 VALUES LESS THAN ('37499999999999'),
PARTITION P4 VALUES LESS THAN ('49999999999999'),
PARTITION P5 VALUES LESS THAN ('62499999999999'),
PARTITION P6 VALUES LESS THAN ('74999999999999'),
PARTITION P7 VALUES LESS THAN ('87499999999999'),
PARTITION P8 VALUES LESS THAN (MAXVALUE)
)
```

```
COMPRESS ADVANCED LOW;
```

```
ALTER TABLE CI_TD_DRLKEY ADD CONSTRAINT XT037P0 PRIMARY KEY(TD_ENTRY_ID, SEQ_NUM) USING
INDEX;
```

```
CREATE INDEX XT037S1 ON CI_TD_DRLKEY (KEY_VALUE, TD_ENTRY_ID) TABLESPACE CM_XT039_IND
COMPRESS ADVANCED LOW;
```

## Child Table: CI\_TD\_ENTRY\_CHA

```
CREATE TABLE CI_TD_ENTRY_CHA
(
 TD_ENTRY_ID CHAR(14) NOT NULL ENABLE,
 CHAR_TYPE_CD CHAR(8) NOT NULL ENABLE,
 SEQ_NUM NUMBER(3,0) DEFAULT 0 NOT NULL ENABLE,
 CHAR_VAL CHAR(16) DEFAULT ' ' NOT NULL ENABLE,
 VERSION NUMBER(5,0) DEFAULT 1 NOT NULL ENABLE,
 ADHOC_CHAR_VAL VARCHAR2(254) DEFAULT ' ' NOT NULL ENABLE,
 CHAR_VAL_FK1 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
 CHAR_VAL_FK2 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
 CHAR_VAL_FK3 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
 CHAR_VAL_FK4 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
 CHAR_VAL_FK5 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
 SRCH_CHAR_VAL VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
 CONSTRAINT CI_TD_ENTRY_CHA_FK FOREIGN KEY(TD_ENTRY_ID) REFERENCES CI_TD_ENTRY
ON DELETE CASCADE)
PARTITION BY REFERENCE (CI_TD_ENTRY_CHA_FK)
ENABLE ROW MOVEMENT;
```

## INDEX

```
CREATE UNIQUE INDEX XT701P0 ON CI_TD_ENTRY_CHA (TD_ENTRY_ID, CHAR_TYPE_CD, SEQ_NUM)
TABLESPACE CM_XT039_IND
GLOBAL PARTITION BY RANGE (TD_ENTRY_ID)
(
 PARTITION P1 VALUES LESS THAN ('12499999999999'),
 PARTITION P2 VALUES LESS THAN ('24999999999999'),
 PARTITION P3 VALUES LESS THAN ('37499999999999'),
 PARTITION P4 VALUES LESS THAN ('49999999999999'),
 PARTITION P5 VALUES LESS THAN ('62499999999999'),
 PARTITION P6 VALUES LESS THAN ('74999999999999'),
 PARTITION P7 VALUES LESS THAN ('87499999999999'),
 PARTITION P8 VALUES LESS THAN (MAXVALUE)
)
COMPRESS ADVANCED LOW;

ALTER TABLE CI_TD_ENTRY_CHA ADD CONSTRAINT XT701P0 PRIMARY KEY(TD_ENTRY_ID, CHAR_TYPE_CD,
SEQ_NUM) USING INDEX;

CREATE INDEX XT701S1 ON CI_TD_ENTRY_CHA (SRCH_CHAR_VAL, CHAR_TYPE_CD, TD_ENTRY_ID)
TABLESPACE CM_XT039_IND COMPRESS ADVANCED LOW;

CREATE INDEX XT701S2 ON CI_TD_ENTRY_CHA (CHAR_VAL_FK1) TABLESPACE CM_XT039_IND
COMPRESS ADVANCED LOW;
```

## Child Table: CI\_TD\_LOG

```
CREATE TABLE CI_TD_LOG
(
 TD_ENTRY_ID CHAR(14) NOT NULL ENABLE,
 SEQ_NUM NUMBER(3,0) NOT NULL ENABLE,
 LOG_DTTM DATE NOT NULL ENABLE,
 LOG_TYPE_FLG CHAR(4) DEFAULT ' ' NOT NULL ENABLE,
 USER_ID CHAR(8) DEFAULT ' ' NOT NULL ENABLE,
 ASSIGNED_TO CHAR(8) DEFAULT ' ' NOT NULL ENABLE,
 VERSION NUMBER(5,0) DEFAULT 1 NOT NULL ENABLE,
 DESCRLONG VARCHAR2(4000) DEFAULT ' ' NOT NULL ENABLE,
 CONSTRAINT CI_TD_LOG_FK FOREIGN KEY(TD_ENTRY_ID) REFERENCES CI_TD_ENTRY ON DELETE CASCADE)
PARTITION BY REFERENCE (CI_TD_LOG_FK)
ENABLE ROW MOVEMENT;
```

## INDEX

```
CREATE UNIQUE INDEX XT721P0 ON CI_TD_LOG (TD_ENTRY_ID, SEQ_NUM) TABLESPACE CM_XT039_IND
GLOBAL PARTITION BY RANGE (TD_ENTRY_ID)
(
 PARTITION P1 VALUES LESS THAN ('12499999999999'),
 PARTITION P2 VALUES LESS THAN ('24999999999999'),
 PARTITION P3 VALUES LESS THAN ('37499999999999'),
 PARTITION P4 VALUES LESS THAN ('49999999999999'),

```

```

PARTITION P5 VALUES LESS THAN ('62499999999999'),
PARTITION P6 VALUES LESS THAN ('74999999999999'),
PARTITION P7 VALUES LESS THAN ('87499999999999'),
PARTITION P8 VALUES LESS THAN (MAXVALUE)
)
COMPRESS ADVANCED LOW;

ALTER TABLE CI_TD_LOG ADD CONSTRAINT XT721P0 PRIMARY KEY(TD_ENTRY_ID, SEQ_NUM) USING INDEX;

CREATE INDEX XT721S1 ON CI_TD_LOG (LOG_DTTM, USER_ID, LOG_TYPE_FLG, TD_ENTRY_ID)
TABLESPACE CM_XT039_IND COMPRESS ADVANCED LOW;

```

## Child Table: CI\_TD\_MSG\_PARM

```

CREATE TABLE CI_TD_MSG_PARM
(
 TD_ENTRY_ID CHAR(14) NOT NULL ENABLE,
 SEQ_NUM NUMBER(3,0) NOT NULL ENABLE,
 MSG_PARM_VAL VARCHAR2(2000) DEFAULT ' ' NOT NULL ENABLE,
 VERSION NUMBER(5,0) DEFAULT 1 NOT NULL ENABLE,
 CONSTRAINT CI_TD_MSG_PARM_FK FOREIGN KEY(TD_ENTRY_ID) REFERENCES CI_TD_ENTRY ON DELETE
 CASCADE)
PARTITION BY REFERENCE (CI_TD_MSG_PARM_FK)
ENABLE ROW MOVEMENT;

```

### INDEX

```

CREATE UNIQUE INDEX XT040P0 ON CI_TD_MSG_PARM (TD_ENTRY_ID, SEQ_NUM) TABLESPACE
CM_XT039_IND
GLOBAL PARTITION BY RANGE (TD_ENTRY_ID)
(
 PARTITION P1 VALUES LESS THAN ('12499999999999'),
 PARTITION P2 VALUES LESS THAN ('24999999999999'),
 PARTITION P3 VALUES LESS THAN ('37499999999999'),
 PARTITION P4 VALUES LESS THAN ('49999999999999'),
 PARTITION P5 VALUES LESS THAN ('62499999999999'),
 PARTITION P6 VALUES LESS THAN ('74999999999999'),
 PARTITION P7 VALUES LESS THAN ('87499999999999'),
 PARTITION P8 VALUES LESS THAN (MAXVALUE)
)
COMPRESS ADVANCED LOW;

ALTER TABLE CI_TD_MSG_PARM ADD CONSTRAINT XT040P0 PRIMARY KEY(TD_ENTRY_ID, SEQ_NUM) USING
INDEX;

```

## Child Table: CI\_TD\_SRTKEY

```

CREATE TABLE CI_TD_SRTKEY
(
 TD_ENTRY_ID CHAR(14) NOT NULL ENABLE,
 SEQ_NUM NUMBER(3,0) NOT NULL ENABLE,
 KEY_VALUE VARCHAR2(50 BYTE) DEFAULT ' ' NOT NULL ENABLE,
 VERSION NUMBER(5,0) DEFAULT 1 NOT NULL ENABLE,
 CONSTRAINT CI_TD_SRTKEY_FK FOREIGN KEY(TD_ENTRY_ID) REFERENCES CI_TD_ENTRY ON DELETE
 CASCADE)
PARTITION BY REFERENCE (CI_TD_SRTKEY_FK)
ENABLE ROW MOVEMENT;

```

### INDEX

```

CREATE UNIQUE INDEX XT041P0 ON CI_TD_SRTKEY (TD_ENTRY_ID, SEQ_NUM) TABLESPACE
CM_XT039_IND
GLOBAL PARTITION BY RANGE (TD_ENTRY_ID)
(
 PARTITION P1 VALUES LESS THAN ('12499999999999'),
 PARTITION P2 VALUES LESS THAN ('24999999999999'),
 PARTITION P3 VALUES LESS THAN ('37499999999999'),
 PARTITION P4 VALUES LESS THAN ('49999999999999'),
 PARTITION P5 VALUES LESS THAN ('62499999999999'),
 PARTITION P6 VALUES LESS THAN ('74999999999999'),
 PARTITION P7 VALUES LESS THAN ('87499999999999'),
 PARTITION P8 VALUES LESS THAN (MAXVALUE)
)
COMPRESS ADVANCED LOW;

```

```
ALTER TABLE CI_TD_SRTKEY ADD CONSTRAINT XT041P0 PRIMARY KEY(TD_ENTRY_ID, SEQ_NUM) USING
INDEX;

CREATE INDEX XT041S1 ON CI_TD_SRTKEY (KEY_VALUE, TD_ENTRY_ID) TABLESPACE CM_XT039_IND
COMPRESS ADVANCED LOW;
```

## Maintenance Object:F1-SYNCREQIN

### Parent Table: F1\_SYNC\_REQ\_IN

```
CREATE BIGFILE TABLESPACE CM_F1T191_P2011JAN DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_F1T191_P2011FEB DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_F1T191_P2011MAR DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_F1T191_P2011APR DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_F1T191_P2011MAY DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_F1T191_P2011JUN DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_F1T191_P2011JUL DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_F1T191_P2011AUG DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_F1T191_P2011SEP DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_F1T191_P2011OCT DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_F1T191_P2011NOV DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_F1T191_P2011DEC DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_F1T191_PMAX DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON MAXSIZE
UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;

CREATE TABLE F1_SYNC_REQ_IN
(
 F1_SYNC_REQ_IN_ID CHAR(14) NOT NULL ENABLE,
 BUS_OBJ_CD CHAR(30) DEFAULT ' ' NOT NULL ENABLE,
 CRE_DTTM DATE NOT NULL ENABLE,
 BO_STATUS_CD CHAR(12) DEFAULT ' ' NOT NULL ENABLE,
 STATUS_UPD_DTTM DATE,
 MAINT_OBJ_CD CHAR(12 BYTE) DEFAULT ' ' NOT NULL ENABLE,
 NT_XID_CD CHAR(30 BYTE) DEFAULT ' ' NOT NULL ENABLE,
 EXT_PK_VALUE1 VARCHAR2(254) DEFAULT ' ' NOT NULL ENABLE,
 EXT_PK_VALUE2 VARCHAR2(254) DEFAULT ' ' NOT NULL ENABLE,
 EXT_PK_VALUE3 VARCHAR2(254) DEFAULT ' ' NOT NULL ENABLE,
 EXT_PK_VALUE4 VARCHAR2(254) DEFAULT ' ' NOT NULL ENABLE,
 EXT_PK_VALUE5 VARCHAR2(254) DEFAULT ' ' NOT NULL ENABLE,
 PK_VALUE1 VARCHAR2(254) DEFAULT ' ' NOT NULL ENABLE,
 BO_DATA_AREA CLOB,
 PRE_TRN_INIT_BO_DATA_AREA CLOB,
 PRE_TRN_FIN_BO_DATA_AREA CLOB,
 POST_TRN_BO_DATA_AREA CLOB,
 VERSION NUMBER(5,0) DEFAULT 1 NOT NULL ENABLE,
 EXT_REFERENCE_ID CHAR(36) DEFAULT ' ' NOT NULL ENABLE,
 F1_INITIAL_LOAD_SYNC_FLG CHAR(4) DEFAULT ' ' NOT NULL ENABLE,
 F1_COMPOSITE_SYNC_FLG CHAR(4) DEFAULT ' ' NOT NULL ENABLE,
 ILM_DT DATE,
 ILM_ARCH_SW CHAR(1)
)
ENABLE ROW MOVEMENT
LOB (BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE)
LOB (PRE_TRN_INIT_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS
MEDIUM CACHE)
LOB (PRE_TRN_FIN_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS
MEDIUM CACHE)
LOB (POST_TRN_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE)
PARTITION BY RANGE(ILM_DT)
SUBPARTITION BY RANGE(F1_SYNC_REQ_IN_ID)
SUBPARTITION TEMPLATE
(
 SUBPARTITION S01 VALUES LESS THAN ('12499999999999'),
 SUBPARTITION S02 VALUES LESS THAN ('24999999999999'),
```



```

SUBPARTITION S03 VALUES LESS THAN ('37499999999999'),
SUBPARTITION S04 VALUES LESS THAN ('49999999999999'),
SUBPARTITION S05 VALUES LESS THAN ('62499999999999'),
SUBPARTITION S06 VALUES LESS THAN ('74999999999999'),
SUBPARTITION S07 VALUES LESS THAN ('87499999999999'),
SUBPARTITION SMAX VALUES LESS THAN (MAXVALUE)
)
(
PARTITION "P2011JAN" VALUES LESS THAN (TO_DATE('2011-02-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
LOB(BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE
TABLESPACE CM_F1T191_P2011JAN)
LOB(PRE_TRN_INIT_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE TABLESPACE CM_F1T191_P2011JAN)
LOB(PRE_TRN_FIN_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE TABLESPACE CM_F1T191_P2011JAN)
LOB(POST_TRN_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE
TABLESPACE CM_F1T191_P2011JAN)
TABLESPACE CM_F1T191_P2011JAN,
PARTITION "P2011FEB" VALUES LESS THAN (TO_DATE('2011-03-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
LOB(BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE
TABLESPACE CM_F1T191_P2011FEB)
LOB(PRE_TRN_INIT_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE TABLESPACE CM_F1T191_P2011FEB)
LOB(PRE_TRN_FIN_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE TABLESPACE CM_F1T191_P2011FEB)
LOB(POST_TRN_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE
TABLESPACE CM_F1T191_P2011FEB)
TABLESPACE CM_F1T191_P2011FEB,
PARTITION "P2011MAR" VALUES LESS THAN (TO_DATE('2011-04-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
LOB(BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE
TABLESPACE CM_F1T191_P2011MAR)
LOB(PRE_TRN_INIT_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE TABLESPACE CM_F1T191_P2011MAR)
LOB(PRE_TRN_FIN_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE TABLESPACE CM_F1T191_P2011MAR)
LOB(POST_TRN_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE
TABLESPACE CM_F1T191_P2011MAR)
TABLESPACE CM_F1T191_P2011MAR,
PARTITION "P2011APR" VALUES LESS THAN (TO_DATE('2011-05-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
LOB(BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE
TABLESPACE CM_F1T191_P2011APR)
LOB(PRE_TRN_INIT_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE TABLESPACE CM_F1T191_P2011APR)
LOB(PRE_TRN_FIN_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE TABLESPACE CM_F1T191_P2011APR)
LOB(POST_TRN_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE
TABLESPACE CM_F1T191_P2011APR)
TABLESPACE CM_F1T191_P2011APR,
PARTITION "P2011MAY" VALUES LESS THAN (TO_DATE('2011-06-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
LOB(BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE
TABLESPACE CM_F1T191_P2011MAY)
LOB(PRE_TRN_INIT_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE TABLESPACE CM_F1T191_P2011MAY)
LOB(PRE_TRN_FIN_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE TABLESPACE CM_F1T191_P2011MAY)
LOB(POST_TRN_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE
TABLESPACE CM_F1T191_P2011MAY)
TABLESPACE CM_F1T191_P2011MAY,
PARTITION "P2011JUN" VALUES LESS THAN (TO_DATE('2011-07-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
LOB(BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE
TABLESPACE CM_F1T191_P2011JUN)
LOB(PRE_TRN_INIT_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE TABLESPACE CM_F1T191_P2011JUN)
LOB(PRE_TRN_FIN_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE TABLESPACE CM_F1T191_P2011JUN)
LOB(POST_TRN_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE
TABLESPACE CM_F1T191_P2011JUN)
TABLESPACE CM_F1T191_P2011JUN,
PARTITION "P2011JUL" VALUES LESS THAN (TO_DATE('2011-08-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
LOB(BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE
TABLESPACE CM_F1T191_P2011JUL)
LOB(PRE_TRN_INIT_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE TABLESPACE CM_F1T191_P2011JUL)
LOB(PRE_TRN_FIN_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE TABLESPACE CM_F1T191_P2011JUL)
LOB(POST_TRN_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE
TABLESPACE CM_F1T191_P2011JUL)
TABLESPACE CM_F1T191_P2011JUL,

```

```

PARTITION "P2011AUG" VALUES LESS THAN (TO_DATE('2011-09-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
LOB(BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE
TABLESPACE CM_F1T191_P2011AUG)
LOB(PRE_TRN_INIT_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE TABLESPACE CM_F1T191_P2011AUG)
LOB(PRE_TRN_FIN_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE TABLESPACE CM_F1T191_P2011AUG)
LOB(POST_TRN_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE
TABLESPACE CM_F1T191_P2011AUG)
TABLESPACE CM_F1T191_P2011AUG,
PARTITION "P2011SEP" VALUES LESS THAN (TO_DATE('2011-10-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
LOB(BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE
TABLESPACE CM_F1T191_P2011SEP)
LOB(PRE_TRN_INIT_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE TABLESPACE CM_F1T191_P2011SEP)
LOB(PRE_TRN_FIN_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE TABLESPACE CM_F1T191_P2011SEP)
LOB(POST_TRN_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE
TABLESPACE CM_F1T191_P2011SEP)
TABLESPACE CM_F1T191_P2011SEP,
PARTITION "P2011OCT" VALUES LESS THAN (TO_DATE('2011-11-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
LOB(BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE
TABLESPACE CM_F1T191_P2011OCT)
LOB(PRE_TRN_INIT_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE TABLESPACE CM_F1T191_P2011OCT)
LOB(PRE_TRN_FIN_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE TABLESPACE CM_F1T191_P2011OCT)
LOB(POST_TRN_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE
TABLESPACE CM_F1T191_P2011OCT)
TABLESPACE CM_F1T191_P2011OCT,
PARTITION "P2011NOV" VALUES LESS THAN (TO_DATE('2011-12-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
LOB(BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE
TABLESPACE CM_F1T191_P2011NOV)
LOB(PRE_TRN_INIT_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE TABLESPACE CM_F1T191_P2011NOV)
LOB(PRE_TRN_FIN_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE TABLESPACE CM_F1T191_P2011NOV)
LOB(POST_TRN_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE
TABLESPACE CM_F1T191_P2011NOV)
TABLESPACE CM_F1T191_P2011NOV,
PARTITION "P2011DEC" VALUES LESS THAN (TO_DATE('2012-01-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
LOB(BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE
TABLESPACE CM_F1T191_P2011DEC)
LOB(PRE_TRN_INIT_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE TABLESPACE CM_F1T191_P2011DEC)
LOB(PRE_TRN_FIN_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE TABLESPACE CM_F1T191_P2011DEC)
LOB(POST_TRN_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE
TABLESPACE CM_F1T191_P2011DEC)
TABLESPACE CM_F1T191_P2011DEC,
PARTITION "PMAX" VALUES LESS THAN (MAXVALUE)
LOB(BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE
TABLESPACE CM_F1T191_PMAX)
LOB(PRE_TRN_INIT_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE TABLESPACE CM_F1T191_PMAX)
LOB(PRE_TRN_FIN_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE TABLESPACE CM_F1T191_PMAX)
LOB(POST_TRN_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE
TABLESPACE CM_F1T191_PMAX)
TABLESPACE CM_F1T191_PMAX
);

```

## INDEX

```

CREATE BIGFILE TABLESPACE CM_F1T191_IND DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON MAXSIZE
UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;

CREATE UNIQUE INDEX F1T191P0 ON F1_SYNC_REQ_IN(F1_SYNC_REQ_IN_ID) TABLESPACE CM_F1T191_IND
GLOBAL PARTITION BY RANGE (F1_SYNC_REQ_IN_ID)
(
PARTITION P1 VALUES LESS THAN ('124999999999999'),
PARTITION P2 VALUES LESS THAN ('249999999999999'),
PARTITION P3 VALUES LESS THAN ('374999999999999'),
PARTITION P4 VALUES LESS THAN ('499999999999999'),
PARTITION P5 VALUES LESS THAN ('624999999999999'),
PARTITION P6 VALUES LESS THAN ('749999999999999'),
PARTITION P7 VALUES LESS THAN ('874999999999999'),
PARTITION P8 VALUES LESS THAN (MAXVALUE)
);

```

```

);

ALTER TABLE F1_SYNC_REQ_IN ADD CONSTRAINT F1T191P0 PRIMARY KEY (F1_SYNC_REQ_IN_ID) USING
INDEX;

CREATE UNIQUE INDEX F1T191S1 ON F1_SYNC_REQ_IN (BO_STATUS_CD, BUS_OBJ_CD,
F1_SYNC_REQ_IN_ID) TABLESPACE CM_F1T191_IND COMPRESS ADVANCED LOW;

CREATE INDEX F1T191S2 ON F1_SYNC_REQ_IN (MAINT_OBJ_CD,EXT_PK_VALUE1,NT_XID_CD,PK_VALUE1)
TABLESPACE CM_F1T191_IND COMPRESS ADVANCED LOW;

CREATE INDEX F1T191S3 ON F1_SYNC_REQ_IN (EXT_REFERENCE_ID) TABLESPACE CM_F1T191_IND;
CREATE UNIQUE INDEX CM_ILM_F1T191S3 ON F1_SYNC_REQ_IN (ILM_DT, ILM_ARCH_SW,
F1_SYNC_REQ_IN_ID) LOCAL COMPRESS ADVANCED LOW;

```

## Child Table: F1\_SYNC\_REQ\_IN\_CHAR

```

CREATE TABLE F1_SYNC_REQ_IN_CHAR
(
 F1_SYNC_REQ_IN_ID CHAR(14) NOT NULL ENABLE,
 CHAR_TYPE_CD CHAR(8) NOT NULL ENABLE,
 SEQ_NUM NUMBER(3,0) NOT NULL ENABLE,
 CHAR_VAL CHAR(16) DEFAULT ' ' NOT NULL ENABLE,
 ADHOC_CHAR_VAL VARCHAR2(254) DEFAULT ' ' NOT NULL ENABLE,
 CHAR_VAL_FK1 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
 CHAR_VAL_FK2 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
 CHAR_VAL_FK3 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
 CHAR_VAL_FK4 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
 CHAR_VAL_FK5 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
 SRCH_CHAR_VAL VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
 VERSION NUMBER(5,0) DEFAULT 1 NOT NULL ENABLE,
 CONSTRAINT F1_SYNC_REQ_IN_CHAR_FK FOREIGN KEY (F1_SYNC_REQ_IN_ID) REFERENCES
F1_SYNC_REQ_IN ON DELETE CASCADE)
PARTITION BY REFERENCE (F1_SYNC_REQ_IN_CHAR_FK)
ENABLE ROW MOVEMENT;

```

### INDEX

```

CREATE UNIQUE INDEX F1T193P0 ON F1_SYNC_REQ_IN_CHAR (F1_SYNC_REQ_IN_ID, CHAR_TYPE_CD,
SEQ_NUM) TABLESPACE CM_F1T191_IND
GLOBAL PARTITION BY RANGE (F1_SYNC_REQ_IN_ID)
(
 PARTITION P1 VALUES LESS THAN ('124999999999999'),
 PARTITION P2 VALUES LESS THAN ('249999999999999'),
 PARTITION P3 VALUES LESS THAN ('374999999999999'),
 PARTITION P4 VALUES LESS THAN ('499999999999999'),
 PARTITION P5 VALUES LESS THAN ('624999999999999'),
 PARTITION P6 VALUES LESS THAN ('749999999999999'),
 PARTITION P7 VALUES LESS THAN ('874999999999999'),
 PARTITION P8 VALUES LESS THAN (MAXVALUE)
)
COMPRESS ADVANCED LOW;

ALTER TABLE F1_SYNC_REQ_IN_CHAR ADD CONSTRAINT F1T193P0 PRIMARY KEY (F1_SYNC_REQ_IN_ID,
CHAR_TYPE_CD, SEQ_NUM) USING INDEX;

CREATE INDEX F1T193S1 ON F1_SYNC_REQ_IN_CHAR (SRCH_CHAR_VAL) TABLESPACE CM_F1T191_IND ;

```

## Child Table: F1\_SYNC\_REQ\_IN\_EXCP

```

CREATE TABLE F1_SYNC_REQ_IN_EXCP
(
 F1_SYNC_REQ_IN_ID CHAR(14) NOT NULL ENABLE,
 SEQNO NUMBER(5,0) NOT NULL ENABLE,
 MESSAGE_CAT_NBR NUMBER(5,0) DEFAULT 0 NOT NULL ENABLE,
 MESSAGE_NBR NUMBER(5,0) DEFAULT 0 NOT NULL ENABLE,
 VERSION NUMBER(5,0) DEFAULT 1 NOT NULL ENABLE,
 CONSTRAINT F1_SYNC_REQ_IN_EXCP_FK FOREIGN KEY (F1_SYNC_REQ_IN_ID) REFERENCES
F1_SYNC_REQ_IN ON DELETE CASCADE)
PARTITION BY REFERENCE (F1_SYNC_REQ_IN_EXCP_FK)
ENABLE ROW MOVEMENT;

```

### INDEX

```

CREATE UNIQUE INDEX F1T197P0 ON F1_SYNC_REQ_IN_EXCP (F1_SYNC_REQ_IN_ID,SEQNO) TABLESPACE
CM_F1T191_IND

```

```

GLOBAL PARTITION BY RANGE (F1_SYNC_REQ_IN_ID)
(
PARTITION P1 VALUES LESS THAN ('12499999999999'),
PARTITION P2 VALUES LESS THAN ('24999999999999'),
PARTITION P3 VALUES LESS THAN ('37499999999999'),
PARTITION P4 VALUES LESS THAN ('49999999999999'),
PARTITION P5 VALUES LESS THAN ('62499999999999'),
PARTITION P6 VALUES LESS THAN ('74999999999999'),
PARTITION P7 VALUES LESS THAN ('87499999999999'),
PARTITION P8 VALUES LESS THAN (MAXVALUE)
)
COMPRESS ADVANCED LOW;

ALTER TABLE F1_SYNC_REQ_IN_EXCP ADD CONSTRAINT F1T197P0 PRIMARY KEY
(F1_SYNC_REQ_IN_ID,SEQNO) USING INDEX;

```

## Child Table: F1\_SYNC\_REQ\_IN\_EXCP\_PARM

```

CREATE TABLE F1_SYNC_REQ_IN_EXCP_PARM
(
F1_SYNC_REQ_IN_ID CHAR(14) NOT NULL ENABLE,
SEQNO NUMBER(5,0) NOT NULL ENABLE,
PARAM_SEQ NUMBER(3,0) NOT NULL ENABLE,
MSG_PARAM_VAL VARCHAR2(2000) DEFAULT ' ' NOT NULL ENABLE,
MSG_PARAM_TYP_FLG CHAR(4) DEFAULT ' ' NOT NULL ENABLE,
VERSION NUMBER(5,0) DEFAULT 1 NOT NULL ENABLE,
CONSTRAINT F1_SYNC_REQ_IN_EXCP_PARAM_FK FOREIGN KEY(F1_SYNC_REQ_IN_ID) REFERENCES
F1_SYNC_REQ_IN ON DELETE CASCADE)
PARTITION BY REFERENCE (F1_SYNC_REQ_IN_EXCP_PARAM_FK)
ENABLE ROW MOVEMENT;

```

## INDEX

```

CREATE UNIQUE INDEX F1T198P0 ON F1_SYNC_REQ_IN_EXCP_PARM(F1_SYNC_REQ_IN_ID,SEQNO,PARAM_SEQ)
TABLESPACE CM_F1T191_IND
GLOBAL PARTITION BY RANGE (F1_SYNC_REQ_IN_ID)
(
PARTITION P1 VALUES LESS THAN ('12499999999999'),
PARTITION P2 VALUES LESS THAN ('24999999999999'),
PARTITION P3 VALUES LESS THAN ('37499999999999'),
PARTITION P4 VALUES LESS THAN ('49999999999999'),
PARTITION P5 VALUES LESS THAN ('62499999999999'),
PARTITION P6 VALUES LESS THAN ('74999999999999'),
PARTITION P7 VALUES LESS THAN ('87499999999999'),
PARTITION P8 VALUES LESS THAN (MAXVALUE)
)
COMPRESS ADVANCED LOW;

ALTER TABLE F1_SYNC_REQ_IN_EXCP_PARM ADD CONSTRAINT F1T198P0 PRIMARY KEY
(F1_SYNC_REQ_IN_ID,SEQNO,PARAM_SEQ) USING INDEX;

```

## Child Table: F1\_SYNC\_REQ\_IN\_LOG

```

CREATE TABLE F1_SYNC_REQ_IN_LOG
(
F1_SYNC_REQ_IN_ID CHAR(14) NOT NULL ENABLE,
SEQNO NUMBER(5,0) NOT NULL ENABLE,
LOG_ENTRY_TYPE_FLG CHAR(4) DEFAULT ' ' NOT NULL ENABLE,
LOG_DTTM DATE NOT NULL ENABLE,
BO_STATUS_CD CHAR(12) DEFAULT ' ' NOT NULL ENABLE,
MESSAGE_CAT_NBR NUMBER(5,0) DEFAULT 0 NOT NULL ENABLE,
MESSAGE_NBR NUMBER(5,0) DEFAULT 0 NOT NULL ENABLE,
CHAR_TYPE_CD CHAR(8) DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL CHAR(16) DEFAULT ' ' NOT NULL ENABLE,
ADHOC_CHAR_VAL VARCHAR2(254) DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK1 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK2 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK3 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK4 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK5 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
DESCR_LONG VARCHAR2(4000) DEFAULT ' ' NOT NULL ENABLE,
USER_ID CHAR(8) DEFAULT ' ' NOT NULL ENABLE,
VERSION NUMBER(5,0) DEFAULT 1 NOT NULL ENABLE,
CONSTRAINT F1_SYNC_REQ_IN_LOG_FK FOREIGN KEY(F1_SYNC_REQ_IN_ID) REFERENCES
F1_SYNC_REQ_IN ON DELETE CASCADE)
PARTITION BY REFERENCE (F1_SYNC_REQ_IN_LOG_FK)
ENABLE ROW MOVEMENT;

```

**INDEX**

```

CREATE UNIQUE INDEX F1T194P0 ON F1_SYNC_REQ_IN_LOG(F1_SYNC_REQ_IN_ID,SEQNO) TABLESPACE
CM_F1T191_IND
GLOBAL PARTITION BY RANGE (F1_SYNC_REQ_IN_ID)
(
PARTITION P1 VALUES LESS THAN ('12499999999999'),
PARTITION P2 VALUES LESS THAN ('24999999999999'),
PARTITION P3 VALUES LESS THAN ('37499999999999'),
PARTITION P4 VALUES LESS THAN ('49999999999999'),
PARTITION P5 VALUES LESS THAN ('62499999999999'),
PARTITION P6 VALUES LESS THAN ('74999999999999'),
PARTITION P7 VALUES LESS THAN ('87499999999999'),
PARTITION P8 VALUES LESS THAN (MAXVALUE)
)
COMPRESS ADVANCED LOW;

ALTER TABLE F1_SYNC_REQ_IN_LOG ADD CONSTRAINT F1T194P0 PRIMARY KEY
(F1_SYNC_REQ_IN_ID,SEQNO) USING INDEX;

CREATE INDEX F1T194S1 ON F1_SYNC_REQ_IN_LOG(CHAR_TYPE_CD,CHAR_VAL_FK1) TABLESPACE
CM_F1T191_IND COMPRESS ADVANCED LOW;

CREATE INDEX F1T194S2 ON F1_SYNC_REQ_IN_LOG(CHAR_TYPE_CD,CHAR_VAL) TABLESPACE
CM_F1T191_IND COMPRESS ADVANCED LOW;

```

**Child Table: F1\_SYNC\_REQ\_IN\_LOG\_PARM**

```

CREATE TABLE F1_SYNC_REQ_IN_LOG_PARM
(
F1_SYNC_REQ_IN_ID CHAR(14) NOT NULL ENABLE,
SEQNO NUMBER(5,0) NOT NULL ENABLE,
PARM_SEQ NUMBER(3,0) NOT NULL ENABLE,
MSG_PARM_VAL VARCHAR2(2000) DEFAULT ' ' NOT NULL ENABLE,
MSG_PARM_TYP_FLG CHAR(4) DEFAULT ' ' NOT NULL ENABLE,
VERSION NUMBER(5,0) DEFAULT 1 NOT NULL ENABLE,
CONSTRAINT F1_SYNC_REQ_IN_LOG_PARM_FK FOREIGN KEY(F1_SYNC_REQ_IN_ID) REFERENCES
F1_SYNC_REQ_IN ON DELETE CASCADE)
PARTITION BY REFERENCE (F1_SYNC_REQ_IN_LOG_PARM_FK)
ENABLE ROW MOVEMENT;

```

**INDEX**

```

CREATE UNIQUE INDEX F1T195P0 ON F1_SYNC_REQ_IN_LOG_PARM(F1_SYNC_REQ_IN_ID,SEQNO,PARM_SEQ)
TABLESPACE CM_F1T191_IND
GLOBAL PARTITION BY RANGE (F1_SYNC_REQ_IN_ID)
(
PARTITION P1 VALUES LESS THAN ('12499999999999'),
PARTITION P2 VALUES LESS THAN ('24999999999999'),
PARTITION P3 VALUES LESS THAN ('37499999999999'),
PARTITION P4 VALUES LESS THAN ('49999999999999'),
PARTITION P5 VALUES LESS THAN ('62499999999999'),
PARTITION P6 VALUES LESS THAN ('74999999999999'),
PARTITION P7 VALUES LESS THAN ('87499999999999'),
PARTITION P8 VALUES LESS THAN (MAXVALUE)
)
COMPRESS ADVANCED LOW;

ALTER TABLE F1_SYNC_REQ_IN_LOG_PARM ADD CONSTRAINT F1T195P0 PRIMARY KEY
(F1_SYNC_REQ_IN_ID,SEQNO,PARM_SEQ) USING INDEX;

```

**Child Table: F1\_SYNC\_REQ\_IN\_REL\_OBJ**

```

CREATE TABLE F1_SYNC_REQ_IN_REL_OBJ
(
F1_SYNC_REQ_IN_ID CHAR(14) NOT NULL ENABLE,
MAINT_OBJ_CD CHAR(12) DEFAULT ' ' NOT NULL ENABLE,
REL_OBJ_TYPE_FLG CHAR(4) DEFAULT ' ' NOT NULL ENABLE,
PK_VALUE1 VARCHAR2(254) DEFAULT ' ' NOT NULL ENABLE,
PK_VALUE2 VARCHAR2(254) DEFAULT ' ' NOT NULL ENABLE,
PK_VALUE3 VARCHAR2(254) DEFAULT ' ' NOT NULL ENABLE,
PK_VALUE4 VARCHAR2(254) DEFAULT ' ' NOT NULL ENABLE,
PK_VALUE5 VARCHAR2(254) DEFAULT ' ' NOT NULL ENABLE,
VERSION NUMBER(5,0) DEFAULT 1 NOT NULL ENABLE,
CONSTRAINT F1_SYNC_REQ_IN_REL_OBJ_FK FOREIGN KEY(F1_SYNC_REQ_IN_ID) REFERENCES
F1_SYNC_REQ_IN ON DELETE CASCADE)
PARTITION BY REFERENCE (F1_SYNC_REQ_IN_REL_OBJ_FK)
ENABLE ROW MOVEMENT;

```

**INDEX**

```

CREATE UNIQUE INDEX F1T192P0 ON F1_SYNC_REQ_IN_REL_OBJ(F1_SYNC_REQ_IN_ID, MAINT_OBJ_CD,
REL_OBJ_TYPE_FLG) TABLESPACE CM_F1T191_IND
GLOBAL PARTITION BY RANGE (F1_SYNC_REQ_IN_ID)
(
PARTITION P1 VALUES LESS THAN ('12499999999999'),
PARTITION P2 VALUES LESS THAN ('24999999999999'),
PARTITION P3 VALUES LESS THAN ('37499999999999'),
PARTITION P4 VALUES LESS THAN ('49999999999999'),
PARTITION P5 VALUES LESS THAN ('62499999999999'),
PARTITION P6 VALUES LESS THAN ('74999999999999'),
PARTITION P7 VALUES LESS THAN ('87499999999999'),
PARTITION P8 VALUES LESS THAN (MAXVALUE)
)
COMPRESS ADVANCED LOW;

ALTER TABLE F1_SYNC_REQ_IN_REL_OBJ ADD CONSTRAINT F1T192P0 PRIMARY KEY (F1_SYNC_REQ_IN_ID,
MAINT_OBJ_CD, REL_OBJ_TYPE_FLG) USING INDEX;

CREATE INDEX F1T192S1 ON F1_SYNC_REQ_IN_REL_OBJ(PK_VALUE1) TABLESPACE CM_F1T191_IND;

```

**Maintenance Object: D1-IMD****Parent Table: D1\_INIT\_MSRMT\_DATA**

```

CREATE BIGFILE TABLESPACE CM_D1T304_P2011JAN_S181 DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011JAN_SMAX DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011FEB_S181 DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011FEB_SMAX DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011MAR_S181 DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011MAR_SMAX DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011APR_S181 DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011APR_SMAX DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011MAY_S181 DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011MAY_SMAX DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011JUN_S181 DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011JUN_SMAX DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011JUL_S181 DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011JUL_SMAX DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011AUG_S181 DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011AUG_SMAX DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011SEP_S181 DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011SEP_SMAX DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011OCT_S181 DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011OCT_SMAX DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011NOV_S181 DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011NOV_SMAX DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011DEC_S181 DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;

```

```

CREATE BIGFILE TABLESPACE CM_D1T304_P2011DEC_SMAX DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_PMAX_S181 DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_PMAX_SMAX DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;

CREATE TABLE D1_INIT_MSRMT_DATA
(
 INIT_MSRMT_DATA_ID CHAR(14) NOT NULL ENABLE,
 MEASR_COMP_ID CHAR(12) DEFAULT ' ' NOT NULL ENABLE,
 D1_FROM_DTTM DATE,
 D1_TO_DTTM DATE,
 DATA_SRC_FLG CHAR(4) DEFAULT ' ' NOT NULL ENABLE,
 TIME_ZONE_CD CHAR(10) DEFAULT ' ' NOT NULL ENABLE,
 BUS_OBJ_CD CHAR(30) DEFAULT ' ' NOT NULL ENABLE,
 BO_STATUS_CD CHAR(12) DEFAULT ' ' NOT NULL ENABLE,
 BO_STATUS_REASON_CD VARCHAR2(30) DEFAULT ' ' NOT NULL ENABLE,
 IMD_BO_DATA_AREA CLOB,
 STATUS_UPD_DTTM DATE NOT NULL ENABLE,
 CRE_DTTM DATE NOT NULL ENABLE,
 VERSION NUMBER(5,0) DEFAULT 1 NOT NULL ENABLE,
 IMD_EXT_ID VARCHAR2(120),
 PREVEE_BO_DATA_AREA CLOB,
 POSTVEE_BO_DATA_AREA CLOB,
 TRACE_BO_DATA_AREA CLOB,
 RAW_BO_DATA_AREA CLOB,
 LAST_UPDATE_DTTM DATE,
 ILM_DT DATE,
 ILM_ARCH_SW CHAR(1),
 RETENTION_PERIOD NUMBER(5,0) DEFAULT 99999 NOT NULL ENABLE
)
ENABLE ROW MOVEMENT PCTFREE 50
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE)
PARTITION BY RANGE (ILM_DT)
SUBPARTITION BY range (RETENTION_PERIOD)
(
 PARTITION "P2011JAN" VALUES LESS THAN (TO_DATE('2011-02-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN')) (
 SUBPARTITION P2011JAN_S181 VALUES LESS THAN (181) TABLESPACE CM_D1T304_P2011JAN_S181
 LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JAN_S181)
 LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JAN_S181)
 LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JAN_S181)
 LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JAN_S181)
 LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JAN_S181)
 ,
 SUBPARTITION P2011JAN_SMAX VALUES LESS THAN (MAXVALUE) TABLESPACE CM_D1T304_P2011JAN_SMAX
 LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JAN_SMAX)
 LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JAN_SMAX)
 LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JAN_SMAX)
 LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JAN_SMAX)
 LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JAN_SMAX)
),
 PARTITION "P2011FEB" VALUES LESS THAN (TO_DATE('2011-03-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN')) (
 SUBPARTITION P2011FEB_S181 VALUES LESS THAN (181) TABLESPACE CM_D1T304_P2011FEB_S181
 LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011FEB_S181)
 LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011FEB_S181)
 LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011FEB_S181)
 LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011FEB_S181)
 LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011FEB_S181)
 ,
 SUBPARTITION P2011FEB_SMAX VALUES LESS THAN (MAXVALUE) TABLESPACE CM_D1T304_P2011FEB_SMAX
 LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011FEB_SMAX)
 LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011FEB_SMAX)
 LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011FEB_SMAX)
 LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011FEB_SMAX)
 LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011FEB_SMAX)
),
 PARTITION "P2011MAR" VALUES LESS THAN (TO_DATE('2011-04-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN')) (
 SUBPARTITION P2011MAR_S181 VALUES LESS THAN (181) TABLESPACE CM_D1T304_P2011MAR_S181
 LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAR_S181)
 LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAR_S181)
 LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAR_S181)
 LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAR_S181)
 LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAR_S181)
 ,
 SUBPARTITION P2011MAR_SMAX VALUES LESS THAN (MAXVALUE) TABLESPACE CM_D1T304_P2011MAR_SMAX
 LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAR_SMAX)

```

```

LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAR_SMAX)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAR_SMAX)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAR_SMAX)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAR_SMAX)
),
PARTITION "P2011APR" VALUES LESS THAN (TO_DATE('2011-05-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))(
SUBPARTITION P2011APR_S181 VALUES LESS THAN (181) TABLESPACE CM_D1T304_P2011APR_S181
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011APR_S181)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011APR_S181)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011APR_S181)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011APR_S181)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011APR_S181)
),
SUBPARTITION P2011APR_SMAX VALUES LESS THAN (MAXVALUE) TABLESPACE CM_D1T304_P2011APR_SMAX
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011APR_SMAX)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011APR_SMAX)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011APR_SMAX)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011APR_SMAX)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011APR_SMAX)
),
PARTITION "P2011MAY" VALUES LESS THAN (TO_DATE('2011-06-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))(
SUBPARTITION P2011MAY_S181 VALUES LESS THAN (181) TABLESPACE CM_D1T304_P2011MAY_S181
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAY_S181)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAY_S181)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAY_S181)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAY_S181)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAY_S181)
),
SUBPARTITION P2011MAY_SMAX VALUES LESS THAN (MAXVALUE) TABLESPACE CM_D1T304_P2011MAY_SMAX
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAY_SMAX)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAY_SMAX)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAY_SMAX)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAY_SMAX)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAY_SMAX)
),
PARTITION "P2011JUN" VALUES LESS THAN (TO_DATE('2011-07-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))(
SUBPARTITION P2011JUN_S181 VALUES LESS THAN (181) TABLESPACE CM_D1T304_P2011JUN_S181
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUN_S181)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUN_S181)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUN_S181)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUN_S181)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUN_S181)
),
SUBPARTITION P2011JUN_SMAX VALUES LESS THAN (MAXVALUE) TABLESPACE CM_D1T304_P2011JUN_SMAX
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUN_SMAX)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUN_SMAX)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUN_SMAX)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUN_SMAX)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUN_SMAX)
),
PARTITION "P2011JUL" VALUES LESS THAN (TO_DATE('2011-08-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))(
SUBPARTITION P2011JUL_S181 VALUES LESS THAN (181) TABLESPACE CM_D1T304_P2011JUL_S181
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUL_S181)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUL_S181)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUL_S181)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUL_S181)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUL_S181)
),
SUBPARTITION P2011JUL_SMAX VALUES LESS THAN (MAXVALUE) TABLESPACE CM_D1T304_P2011JUL_SMAX
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUL_SMAX)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUL_SMAX)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUL_SMAX)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUL_SMAX)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUL_SMAX)
),
PARTITION "P2011AUG" VALUES LESS THAN (TO_DATE('2011-09-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))(
SUBPARTITION P2011AUG_S181 VALUES LESS THAN (181) TABLESPACE CM_D1T304_P2011AUG_S181
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011AUG_S181)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011AUG_S181)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011AUG_S181)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011AUG_S181)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011AUG_S181)
),
SUBPARTITION P2011AUG_SMAX VALUES LESS THAN (MAXVALUE) TABLESPACE CM_D1T304_P2011AUG_SMAX
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011AUG_SMAX)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011AUG_SMAX)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011AUG_SMAX)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011AUG_SMAX)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011AUG_SMAX)
),

```



```

PARTITION "P2011SEP" VALUES LESS THAN (TO_DATE('2011-10-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS CALENDAR=GREGORIAN')) (
SUBPARTITION P2011SEP_S181 VALUES LESS THAN (181) TABLESPACE CM_D1T304_P2011SEP_S181
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011SEP_S181)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011SEP_S181)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011SEP_S181)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011SEP_S181)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011SEP_S181)
),
SUBPARTITION P2011SEP_SMAX VALUES LESS THAN (MAXVALUE) TABLESPACE CM_D1T304_P2011SEP_SMAX
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011SEP_SMAX)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011SEP_SMAX)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011SEP_SMAX)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011SEP_SMAX)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011SEP_SMAX)
),
PARTITION "P2011OCT" VALUES LESS THAN (TO_DATE('2011-11-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS CALENDAR=GREGORIAN')) (
SUBPARTITION P2011OCT_S181 VALUES LESS THAN (181) TABLESPACE CM_D1T304_P2011OCT_S181
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011OCT_S181)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011OCT_S181)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011OCT_S181)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011OCT_S181)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011OCT_S181)
),
SUBPARTITION P2011OCT_SMAX VALUES LESS THAN (MAXVALUE) TABLESPACE CM_D1T304_P2011OCT_SMAX
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011OCT_SMAX)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011OCT_SMAX)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011OCT_SMAX)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011OCT_SMAX)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011OCT_SMAX)
),
PARTITION "P2011NOV" VALUES LESS THAN (TO_DATE('2011-12-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS CALENDAR=GREGORIAN')) (
SUBPARTITION P2011NOV_S181 VALUES LESS THAN (181) TABLESPACE CM_D1T304_P2011NOV_S181
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011NOV_S181)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011NOV_S181)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011NOV_S181)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011NOV_S181)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011NOV_S181)
),
SUBPARTITION P2011NOV_SMAX VALUES LESS THAN (MAXVALUE) TABLESPACE CM_D1T304_P2011NOV_SMAX
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011NOV_SMAX)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011NOV_SMAX)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011NOV_SMAX)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011NOV_SMAX)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011NOV_SMAX)
),
PARTITION "P2011DEC" VALUES LESS THAN (TO_DATE('2012-01-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS CALENDAR=GREGORIAN')) (
SUBPARTITION P2011DEC_S181 VALUES LESS THAN (181) TABLESPACE CM_D1T304_P2011DEC_S181
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011DEC_S181)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011DEC_S181)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011DEC_S181)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011DEC_S181)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011DEC_S181)
),
SUBPARTITION P2011DEC_SMAX VALUES LESS THAN (MAXVALUE) TABLESPACE CM_D1T304_P2011DEC_SMAX
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011DEC_SMAX)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011DEC_SMAX)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011DEC_SMAX)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011DEC_SMAX)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011DEC_SMAX)
),
PARTITION "P2011PMAX" VALUES LESS THAN (MAXVALUE) (
SUBPARTITION P2011PMAX_S181 VALUES LESS THAN (181) TABLESPACE CM_D1T304_P2011PMAX_S181
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011PMAX_S181)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011PMAX_S181)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011PMAX_S181)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011PMAX_S181)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011PMAX_S181)
),
SUBPARTITION P2011PMAX_SMAX VALUES LESS THAN (MAXVALUE) TABLESPACE CM_D1T304_P2011PMAX_SMAX
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011PMAX_SMAX)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011PMAX_SMAX)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011PMAX_SMAX)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011PMAX_SMAX)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011PMAX_SMAX)
)
);

```

**INDEX**

```

CREATE BIGFILE TABLESPACE CM_D1T304_IND DATAFILE '+DATA' SIZE 50M AUTOEXTEND ON MAXSIZE
UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;

CREATE UNIQUE INDEX D1T304P0 ON D1_INIT_MSRMT_DATA(INIT_MSRMT_DATA_ID) TABLESPACE
CM_D1T304_IND
GLOBAL PARTITION BY RANGE (INIT_MSRMT_DATA_ID)
(
 PARTITION P1 VALUES LESS THAN ('12499999999999'),
 PARTITION P2 VALUES LESS THAN ('24999999999999'),
 PARTITION P3 VALUES LESS THAN ('37499999999999'),
 PARTITION P4 VALUES LESS THAN ('49999999999999'),
 PARTITION P5 VALUES LESS THAN ('62499999999999'),
 PARTITION P6 VALUES LESS THAN ('74999999999999'),
 PARTITION P7 VALUES LESS THAN ('87499999999999'),
 PARTITION P8 VALUES LESS THAN (MAXVALUE)
);

ALTER TABLE D1_INIT_MSRMT_DATA ADD CONSTRAINT D1T304P0 PRIMARY KEY(INIT_MSRMT_DATA_ID)
USING INDEX;

CREATE INDEX D1T304S1 ON D1_INIT_MSRMT_DATA (MEASR_COMP_ID,D1_TO_DTTM) TABLESPACE
CM_D1T304_IND
GLOBAL PARTITION BY RANGE (MEASR_COMP_ID)
(
 PARTITION P1 VALUES LESS THAN ('12499999999999'),
 PARTITION P2 VALUES LESS THAN ('24999999999999'),
 PARTITION P3 VALUES LESS THAN ('37499999999999'),
 PARTITION P4 VALUES LESS THAN ('49999999999999'),
 PARTITION P5 VALUES LESS THAN ('62499999999999'),
 PARTITION P6 VALUES LESS THAN ('74999999999999'),
 PARTITION P7 VALUES LESS THAN ('87499999999999'),
 PARTITION P8 VALUES LESS THAN (MAXVALUE)
)
COMPRESS ADVANCED LOW;

CREATE UNIQUE INDEX CM_ILM_D1T304S4 ON D1_INIT_MSRMT_DATA (ILM_DT, RETENTION_PERIOD,
ILM_ARCH_SW, INIT_MSRMT_DATA_ID) LOCAL COMPRESS ADVANCED LOW;

```

**Child Table: D1\_INIT\_MSRMT\_DATA\_CHAR**

```

CREATE TABLE D1_INIT_MSRMT_DATA_CHAR
(
 INIT_MSRMT_DATA_ID CHAR(14) NOT NULL ENABLE,
 CHAR_TYPE_CD CHAR(8) NOT NULL ENABLE,
 SEQ_NUM NUMBER(3,0) NOT NULL ENABLE,
 CHAR_VAL CHAR(16) DEFAULT ' ' NOT NULL ENABLE,
 ADHOC_CHAR_VAL VARCHAR2(254) DEFAULT ' ' NOT NULL ENABLE,
 CHAR_VAL_FK1 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
 CHAR_VAL_FK2 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
 CHAR_VAL_FK3 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
 CHAR_VAL_FK4 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
 CHAR_VAL_FK5 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
 SRCH_CHAR_VAL VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
 VERSION NUMBER(5,0) DEFAULT 1 NOT NULL ENABLE,
 LAST_UPDATE_DTTM DATE,
 CONSTRAINT D1_INIT_MSRMT_DATA_CHAR_FK FOREIGN KEY(INIT_MSRMT_DATA_ID) REFERENCES
D1_INIT_MSRMT_DATA ON DELETE CASCADE)
PARTITION BY REFERENCE (D1_INIT_MSRMT_DATA_CHAR_FK)
ENABLE ROW MOVEMENT;

```

**INDEX**

```

CREATE UNIQUE INDEX D1T305P0 ON D1_INIT_MSRMT_DATA_CHAR(INIT_MSRMT_DATA_ID, CHAR_TYPE_CD,
SEQ_NUM) TABLESPACE CM_D1T304_IND
GLOBAL PARTITION BY RANGE(INIT_MSRMT_DATA_ID)
(
 PARTITION P1 VALUES LESS THAN ('12499999999999'),
 PARTITION P2 VALUES LESS THAN ('24999999999999'),
 PARTITION P3 VALUES LESS THAN ('37499999999999'),
 PARTITION P4 VALUES LESS THAN ('49999999999999'),
 PARTITION P5 VALUES LESS THAN ('62499999999999'),
 PARTITION P6 VALUES LESS THAN ('74999999999999'),
 PARTITION P7 VALUES LESS THAN ('87499999999999'),
 PARTITION P8 VALUES LESS THAN (MAXVALUE)
) COMPRESS ADVANCED LOW;

```

```

ALTER TABLE D1_INIT_MSRMT_DATA_CHAR ADD CONSTRAINT D1T305P0 PRIMARY KEY
(INIT_MSRMT_DATA_ID, CHAR_TYPE_CD, SEQ_NUM) USING INDEX;

CREATE INDEX D1T305S1 ON D1_INIT_MSRMT_DATA_CHAR(SRCH_CHAR_VAL)
GLOBAL PARTITION BY HASH(SRCH_CHAR_VAL)
(
PARTITION P1 TABLESPACE CM_D1T304_IND,
PARTITION P2 TABLESPACE CM_D1T304_IND,
PARTITION P3 TABLESPACE CM_D1T304_IND,
PARTITION P4 TABLESPACE CM_D1T304_IND,
PARTITION P5 TABLESPACE CM_D1T304_IND,
PARTITION P6 TABLESPACE CM_D1T304_IND,
PARTITION P7 TABLESPACE CM_D1T304_IND,
PARTITION P8 TABLESPACE CM_D1T304_IND
);

```

## Child Table: D1\_INIT\_MSRMT\_DATA\_LOG

```

CREATE TABLE D1_INIT_MSRMT_DATA_LOG
(
INIT_MSRMT_DATA_ID CHAR(14) NOT NULL ENABLE,
SEQNO NUMBER(5,0) NOT NULL ENABLE,
BO_STATUS_CD CHAR(12) DEFAULT ' ' NOT NULL ENABLE,
BO_STATUS_REASON_CD VARCHAR2(30) DEFAULT ' ' NOT NULL ENABLE,
CHAR_TYPE_CD CHAR(8) DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL CHAR(16) DEFAULT ' ' NOT NULL ENABLE,
ADHOC_CHAR_VAL VARCHAR2(254) DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK1 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK2 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK3 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK4 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK5 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
DESCRLONG VARCHAR2(4000) DEFAULT ' ' NOT NULL ENABLE,
LOG_DTTM DATE NOT NULL ENABLE,
LOG_ENTRY_TYPE_FLG CHAR(4) DEFAULT ' ' NOT NULL ENABLE,
MESSAGE_CAT_NBR NUMBER(5,0) DEFAULT 0 NOT NULL ENABLE,
MESSAGE_NBR NUMBER(5,0) DEFAULT 0 NOT NULL ENABLE,
USER_ID CHAR(8) DEFAULT ' ' NOT NULL ENABLE,
VERSION NUMBER(5,0) DEFAULT 1 NOT NULL ENABLE,
LAST_UPDATE_DTTM DATE,
CONSTRAINT D1_INIT_MSRMT_DATA_LOG_FK FOREIGN KEY(INIT_MSRMT_DATA_ID) REFERENCES
D1_INIT_MSRMT_DATA ON DELETE CASCADE)
PARTITION BY REFERENCE (D1_INIT_MSRMT_DATA_LOG_FK)
ENABLE ROW MOVEMENT;

```

## INDEX

```

CREATE UNIQUE INDEX D1T306P0 ON D1_INIT_MSRMT_DATA_LOG (INIT_MSRMT_DATA_ID, SEQNO)
TABLESPACE CM_D1T304_IND
GLOBAL PARTITION BY RANGE(INIT_MSRMT_DATA_ID)
(
PARTITION P1 VALUES LESS THAN ('12499999999999'),
PARTITION P2 VALUES LESS THAN ('24999999999999'),
PARTITION P3 VALUES LESS THAN ('37499999999999'),
PARTITION P4 VALUES LESS THAN ('49999999999999'),
PARTITION P5 VALUES LESS THAN ('62499999999999'),
PARTITION P6 VALUES LESS THAN ('74999999999999'),
PARTITION P7 VALUES LESS THAN ('87499999999999'),
PARTITION P8 VALUES LESS THAN (MAXVALUE)
) COMPRESS ADVANCED LOW;

ALTER TABLE D1_INIT_MSRMT_DATA_LOG ADD CONSTRAINT D1T306P0 PRIMARY KEY
(INIT_MSRMT_DATA_ID, SEQNO) USING INDEX;

```

## Child Table: D1\_INIT\_MSRMT\_DATA\_LOG\_PARM

```

CREATE TABLE D1_INIT_MSRMT_DATA_LOG_PARM
(
INIT_MSRMT_DATA_ID CHAR(14) NOT NULL ENABLE,
SEQNO NUMBER(5,0) NOT NULL ENABLE,
PARM_SEQ NUMBER(3,0) NOT NULL ENABLE,

```

```

MSG_PARM_VAL VARCHAR2(2000) DEFAULT ' ' NOT NULL ENABLE,
MSG_PARM_TYP_FLG CHAR(4) DEFAULT ' ' NOT NULL ENABLE,
VERSION NUMBER(5,0) DEFAULT 1 NOT NULL ENABLE,
LAST_UPDATE_DTTM DATE,
CONSTRAINT D1_INIT_MSRMT_DATA_LOG_PARM_FK FOREIGN KEY(INIT_MSRMT_DATA_ID) REFERENCES
D1_INIT_MSRMT_DATA ON DELETE CASCADE)
PARTITION BY REFERENCE (D1_INIT_MSRMT_DATA_LOG_PARM_FK)
ENABLE ROW MOVEMENT;

```

## INDEX

```

CREATE UNIQUE INDEX D1T307P0 ON D1_INIT_MSRMT_DATA_LOG_PARM(INIT_MSRMT_DATA_ID, SEQNO,
PARM_SEQ) TABLESPACE CM_D1T304_IND
GLOBAL PARTITION BY RANGE(INIT_MSRMT_DATA_ID)
(
PARTITION P1 VALUES LESS THAN ('12499999999999'),
PARTITION P2 VALUES LESS THAN ('24999999999999'),
PARTITION P3 VALUES LESS THAN ('37499999999999'),
PARTITION P4 VALUES LESS THAN ('49999999999999'),
PARTITION P5 VALUES LESS THAN ('62499999999999'),
PARTITION P6 VALUES LESS THAN ('74999999999999'),
PARTITION P7 VALUES LESS THAN ('87499999999999'),
PARTITION P8 VALUES LESS THAN (MAXVALUE)
) COMPRESS ADVANCED LOW;

ALTER TABLE D1_INIT_MSRMT_DATA_LOG_PARM ADD CONSTRAINT D1T307P0 PRIMARY KEY
(INIT_MSRMT_DATA_ID, SEQNO, PARM_SEQ) USING INDEX;

```

## Child Table: D1\_INIT\_MSRMT\_DATA\_K

```

CREATE BIGFILE TABLESPACE CM_D1T314_IND DATAFILE '+DATA' SIZE 50M AUTOEXTEND ON MAXSIZE
UNLIMITED;

CREATE TABLE D1_INIT_MSRMT_DATA_K
(
INIT_MSRMT_DATA_ID CHAR(14) NOT NULL ENABLE,
ENV_ID NUMBER(6,0) NOT NULL ENABLE,
CONSTRAINT D1T314P0 PRIMARY KEY (INIT_MSRMT_DATA_ID, ENV_ID) ENABLE
)
ORGANIZATION INDEX
Partition by range(INIT_MSRMT_DATA_ID)
(
PARTITION P1 VALUES LESS THAN ('12499999999999'),
PARTITION P2 VALUES LESS THAN ('24999999999999'),
PARTITION P3 VALUES LESS THAN ('37499999999999'),
PARTITION P4 VALUES LESS THAN ('49999999999999'),
PARTITION P5 VALUES LESS THAN ('62499999999999'),
PARTITION P6 VALUES LESS THAN ('74999999999999'),
PARTITION P7 VALUES LESS THAN ('87499999999999'),
PARTITION P8 VALUES LESS THAN (MAXVALUE)
)
TABLESPACE CM_D1T314_IND;

```

# Appendix B

## Sample SQL for Enabling ILM in SGG (Existing Installation)

This section provides additional details related to supporting ILM in an existing installation. It includes the sample syntax for each step using the To Do Entry maintenance object as an example. Other maintenance object's implementations can follow a similar pattern.

1. Rename existing table CI\_TD\_ENTRY and primary key index as a backup. It is suggested to use an ILM\_ prefix. The following are sample statements:

```
ALTER TABLE CI_TD_ENTRY RENAME TO ILM_TD_ENTRY;
ALTER INDEX XT039P0 RENAME TO ILM_XT039P0;
```

2. Generate DDL for the secondary index.

```
set heading off;
set echo off;
Set pages 999;
set long 90000;

spool ddl_list.sql
select dbms_metadata.get_ddl('INDEX','XT039S2','CISADM') from dual;
select dbms_metadata.get_ddl('INDEX','XT039S3','CISADM') from dual;
select dbms_metadata.get_ddl('INDEX','XT039S4','CISADM') from dual;
select dbms_metadata.get_ddl('INDEX','XT039S5','CISADM') from dual;
select dbms_metadata.get_ddl('INDEX','XT039S6','CISADM') from dual;
select dbms_metadata.get_ddl('INDEX','XT039S7','CISADM') from dual;
select dbms_metadata.get_ddl('INDEX','XT039S8','CISADM') from dual;
select dbms_metadata.get_ddl('INDEX','CM_ILM_XT039S8','CISADM') from dual;
spool off;
```

3. Drop secondary indexes.

```
DROP INDEX CISADM.XT039S2;
DROP INDEX CISADM.XT039S3;
DROP INDEX CISADM.XT039S4;
DROP INDEX CISADM.XT039S5;
DROP INDEX CISADM.XT039S6;
DROP INDEX CISADM.XT039S7;
DROP INDEX CISADM.XT039S8;
DROP INDEX CISADM.CM_ILM_XT039S8;
```

4. Create a partitioned table.

In the following example ILM\_DT value is inserted from column CRE\_DTTM. The degree setting of 'parallel' in the DDL can be adjusted according to the table's data, its means and its size.

```
CREATE TABLE CI_TD_ENTRY (
 TD_ENTRY_ID CHAR(14) NOT NULL ENABLE,
 BATCH_CD CHAR(8) DEFAULT ' ' NOT NULL ENABLE,
 BATCH_NBR NUMBER(10,0) DEFAULT 0 NOT NULL ENABLE,
 MESSAGE_CAT_NBR NUMBER(5,0) DEFAULT 0 NOT NULL ENABLE,
 MESSAGE_NBR NUMBER(5,0) DEFAULT 0 NOT NULL ENABLE,
```

```

ASSIGNED_TO CHAR(8) DEFAULT ' ' NOT NULL ENABLE,
TD_TYPE_CD CHAR(8) DEFAULT ' ' NOT NULL ENABLE,
ROLE_ID CHAR(10) DEFAULT ' ' NOT NULL ENABLE,
ENTRY_STATUS_FLG CHAR(2) DEFAULT ' ' NOT NULL ENABLE,
VERSION NUMBER(5,0) DEFAULT 1 NOT NULL ENABLE,
CRE_DTTM DATE,
ASSIGNED_DTTM DATE,
COMPLETE_DTTM DATE,
COMPLETE_USER_ID CHAR(8) DEFAULT ' ' NOT NULL ENABLE,
COMMENTS VARCHAR2(254) DEFAULT ' ' NOT NULL ENABLE,
ASSIGNED_USER_ID CHAR(8) DEFAULT ' ' NOT NULL ENABLE,
TD_PRIORITY_FLG CHAR(4) DEFAULT ' ' NOT NULL ENABLE,
ILM_DT DATE,
ILM_ARCH_SW CHAR(1)
) NOLOGGING PARALLEL
ENABLE ROW MOVEMENT
PARTITION BY RANGE (ILM_DT)
SUBPARTITION BY RANGE (TD_ENTRY_ID) SUBPARTITION TEMPLATE
(
SUBPARTITION S01 VALUES LESS THAN ('1249999999999999'),
SUBPARTITION S02 VALUES LESS THAN ('2499999999999999'),
SUBPARTITION S03 VALUES LESS THAN ('3749999999999999'),
SUBPARTITION S04 VALUES LESS THAN ('4999999999999999'),
SUBPARTITION S05 VALUES LESS THAN ('6249999999999999'),
SUBPARTITION S06 VALUES LESS THAN ('7499999999999999'),
SUBPARTITION S07 VALUES LESS THAN ('8749999999999999'),
SUBPARTITION SMAX VALUES LESS THAN (MAXVALUE)
)
(
PARTITION "P2011JAN" VALUES LESS THAN (TO_DATE('2011-02-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLESPACE CM_XT039_P2011JAN,
PARTITION "P2011FEB" VALUES LESS THAN (TO_DATE('2011-03-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLESPACE CM_XT039_P2011FEB,
PARTITION "P2011MAR" VALUES LESS THAN (TO_DATE('2011-04-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLESPACE CM_XT039_P2011MAR,
PARTITION "P2011APR" VALUES LESS THAN (TO_DATE('2011-05-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLESPACE CM_XT039_P2011APR,
PARTITION "P2011MAY" VALUES LESS THAN (TO_DATE('2011-06-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLESPACE CM_XT039_P2011MAY,
PARTITION "P2011JUN" VALUES LESS THAN (TO_DATE('2011-07-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLESPACE CM_XT039_P2011JUN,
PARTITION "P2011JUL" VALUES LESS THAN (TO_DATE('2011-08-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLESPACE CM_XT039_P2011JUL,
PARTITION "P2011AUG" VALUES LESS THAN (TO_DATE('2011-09-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLESPACE CM_XT039_P2011AUG,
PARTITION "P2011SEP" VALUES LESS THAN (TO_DATE('2011-10-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLESPACE CM_XT039_P2011SEP,
PARTITION "P2011OCT" VALUES LESS THAN (TO_DATE('2011-11-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLESPACE CM_XT039_P2011OCT,
PARTITION "P2011NOV" VALUES LESS THAN (TO_DATE('2011-12-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLESPACE CM_XT039_P2011NOV,
PARTITION "P2011NOV" VALUES LESS THAN (MAXVALUE)
TABLESPACE CM_XT039_PMAX
)as select /* PARALLEL */
TD_ENTRY_ID,
BATCH_CD,
BATCH_NBR,
MESSAGE_CAT_NBR,
MESSAGE_NBR,
ASSIGNED_TO,
TD_TYPE_CD,
ROLE_ID,
ENTRY_STATUS_FLG,
VERSION,
CRE_DTTM,
ASSIGNED_DTTM,
COMPLETE_DTTM,
COMPLETE_USER_ID,
COMMENTS,
ASSIGNED_USER_ID,
TD_PRIORITY_FLG,
CRE_DTTM as ILM_DT,
ILM_ARCH_SW
from ILM_TD_ENTRY

```

- /
5. Enable logging option for table CI\_TD\_ENTRY.
 

```
ALTER TABLE CI_TD_ENTRY NOPARALLEL LOGGING;
```
  6. Create primary index for parent table CI\_TD\_ENTRY.
 

```
CREATE BIGFILE TABLESPACE CM_XT039_IND DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;

CREATE UNIQUE INDEX XT039P0 ON CI_TD_ENTRY NOLOGGING PARALLEL (
TD_ENTRY_ID
)
PARTITION P1 VALUES LESS THAN ('124999999999999'),
PARTITION P2 VALUES LESS THAN ('249999999999999'),
PARTITION P3 VALUES LESS THAN ('374999999999999'),
PARTITION P4 VALUES LESS THAN ('499999999999999'),
PARTITION P5 VALUES LESS THAN ('624999999999999'),
PARTITION P6 VALUES LESS THAN ('749999999999999'),
PARTITION P7 VALUES LESS THAN ('874999999999999'),
PARTITION P8 VALUES LESS THAN (MAXVALUE)
) TABLESPACE CM_XT039_IND
/

ALTER INDEX XT039P0 LOGGING NOPARALLEL;
```
  7. Add Primary Key for Parent table CI\_TD\_ENTRY
 

```
ALTER TABLE CI_TD_ENTRY ADD CONSTRAINT XT039P0 PRIMARY KEY(TD_ENTRY_ID) USING INDEX
/
```
  8. Create Secondary Indexes for Parent table CI\_TD\_ENTRY.
 

```
CREATE UNIQUE INDEX CM_ILM_XT039S8 ON CI_TD_ENTRY (ILM_DT, ILM_ARCH_SW, TD_ENTRY_ID)
LOCAL COMPRESS ADVANCED LOW
/

CREATE UNIQUE INDEX XT039S2 ON CI_TD_ENTRY (ASSIGNED_TO, TD_ENTRY_ID) TABLESPACE
CM_XT039_IND COMPRESS ADVANCED LOW
/

CREATE INDEX XT039S3 ON CI_TD_ENTRY (ENTRY_STATUS_FLG, ASSIGNED_TO) TABLESPACE
CM_XT039_IND COMPRESS ADVANCED LOW
/

CREATE INDEX XT039S4 ON CI_TD_ENTRY (ROLE_ID, TD_TYPE_CD, ENTRY_STATUS_FLG,
TD_PRIORITY_FLG, ASSIGNED_TO, CRE_DTTM) TABLESPACE CM_XT039_IND COMPRESS ADVANCED LOW
/

CREATE INDEX XT039S5 ON CI_TD_ENTRY (BATCH_CD, BATCH_NBR, ENTRY_STATUS_FLG)
TABLESPACE CM_XT039_IND COMPRESS ADVANCED LOW
/

CREATE UNIQUE INDEX XT039S6 ON CI_TD_ENTRY (TD_ENTRY_ID, ASSIGNED_TO,
ENTRY_STATUS_FLG)TABLESPACE CM_XT039_IND COMPRESS ADVANCED LOW
/

CREATE UNIQUE INDEX XT039S7 ON CI_TD_ENTRY (COMPLETE_USER_ID, COMPLETE_DTTM,
TD_ENTRY_ID) TABLESPACE CM_XT039_IND COMPRESS ADVANCED LOW
/

CREATE INDEX XT039S8 ON CI_TD_ENTRY (ENTRY_STATUS_FLG, TD_TYPE_CD, MESSAGE_CAT_NBR,
MESSAGE_NBR) TABLESPACE CM_XT039_IND COMPRESS ADVANCED LOW
/

CREATE UNIQUE INDEX CM_ILM_XT039S8 ON CI_TD_ENTRY (ILM_DT, ILM_ARCH_SW, TD_ENTRY_ID)
LOCAL COMPRESS ADVANCED LOW;
```
  9. After verification of the ILM based tables, user can drop the backup tables “ILM” renamed table.
  10. Create all child Tables, Primary Key, Primary Indexes and Secondary Indexes as shown below.

Repeat the following steps for all child tables.

**Create Child Table CI\_TD\_DRLKEY**

```

CREATE TABLE CI_TD_DRLKEY
(
 TD_ENTRY_ID NOT NULL ENABLE,
 SEQ_NUM NOT NULL ENABLE,
 KEY_VALUE DEFAULT ' ' NOT NULL ENABLE,
 VERSION DEFAULT 1 NOT NULL ENABLE,
 CONSTRAINT CI_TD_DRLKEY_FK FOREIGN KEY(TD_ENTRY_ID) REFERENCES CI_TD_ENTRY
 ON DELETE CASCADE)
PARTITION BY REFERENCE (CI_TD_DRLKEY_FK)
ENABLE ROW MOVEMENT
AS SELECT /*+ PARALLEL */ * FROM ILM_CI_TD_DRLKEY;

```

**Create Index**

```

CREATE UNIQUE INDEX XT037P0 ON CI_TD_DRLKEY (TD_ENTRY_ID, SEQ_NUM) TABLESPACE
CM_XT039_IND NOLOGGING PARALLEL
GLOBAL PARTITION BY RANGE (TD_ENTRY_ID)
(
 PARTITION P1 VALUES LESS THAN ('124999999999'),
 PARTITION P2 VALUES LESS THAN ('249999999999'),
 PARTITION P3 VALUES LESS THAN ('374999999999'),
 PARTITION P4 VALUES LESS THAN ('499999999999'),
 PARTITION P5 VALUES LESS THAN ('624999999999'),
 PARTITION P6 VALUES LESS THAN ('749999999999'),
 PARTITION P7 VALUES LESS THAN ('874999999999'),
 PARTITION P8 VALUES LESS THAN (MAXVALUE)
)
COMPRESS ADVANCED LOW;

ALTER INDEX XT037P0 LOGGING NOPARALLEL;

ALTER TABLE CI_TD_DRLKEY ADD CONSTRAINT XT037P0 PRIMARY KEY(TD_ENTRY_ID, SEQ_NUM) USING
INDEX;

CREATE INDEX XT037S1 ON CI_TD_DRLKEY (KEY_VALUE, TD_ENTRY_ID) TABLESPACE CM_XT039_IND
COMPRESS ADVANCED LOW;

```



# Appendix C

---

## Sample SQL for Enabling ILM with Sub Retention in SGG (Existing Installation)

This section provides additional details including the sample syntax for each step using the Initial Measurement Data maintenance object as an example. Other maintenance object's implementations can follow a similar pattern.

1. Rename existing D1\_INIT\_MSRMT\_DATA tables and primary key indexes and constraints as a backup. It is suggested to use an ILM\_ prefix. The following are sample statements:

```
ALTER TABLE D1_INIT_MSRMT_DATA RENAME TO ILM_D1_INIT_MSRMT_DATA;
```

```
ALTER TABLE D1_INIT_MSRMT_DATA RENAME CONSTRAINT D1T304P0 TO ILM_D1T304P0;
```

```
ALTER INDEX D1T304P0 RENAME TO ILM_D1T304P0;
```

```
ALTER TABLE D1_INIT_MSRMT_DATA_CHAR RENAME TO ILM_D1_INIT_MSRMT_DATA_CHAR;
```

```
ALTER TABLE D1_INIT_MSRMT_DATA_CHAR RENAME CONSTRAINT D1T305P0 TO ILM_D1T305P0;
```

```
ALTER INDEX D1T305P0 RENAME TO ILM_D1T305P0;
```

```
ALTER TABLE D1_INIT_MSRMT_DATA_LOG RENAME TO ILM_D1_INIT_MSRMT_DATA_LOG;
```

```
ALTER TABLE D1_INIT_MSRMT_DATA_LOG RENAME CONSTRAINT D1T306P0 TO ILM_D1T306P0;
```

```
ALTER INDEX D1T306P0 RENAME TO ILM_D1T306P0;
```

```
ALTER TABLE D1_INIT_MSRMT_DATA_LOG_PARM RENAME TO ILM_D1_INIT_MSRMT_DATA_LOG_PARM;
```

```
ALTER TABLE D1_INIT_MSRMT_DATA_LOG_PARM RENAME CONSTRAINT D1T307P0 TO ILM_D1T307P0;
```

```
ALTER INDEX D1T307P0 RENAME TO ILM_D1T307P0;
```

```
ALTER TABLE D1_INIT_MSRMT_DATA_K RENAME TO ILM_D1_INIT_MSRMT_DATA_K;
```

```
ALTER TABLE D1_INIT_MSRMT_DATA_K RENAME CONSTRAINT D1T314P0 TO
ILM_D1T314P0;
```

```
ALTER INDEX D1T314P0 RENAME TO ILM_D1T314P0;
```

## 2. Generate DDL for the secondary index.

```
set heading off;
set echo off;
Set pages 999;
set long 90000;
```

```
spool ddl_list.sql
select dbms_metadata.get_ddl('INDEX','D1T304S1','CISADM') from
dual;
spool off;
```

## 3. Drop secondary indexes.

```
DROP INDEX CISADM.D1T304S1;
```

## 4. Create Partitioned Table.

In the following example ILM\_DT value is inserted from column CRE\_DTTM. The degree setting of 'parallel' in the DDL can be adjusted according to the table's data, its means and its size. Use the CTAS queries listed in Chapter 5 to create temporary tables for ACTIVITY, DEVICE EVENT, and INITIAL MEASUREMENT DATA and use the following statements to create the partitioned tables.

### Activity

```
CREATE TABLE D1_ACTIVITY (
D1_ACTIVITY_ID NOT NULL,
BUS_OBJ_CD NOT NULL,
BO_STATUS_CD NOT NULL,
ACTIVITY_TYPE_CD NOT NULL,
START_DTTM NOT NULL,
END_DTTM,
CRE_DTTM NOT NULL,
STATUS_UPD_DTTM NOT NULL,
BO_STATUS_REASON_CD NOT NULL,
VERSION NOT NULL,
EFF_DTTM,
BO_DATA_AREA,
FIELD_TASK_TYPE,
CANCEL_REASON,
ILM_DT,
ILM_ARCH_SW,
RETENTION_PERIOD NOT NULL
)
AS
SELECT
A.D1_ACTIVITY_ID,
A.BUS_OBJ_CD,
A.BO_STATUS_CD,
A.ACTIVITY_TYPE_CD,
A.START_DTTM,
A.END_DTTM,
A.CRE_DTTM,
A.STATUS_UPD_DTTM,
A.BO_STATUS_REASON_CD,
A.VERSION,
A.EFF_DTTM,
A.BO_DATA_AREA,
A.FIELD_TASK_TYPE,
A.CANCEL_REASON,
A.CRE_DTTM as ILM_DT,
'N' as ILM_ARCH_SW,
CAST(COALESCE((SELECT B.RETPERIOD
FROM ILM_ACTIVITY_RETENTION_TMP B
WHERE B.ACTIVITY_TYPE_CD = A.ACTIVITY_TYPE_CD)
,CAST((select maint_obj_opt_val
from ci_md_mo_opt mmound1
where maint_obj_cd = 'D1-ACTIVITY'
```

```

and maint_obj_opt_flg = 'FLRP'
and seq_num =
(select max(seq_num)
from ci_md_mo_opt mmo
where maint_obj_cd = 'D1-ACTIVITY'
and maint_obj_opt_flg = 'FLRP')) as NUMBER(5)
,CAST((select extractvalue(xmlparse(content fw_mcfg.mst_config_data)
,'generalMasterConfiguration/defaultRetentionPeriod')
from fl_mst_config fw_mcfg
where fw_mcfg.bus_obj_cd = 'F1-ILMMSConfig') as NUMBER(5)
, 99999) as NUMBER(5)) as RETENTION_PERIOD
FROM ILM_D1_ACTIVITY A
/

```

### Device Event

```

CREATE TABLE D1_DVC_EVT (
DVC_EVT_ID NOT NULL,
DVC_EVT_TYPE_CD,
BUS_OBJ_CD NOT NULL,
EXT_EVT_NAME_FLG,
D1_SPR_CD,
BO_STATUS_CD NOT NULL,
STATUS_UPD_DTTM NOT NULL,
BO_STATUS_REASON_CD NOT NULL,
DVC_EVT_DTTM NOT NULL,
CRE_DTTM NOT NULL,
VERSION NOT NULL,
DVC_EVT_END_DTTM,
BO_DATA_AREA,
D1_DEVICE_ID,
ILM_DT NOT NULL,
ILM_ARCH_SW,
RETENTION_PERIOD NOT NULL)
AS
SELECT
A.DVC_EVT_ID,
A.DVC_EVT_TYPE_CD,
A.BUS_OBJ_CD,
A.EXT_EVT_NAME_FLG,
A.D1_SPR_CD,
A.BO_STATUS_CD,
A.STATUS_UPD_DTTM,
A.BO_STATUS_REASON_CD,
A.DVC_EVT_DTTM,
A.CRE_DTTM,
A.VERSION,
A.DVC_EVT_END_DTTM,
A.BO_DATA_AREA,
A.D1_DEVICE_ID,
A.CRE_DTTM as ILM_DT,
'N' as ILM_ARCH_SW,
CAST(COALESCE((SELECT B.RETPERIOD
FROM ILM_DVC_EVT_RETENTION_TMP B
WHERE B.DVC_EVT_TYPE_CD = A.DVC_EVT_TYPE_CD)
,CAST((select maint_obj_opt_val
from ci_md_mo_opt mmo
where maint_obj_cd = 'D1-DVCEVENT'
and maint_obj_opt_flg = 'FLRP'
and seq_num =
(select max(seq_num)
from ci_md_mo_opt mmo
where maint_obj_cd = 'D1-DVCEVENT'
and maint_obj_opt_flg = 'FLRP')) as NUMBER(5)
,CAST((select extractvalue(xmlparse(content fw_mcfg.mst_config_data)
,'generalMasterConfiguration/defaultRetentionPeriod')
from fl_mst_config fw_mcfg
where fw_mcfg.bus_obj_cd = 'F1-ILMMSConfig') as NUMBER(5)
, 99999) as NUMBER(5)) as RETENTION_PERIOD
FROM ILM_D1_DVC_EVT A
/

```

### Initial Measurement Data

```

CREATE TABLE ILM_IMD_RETENTION_TMP
AS
select mct.measr_comp_type_cd
/*retrieve the retention period for MC Types in this order of precedence:
1. The UOM based retention period from the MDM master configuration
2. The interval IMD retention period from the MDM master configuration

```

```

3. The MO level retention period from the MO options
4. The installation level retention period from the FW master configuration
*/
, CAST(coalesce((select retPeriod
from (select 'D1IN' interval_scalar_flg
, extractvalue(value(p),'uomRetentionPeriodList/uom') D1_UOM_CD
, extractvalue(value(p),'uomRetentionPeriodList/retentionPeriod') retPeriod
from fl_mst_config mdm_mcfg
, table(xmlsequence(extract(xmlparse(content
mdm_mcfg.mst_config_data),
'imdRetentionPeriod/intervalImdRetentionPeriods/uomRetentionPeriods/
uomRetentionPeriodList')) p
where mdm_mcfg.bus_obj_cd = 'D1-ILMMSConfig'
union
select 'D1SC' INTERVAL_SCALAR_FLG
, extractvalue(value(p),'uomRetentionPeriodList/uom') D1_UOM_CD
, extractvalue(value(p),'uomRetentionPeriodList/retentionPeriod') retPeriod
from fl_mst_config mdm_mcfg
, table(xmlsequence(extract(xmlparse(content
mdm_mcfg.mst_config_data),
'imdRetentionPeriod/scalarImdRetentionPeriods/uomRetentionPeriods/
uomRetentionPeriodList')) p
where mdm_mcfg.bus_obj_cd = 'D1-ILMMSConfig') uomMap
where uomMap.interval_scalar_flg = mct.interval_scalar_flg
and trim(mctvi.dl_uom_cd) = trim(uomMap.dl_uom_cd)--UOM
, DECODE(mct.interval_scalar_flg
,'D1IN'
,extractvalue(xmlparse(content mdm_mcfg.mst_config_data),
'imdRetentionPeriod/intervalImdRetentionPeriods/intervalRetentionPeriod') --interval IMD
,extractvalue(xmlparse(content mdm_mcfg.mst_config_data),
'imdRetentionPeriod/scalarImdRetentionPeriods/scalarRetentionPeriod') --scalar IMD
)
, (select maint_obj_opt_val
from ci_md_mo_opt mmo
where maint_obj_cd = 'D1-IMD'
and maint_obj_opt_flg = 'FLRP'
and seq_num = (select max(seq_num)
from ci_md_mo_opt mmo
where maint_obj_cd = 'D1-IMD'
and maint_obj_opt_flg = 'FLRP')) --IMD
, extractvalue(xmlparse(content fw_mcfg.mst_config_data),
'generalMasterConfiguration/defaultRetentionPeriod') --Install
) as NUMBER(5)) retPeriod
from dl_measr_comp_type mct
, dl_mc_type_value_identifier mctvi
, fl_mst_config fw_mcfg
, fl_mst_config mdm_mcfg
where mct.measr_comp_type_cd = mctvi.measr_comp_type_cd
and mctvi.value_id_type_flg = 'D1MS'
and fw_mcfg.bus_obj_cd = 'F1-ILMMSConfig'
and mdm_mcfg.bus_obj_cd = 'D1-ILMMSConfig'
order by 1;

```

```

CREATE BIGFILE TABLESPACE CM_D1T304_P2011JAN_S181 DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011JAN_SMAX DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011FEB_S181 DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011FEB_SMAX DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011MAR_S181 DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011MAR_SMAX DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011APR_S181 DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011APR_SMAX DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011MAY_S181 DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011MAY_SMAX DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011JUN_S181 DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011JUN_SMAX DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011JUL_S181 DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011JUL_SMAX DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011AUG_S181 DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;

```

```

CREATE BIGFILE TABLESPACE CM_D1T304_P2011AUG_SMAX DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011SEP_S181 DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011SEP_SMAX DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011OCT_S181 DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011OCT_SMAX DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011NOV_S181 DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011NOV_SMAX DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011DEC_S181 DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011DEC_SMAX DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_PMAX_S181 DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
CREATE BIGFILE TABLESPACE CM_D1T304_PMAX_SMAX DATAFILE '+DATADG' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;

CREATE TABLE D1_INIT_MSRMT_DATA
(
 INIT_MSRMT_DATA_ID NOT NULL,
 MEASR_COMP_ID NOT NULL,
 D1_FROM_DTTM,
 D1_TO_DTTM,
 DATA_SRC_FLG NOT NULL,
 TIME_ZONE_CD NOT NULL,
 BUS_OBJ_CD NOT NULL,
 BO_STATUS_CD NOT NULL,
 BO_STATUS_REASON_CD NOT NULL,
 IMD_BO_DATA_AREA,
 STATUS_UPD_DTTM NOT NULL,
 CRE_DTTM NOT NULL,
 VERSION NOT NULL,
 IMD_EXT_ID,
 PREVEE_BO_DATA_AREA,
 POSTVEE_BO_DATA_AREA,
 TRACE_BO_DATA_AREA,
 RAW_BO_DATA_AREA,
 LAST_UPDATE_DTTM,
 ILM_DT,
 ILM_ARCH_SW,
 RETENTION_PERIOD NOT NULL
)
nologging parallel (degree 10)
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM CACHE)
PARTITION BY RANGE (ILM_DT) SUBPARTITION BY RANGE (RETENTION_PERIOD)
(
 PARTITION "P2011JAN" VALUES LESS THAN (TO_DATE('2011-02-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN')) (
 SUBPARTITION P2011JAN_S181 VALUES LESS THAN (181) TABLESPACE CM_D1T304_P2011JAN_S181
 LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JAN_S181)
 LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JAN_S181)
 LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JAN_S181)
 LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JAN_S181)
 LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JAN_S181)
 ,
 SUBPARTITION P2011JAN_SMAX VALUES LESS THAN (MAXVALUE) TABLESPACE CM_D1T304_P2011JAN_SMAX
 LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JAN_SMAX)
 LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JAN_SMAX)
 LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JAN_SMAX)
 LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JAN_SMAX)
 LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JAN_SMAX)
),
 PARTITION "P2011FEB" VALUES LESS THAN (TO_DATE('2011-03-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN')) (
 SUBPARTITION P2011FEB_S181 VALUES LESS THAN (181) TABLESPACE CM_D1T304_P2011FEB_S181
 LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011FEB_S181)
 LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011FEB_S181)
 LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011FEB_S181)
 LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011FEB_S181)
 LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011FEB_S181)
 ,
 SUBPARTITION P2011FEB_SMAX VALUES LESS THAN (MAXVALUE) TABLESPACE CM_D1T304_P2011FEB_SMAX
 LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011FEB_SMAX)

```

```

LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011FEB_SMAX)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011FEB_SMAX)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011FEB_SMAX)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011FEB_SMAX)
),
PARTITION "P2011MAR" VALUES LESS THAN (TO_DATE('2011-04-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))(
SUBPARTITION P2011MAR_S181 VALUES LESS THAN (181) TABLESPACE CM_D1T304_P2011MAR_S181
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAR_S181)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAR_S181)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAR_S181)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAR_S181)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAR_S181)
),
SUBPARTITION P2011MAR_SMAX VALUES LESS THAN (MAXVALUE) TABLESPACE CM_D1T304_P2011MAR_SMAX
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAR_SMAX)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAR_SMAX)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAR_SMAX)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAR_SMAX)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAR_SMAX)
),
PARTITION "P2011APR" VALUES LESS THAN (TO_DATE('2011-05-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))(
SUBPARTITION P2011APR_S181 VALUES LESS THAN (181) TABLESPACE CM_D1T304_P2011APR_S181
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011APR_S181)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011APR_S181)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011APR_S181)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011APR_S181)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011APR_S181)
),
SUBPARTITION P2011APR_SMAX VALUES LESS THAN (MAXVALUE) TABLESPACE CM_D1T304_P2011APR_SMAX
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011APR_SMAX)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011APR_SMAX)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011APR_SMAX)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011APR_SMAX)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011APR_SMAX)
),
PARTITION "P2011MAY" VALUES LESS THAN (TO_DATE('2011-06-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))(
SUBPARTITION P2011MAY_S181 VALUES LESS THAN (181) TABLESPACE CM_D1T304_P2011MAY_S181
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAY_S181)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAY_S181)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAY_S181)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAY_S181)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAY_S181)
),
SUBPARTITION P2011MAY_SMAX VALUES LESS THAN (MAXVALUE) TABLESPACE CM_D1T304_P2011MAY_SMAX
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAY_SMAX)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAY_SMAX)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAY_SMAX)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAY_SMAX)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011MAY_SMAX)
),
PARTITION "P2011JUN" VALUES LESS THAN (TO_DATE('2011-07-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))(
SUBPARTITION P2011JUN_S181 VALUES LESS THAN (181) TABLESPACE CM_D1T304_P2011JUN_S181
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUN_S181)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUN_S181)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUN_S181)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUN_S181)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUN_S181)
),
SUBPARTITION P2011JUN_SMAX VALUES LESS THAN (MAXVALUE) TABLESPACE CM_D1T304_P2011JUN_SMAX
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUN_SMAX)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUN_SMAX)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUN_SMAX)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUN_SMAX)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUN_SMAX)
),
PARTITION "P2011JUL" VALUES LESS THAN (TO_DATE('2011-08-01 00:00:01', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))(
SUBPARTITION P2011JUL_S181 VALUES LESS THAN (181) TABLESPACE CM_D1T304_P2011JUL_S181
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUL_S181)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUL_S181)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUL_S181)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUL_S181)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUL_S181)
),
SUBPARTITION P2011JUL_SMAX VALUES LESS THAN (MAXVALUE) TABLESPACE CM_D1T304_P2011JUL_SMAX
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUL_SMAX)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUL_SMAX)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUL_SMAX)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUL_SMAX)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_P2011JUL_SMAX)
),

```



```

LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_PMAX_S181)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_PMAX_S181)
,
SUBPARTITION PMAX_SMAX VALUES LESS THAN (MAXVALUE) TABLESPACE CM_D1T304_PMAX_SMAX
LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_PMAX_SMAX)
LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_PMAX_SMAX)
LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_PMAX_SMAX)
LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_PMAX_SMAX)
LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE CM_D1T304_PMAX_SMAX)
)) ENABLE ROW MOVEMENT AS
SELECT
A.INIT_MSRMT_DATA_ID,
A.MEASR_COMP_ID,
A.D1_FROM_DTTM,
A.D1_TO_DTTM,
A.DATA_SRC_FLG,
A.TIME_ZONE_CD,
A.BUS_OBJ_CD,
A.BO_STATUS_CD,
A.BO_STATUS_REASON_CD,
A.IMD_BO_DATA_AREA,
A.STATUS_UPD_DTTM,
A.CRE_DTTM,
A.VERSION,
A.IMD_EXT_ID,
A.PREVEE_BO_DATA_AREA,
A.POSTVEE_BO_DATA_AREA,
A.TRACE_BO_DATA_AREA,
A.RAW_BO_DATA_AREA,
A.LAST_UPDATE_DTTM,
A.CRE_DTTM as ILM_DT,
'N' as ILM_ARCH_SW,
CAST(COALESCE((SELECT C.RETPERIOD
FROM D1_MEASR_COMP B, ILM_IMD_RETENTION_TMP C
WHERE B.MEASR_COMP_ID = A.MEASR_COMP_ID
AND C.MEASR_COMP_TYPE_CD = B.MEASR_COMP_TYPE_CD)
,CAST(select maint_obj_opt_val
from ci_md_mo_opt mmo
where maint_obj_cd = 'D1-IMD'
and maint_obj_opt_flg = 'FLRP'
and seq_num =
(select max(seq_num)
from ci_md_mo_opt mmo
where maint_obj_cd = 'D1-IMD'
and maint_obj_opt_flg = 'FLRP')) as NUMBER(5))
,CAST(select extractvalue(xmlparse(content fw_mcfg.mst_config_data)
,'generalMasterConfiguration/defaultRetentionPeriod')
from fl_mst_config fw_mcfg
where fw_mcfg.bus_obj_cd = 'F1-ILMMSConfig') as NUMBER(5))
, 99999) as NUMBER(5)) as RETENTION_PERIOD
FROM ILM_D1_INIT_MSRMT_DATA A
/

```

5. Enable logging option for table D1\_INIT\_MSRMT\_DATA.

```
ALTER TABLE D1_INIT_MSRMT_DATA NOPARALLEL LOGGING;
```

6. Create Primary Index for Parent table D1\_INIT\_MSRMT\_DATA.

```
CREATE BIGFILE TABLESPACE CM_D1T304_IND DATAFILE '+DATA' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
```

```
CREATE UNIQUE INDEX D1T304P0 ON D1_INIT_MSRMT_DATA NOLOGGING PARALLEL
(INIT_MSRMT_DATA_ID)
GLOBAL PARTITION BY RANGE (INIT_MSRMT_DATA_ID) (
PARTITION P1 VALUES LESS THAN ('124999999999999'),
PARTITION P2 VALUES LESS THAN ('249999999999999'),
PARTITION P3 VALUES LESS THAN ('374999999999999'),
PARTITION P4 VALUES LESS THAN ('499999999999999'),
PARTITION P5 VALUES LESS THAN ('624999999999999'),
PARTITION P6 VALUES LESS THAN ('744999999999999'),
PARTITION P7 VALUES LESS THAN ('874999999999999'),
PARTITION P8 VALUES LESS THAN (MAXVALUE)
) COMPRESS ADVANCED LOW
/

```

```
ALTER INDEX D1T304P0 LOGGING NOPARALLEL;
```

7. Add Primary Key for Parent table D1\_INIT\_MSRMT\_DATA

```
ALTER TABLE D1_INIT_MSRMT_DATA ADD CONSTRAINT D1T304P0 PRIMARY KEY(INIT_MSRMT_DATA_ID)
USING INDEX
/

```



## 8. Create Secondary Indexes for Parent table D1\_INIT\_MSRMT\_DATA

```

CREATE INDEX D1T304S1 ON D1_INIT_MSRMT_DATA (MEASR_COMP_ID, D1_TO_DTTM)
GLOBAL PARTITION BY RANGE (MEASR_COMP_ID) (
PARTITION P1 VALUES LESS THAN ('1249999999999'),
PARTITION P2 VALUES LESS THAN ('2499999999999'),
PARTITION P3 VALUES LESS THAN ('3749999999999'),
PARTITION P4 VALUES LESS THAN ('4999999999999'),
PARTITION P5 VALUES LESS THAN ('6249999999999'),
PARTITION P6 VALUES LESS THAN ('7499999999999'),
PARTITION P7 VALUES LESS THAN ('8749999999999'),
PARTITION P8 VALUES LESS THAN (MAXVALUE)
) COMPRESS ADVANCED LOW
/

CREATE UNIQUE INDEX CM_ILM_D1T304S4 ON D1_INIT_MSRMT_DATA (ILM_DT, RETENTION_PERIOD,
ILM_ARCH_SW, INIT_MSRMT_DATA_ID) LOCAL COMPRESS ADVANCED LOW
/

```

## 9. Create Child Tables, Primary Key, Primary Indexes and Secondary Indexes as shown below.

**Create Child Table D1\_INIT\_MSRMT\_DATA\_CHAR**

```

CREATE TABLE D1_INIT_MSRMT_DATA_CHAR
(
INIT_MSRMT_DATA_ID NOT NULL ENABLE,
CHAR_TYPE_CD NOT NULL ENABLE,
SEQ_NUM NOT NULL ENABLE,
CHAR_VAL DEFAULT ' ' NOT NULL ENABLE,
ADHOC_CHAR_VAL DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK1 DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK2 DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK3 DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK4 DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK5 DEFAULT ' ' NOT NULL ENABLE,
SRCH_CHAR_VAL DEFAULT ' ' NOT NULL ENABLE,
VERSION DEFAULT 1 NOT NULL ENABLE,
LAST_UPDATE_DTTM ,
CONSTRAINT D1_INIT_MSRMT_DATA_CHAR_FK FOREIGN KEY (INIT_MSRMT_DATA_ID) REFERENCES
D1_INIT_MSRMT_DATA ON DELETE CASCADE)
PARTITION BY REFERENCE (D1_INIT_MSRMT_DATA_CHAR_FK) ENABLE ROW MOVEMENT NOLOGGING
PARALLEL
AS SELECT /*+ PARALLEL */ * FROM ILM_D1_INIT_MSRMT_DATA_CHAR
/

ALTER TABLE D1_INIT_MSRMT_DATA_CHAR LOGGING NOPARALLEL
/

```

**Create Primary Index for Child Table D1\_INIT\_MSRMT\_DATA\_CHAR**

```

CREATE UNIQUE INDEX D1T305P0 ON D1_INIT_MSRMT_DATA_CHAR (INIT_MSRMT_DATA_ID,
CHAR_TYPE_CD, SEQ_NUM)
TABLESPACE CM_D1T304_IND NOLOGGING PARALLEL
GLOBAL PARTITION BY RANGE (INIT_MSRMT_DATA_ID) (
PARTITION P1 VALUES LESS THAN ('1249999999999'),
PARTITION P2 VALUES LESS THAN ('2499999999999'),
PARTITION P3 VALUES LESS THAN ('3749999999999'),
PARTITION P4 VALUES LESS THAN ('4999999999999'),
PARTITION P5 VALUES LESS THAN ('6249999999999'),
PARTITION P6 VALUES LESS THAN ('7499999999999'),
PARTITION P7 VALUES LESS THAN ('8749999999999'),
PARTITION P8 VALUES LESS THAN (MAXVALUE)
) COMPRESS ADVANCED LOW
/

ALTER INDEX D1T305P0 LOGGING NOPARALLEL
/

```

**Create Primary Key for Child Table D1\_INIT\_MSRMT\_DATA\_CHAR**

```

ALTER TABLE D1_INIT_MSRMT_DATA_CHAR ADD CONSTRAINT D1T305P0 PRIMARY KEY
(INIT_MSRMT_DATA_ID, CHAR_TYPE_CD, SEQ_NUM) USING INDEX
/

```

**Create Secondary Indexes for Child Table D1\_INIT\_MSRMT\_DATA\_CHAR**

```

CREATE INDEX D1T305S1 ON D1_INIT_MSRMT_DATA_CHAR(SRCH_CHAR_VAL) GLOBAL PARTITION BY
HASH(SRCH_CHAR_VAL)
(
PARTITION P1 TABLESPACE CM_D1T304_IND,
PARTITION P2 TABLESPACE CM_D1T304_IND,
PARTITION P3 TABLESPACE CM_D1T304_IND,
PARTITION P4 TABLESPACE CM_D1T304_IND,
PARTITION P5 TABLESPACE CM_D1T304_IND,
PARTITION P6 TABLESPACE CM_D1T304_IND,
PARTITION P7 TABLESPACE CM_D1T304_IND,
PARTITION P8 TABLESPACE CM_D1T304_IND
)
/

```

**Create Child Table D1\_INIT\_MSRMT\_DATA\_LOG**

```

CREATE TABLE D1_INIT_MSRMT_DATA_LOG (
INIT_MSRMT_DATA_ID NOT NULL ENABLE,
SEQNO NOT NULL ENABLE,
BO_STATUS_CD DEFAULT ' ' NOT NULL ENABLE,
BO_STATUS_REASON_CD DEFAULT ' ' NOT NULL ENABLE,
CHAR_TYPE_CD DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL DEFAULT ' ' NOT NULL ENABLE,
ADHOC_CHAR_VAL DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK1 DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK2 DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK3 DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK4 DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK5 DEFAULT ' ' NOT NULL ENABLE,
DESCRLONG DEFAULT ' ' NOT NULL ENABLE,
LOG_DTTM NOT NULL ENABLE,
LOG_ENTRY_TYPE_FLG DEFAULT ' ' NOT NULL ENABLE,
MESSAGE_CAT_NBR DEFAULT 0 NOT NULL ENABLE,
MESSAGE_NBR DEFAULT 0 NOT NULL ENABLE,
USER_ID DEFAULT ' ' NOT NULL ENABLE,
VERSION DEFAULT 1 NOT NULL ENABLE,
LAST_UPDATE_DTTM,
CONSTRAINT D1_INIT_MSRMT_DATA_LOG_FK FOREIGN KEY(INIT_MSRMT_DATA_ID) REFERENCES
D1_INIT_MSRMT_DATA ON DELETE CASCADE)
PARTITION BY REFERENCE (D1_INIT_MSRMT_DATA_LOG_FK) ENABLE ROW MOVEMENT NOLOGGING
PARALLEL
AS SELECT /*+ PARALLEL */ * FROM ILM_D1_INIT_MSRMT_DATA_LOG
/

ALTER TABLE D1_INIT_MSRMT_DATA_LOG LOGGING NOPARALLEL
/

```

**Create Primary Index for Child Table D1\_INIT\_MSRMT\_DATA\_LOG**

```

CREATE UNIQUE INDEX D1T306P0 ON D1_INIT_MSRMT_DATA_LOG(INIT_MSRMT_DATA_ID, SEQNO)
TABLESPACE CM_D1T304_IND NOLOGGING PARALLEL
GLOBAL PARTITION BY RANGE(INIT_MSRMT_DATA_ID) (
PARTITION P1 VALUES LESS THAN ('124999999999999'),
PARTITION P2 VALUES LESS THAN ('249999999999999'),
PARTITION P3 VALUES LESS THAN ('374999999999999'),
PARTITION P4 VALUES LESS THAN ('499999999999999'),
PARTITION P5 VALUES LESS THAN ('624999999999999'),
PARTITION P6 VALUES LESS THAN ('749999999999999'),
PARTITION P7 VALUES LESS THAN ('874999999999999'),
PARTITION P8 VALUES LESS THAN (MAXVALUE)
) COMPRESS ADVANCED LOW
/

ALTER INDEX D1T306P0 LOGGING NOPARALLEL
/

```

**Create Primary Key for Child Table D1\_INIT\_MSRMT\_DATA\_LOG**

```

ALTER TABLE D1_INIT_MSRMT_DATA_LOG ADD CONSTRAINT D1T306P0 PRIMARY KEY
(INIT_MSRMT_DATA_ID, SEQNO) USING INDEX
/

```

**Create Child Table D1\_INIT\_MSRMT\_DATA\_LOG\_PARM**

```

CREATE TABLE D1_INIT_MSRMT_DATA_LOG_PARM (

```

```

INIT_MSRMT_DATA_ID NOT NULL ENABLE,
SEQNO NOT NULL ENABLE,
P_ARM_SEQ NOT NULL ENABLE,
MSG_PARM_VAL DEFAULT ' ' NOT NULL ENABLE,
MSG_PARM_TYP_FLG DEFAULT ' ' NOT NULL ENABLE,
VERSION DEFAULT 1 NOT NULL ENABLE,
LAST_UPDATE_DTTM ,
CONSTRAINT D1_INIT_MSRMT_DATA_LOG_PARM_FK FOREIGN KEY(INIT_MSRMT_DATA_ID) REFERENCE
D1_INIT_MSRMT_DATA ON DELETE CASCADE)
PARTITION BY REFERENCE (D1_INIT_MSRMT_DATA_LOG_PARM_FK) ENABLE ROW MOVEMENT NOLOGGING
PARALLEL
AS SELECT /*+ PARALLEL */ * FROM ILM_D1_INIT_MSRMT_DATA_LOG_PARM
/

ALTER TABLE D1_INIT_MSRMT_DATA_LOG_PARM LOGGING NOPARALLEL
/

```

### Create Primary Index for Child Table D1\_INIT\_MSRMT\_DATA\_LOG\_PARM

```

CREATE UNIQUE INDEX D1T307P0 ON D1_INIT_MSRMT_DATA_LOG_PARM(INIT_MSRMT_DATA_ID, SEQNO,
P_ARM_SEQ)
TABLESPACE CM_D1T304_IND NOLOGGING PARALLEL GLOBAL PARTITION BY
RANGE(INIT_MSRMT_DATA_ID) (

PARTITION P1 VALUES LESS THAN ('124999999999999'),
PARTITION P2 VALUES LESS THAN ('249999999999999'),
PARTITION P3 VALUES LESS THAN ('374999999999999'),
PARTITION P4 VALUES LESS THAN ('499999999999999'),
PARTITION P5 VALUES LESS THAN ('624999999999999'),
PARTITION P6 VALUES LESS THAN ('749999999999999'),
PARTITION P7 VALUES LESS THAN ('874999999999999'),
PARTITION P8 VALUES LESS THAN (MAXVALUE)
) COMPRESS ADVANCED LOW
/

ALTER INDEX D1T306P0 LOGGING NOPARALLEL
/

```

### Create Primary Key for Child Table D1\_INIT\_MSRMT\_DATA\_LOG\_PARM

```

ALTER TABLE D1_INIT_MSRMT_DATA_LOG ADD CONSTRAINT D1T307P0 PRIMARY KEY
(INIT_MSRMT_DATA_ID, SEQNO, P_ARM_SEQ) USING INDEX
/

```

### Create Child Table D1\_INIT\_MSRMT\_DATA\_K

```

CREATE BIGFILE TABLESPACE CM_D1T314_IND DATAFILE '+DATA' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED;

CREATE TABLE D1_INIT_MSRMT_DATA_K (
INIT_MSRMT_DATA_ID NOT NULL ENABLE,
ENV_ID NOT NULL ENABLE,
CONSTRAINT D1T314P0 PRIMARY KEY (INIT_MSRMT_DATA_ID, ENV_ID) ENABLE
)
ORGANIZATION INDEX
Partition by range(INIT_MSRMT_DATA_ID) (
PARTITION P1 VALUES LESS THAN ('124999999999999'),
PARTITION P2 VALUES LESS THAN ('249999999999999'),
PARTITION P3 VALUES LESS THAN ('374999999999999'),
PARTITION P4 VALUES LESS THAN ('499999999999999'),
PARTITION P5 VALUES LESS THAN ('624999999999999'),
PARTITION P6 VALUES LESS THAN ('749999999999999'),
PARTITION P7 VALUES LESS THAN ('874999999999999'),
PARTITION P8 VALUES LESS THAN (MAXVALUE)
)
TABLESPACE CM_D1T314_IND
AS SELECT /*+ PARALLEL */ * FROM ILM_D1_INIT_MSRMT_DATA_K
/

ALTER TABLE D1_INIT_MSRMT_DATA_K LOGGING NOPARALLEL
/

```

- After verification of the ILM based tables, the user can drop the backup “ILM” renamed tables.

# Appendix D

---

## Sample SQL for Periodic Maintenance

This appendix provides additional details related to creating new partitions over time as well as archiving and restoring partitions. The To Do Entry and Inbound Sync Request maintenance objects are used as examples. This section contains the following steps:

- [Add Partition](#)
- [Archive Partition](#)
- [Restore Partition](#)

## Add Partition

1. Create separate tablespace for new partition

```
CREATE BIGFILE TABLESPACE CM_XT039_P2016JAN DATAFILE '+DATA' SIZE 50M AUTOEXTEND ON
MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
```

2. Add partition using split operation on MAXVALUE Partition

```
ALTER TABLE CISADM.CI_TD_ENTRY SPLIT PARTITION PMAX AT (TO_DATE('2016-02-01
00:00:01','SYYYYY-MM-DD HH24:MI:SS'))
INTO
(
PARTITION P2016JAN TABLESPACE CM_XT039_P2016JAN, PARTITION PMAX
)
UPDATE INDEXES;
```

In case table contains LOBS like F1\_SYNC\_REQ\_IN, there will be additional statement in split partition DDL indicating tablespace on which LOB should go.

```
ALTER TABLE CISADM.F1_SYNC_REQ_IN SPLIT PARTITION PMAX AT (TO_DATE('2016-02-01
00:00:01','SYYYYY-MM-DD HH24:MI:SS'))
INTO
(
PARTITION P2016JAN TABLESPACE CM_FIT191_P2016JAN
LOB(BO_DATA_AREA, POST_TRN_BO_DATA_AREA, PRE_TRN_FIN_BO_DATA_AREA,
PRE_TRN_INIT_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS MEDIUM
CACHE TABLESPACE CM_FIT191_P2016JAN)
,
PARTITION PMAX
)
UPDATE INDEXES;
```

3. Enable advanced compression after SPLIT partition as it will disable the compression.

```
ALTER TABLE CISADM.CI_TD_SRTKEY ROW STORE COMPRESS ADVANCED;
ALTER TABLE CISADM.CI_TD_MSG_PARM ROW STORE COMPRESS ADVANCED;
ALTER TABLE CISADM.CI_TD_DRLKEY ROW STORE COMPRESS ADVANCED;
ALTER TABLE CISADM.CI_TD_ENTRY_CHA ROW STORE COMPRESS ADVANCED;
ALTER TABLE CISADM.CI_TD_LOG ROW STORE COMPRESS ADVANCED;
```

## Archive Partition

1. Make the tablespace to be archived READ ONLY.

```
ALTER TABLESPACE CM_XT039_P2017JAN READ ONLY;
```

2. Check the feasibility of archive using ILM\_ARCH\_SW = 'N'.

```
Select count(1) from CISADM.CI_TD_ENTRY PARTITION P2017JAN where ILM_ARCH_SW = 'N';
```

- If Yes (count of records of above query is ZERO), then proceed for further steps.
- If No (count of records of above query is Non ZERO), then make the tablespace back to READ WRITE MODE as Archive is not Feasible at the time.

```
ALTER TABLESPACE CM_XT039_P2017JAN READ WRITE;
```

3. Create separate archive tablespace for partition need to be archived.

```
CREATE BIGFILE TABLESPACE CM_XT039_P2017JAN_ARC DATAFILE '+DATA' SIZE 50M AUTOEXTEND
ON MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;
```

4. Create staging tables and load data for all child tables for the MO first.

- a. CI\_TD\_ENTRY\_CHA

```

CREATE TABLE CM_XT701_P2017JAN_ARC PARALLEL NOLOGGING
TABLESPACE CM_XT039_P2017JAN_ARC
AS
(
SELECT /*+ PARALLEL */ * FROM CISADM.CI_TD_ENTRY_CHA PARTITION
(P2017JAN_S01)
UNION ALL
SELECT /*+ PARALLEL */ * FROM CI_TD_ENTRY_CHA PARTITION
(P2017JAN_S02)
UNION ALL
.
.
.
UNION ALL
SELECT /*+ PARALLEL */ * FROM CI_TD_ENTRY_CHA PARTITION
(P2017JAN_S08)
);
ALTER TABLE CM_XT701_P2017JAN_ARC NOPARALLEL LOGGING;

```

**b. CI\_TD\_MSG\_PARM**

```

CREATE TABLE CM_XT04_P2017JAN_ARC PARALLEL NOLOGGING TABLESPACE
CM_XT039_P2017JAN_ARC
AS
(
SELECT /*+ PARALLEL */ * FROM CISADM.CI_TD_MSG_PARM PARTITION
(P2017JAN_S01)
UNION ALL
SELECT /*+ PARALLEL */ * FROM CI_TD_MSG_PARM PARTITION
(P2017JAN_S02)
UNION ALL
.
.
.
UNION ALL
SELECT /*+ PARALLEL */ * FROM CI_TD_MSG_PARM PARTITION
(P2017JAN_S08)
);
ALTER TABLE CM_XT04_P2017JAN_ARC NOPARALLEL LOGGING;

```

**c. CI\_TD\_LOG**

```

CREATE TABLE CM_XT721_P2017JAN_ARC PARALLEL NOLOGGING
TABLESPACE CM_XT039_P2017JAN_ARC
AS
(
SELECT /*+ PARALLEL */ * FROM CISADM.CI_TD_LOG PARTITION
(P2017JAN_S01)
UNION ALL
SELECT /*+ PARALLEL */ * FROM CI_TD_LOG PARTITION (P2017JAN_S02)
UNION ALL
.
.
.
UNION ALL
SELECT /*+ PARALLEL */ * FROM CI_TD_LOG PARTITION (P2017JAN_S08)
);
ALTER TABLE CM_XT721_P2017JAN_ARC NOPARALLEL LOGGING;

```

## d. CI\_TD\_SRTKEY

```

CREATE TABLE CM_XT041_P2017JAN_ARC PARALLEL NOLOGGING
TABLESPACE CM_XT039_P2017JAN_ARC
AS
(
SELECT /*+ PARALLEL */ * FROM CISADM.CI_TD_SRTKEY PARTITION
(P2017JAN_S01)
UNION ALL
SELECT /*+ PARALLEL */ * FROM CI_TD_SRTKEY PARTITION
(P2017JAN_S02)
UNION ALL
.
.
.
UNION ALL
SELECT /*+ PARALLEL */ * FROM CI_TD_SRTKEY PARTITION
(P2017JAN_S08)
);
ALTER TABLE CM_XT041_P2017JAN_ARC NOPARALLEL LOGGING;

```

## e. CI\_TD\_DRLKEY

```

CREATE TABLE CM_XT037_P2017JAN_ARC PARALLEL NOLOGGING
TABLESPACE CM_XT039_P2017JAN_ARC
AS
(
SELECT /*+ PARALLEL */ * FROM CISADM.CI_TD_DRLKEY PARTITION
(P2017JAN_S01)
UNION ALL
SELECT /*+ PARALLEL */ * FROM CISADM.CI_TD_DRLKEY PARTITION
(P2017JAN_S02)
UNION ALL
.
.
.
UNION ALL
SELECT /*+ PARALLEL */ * FROM CISADM.CI_TD_DRLKEY PARTITION
(P2017JAN_S08)
);
ALTER TABLE CM_XT037_P2017JAN_ARC NOPARALLEL LOGGING;

```

## 5. Create staging table and load data for parent table.

```

CREATE TABLE CM_XT039_P2017JAN_ARC NOLOGGING PARALLEL
TABLESPACE CM_XT039_P2017JAN_ARC AS
SELECT /*+ PARALLEL */ * FROM CISADM.CI_TD_ENTRY PARTITION
(P2017JAN);

ALTER TABLE CM_XT039_P2017JAN_ARC NOPARALLEL LOGGING;

```

## 6. Export tablespace using 'TRANSPORT\_TABLESPACES' method.

```

ALTER TABLESPACE CM_XT039_P2017JAN_ARC READ ONLY;

expdp system/manager DIRECTORY=DUMP_DIR DUMPFILE=
CM_XT039_P2017JAN_ARC.DMP TRANSPORT_TABLESPACES =
CM_XT039_P2017JAN_ARC LOGFILE=EXP_CM_XT039_P2017JAN_ARC.LOG
TRANSPORT_FULL_CHECK=Y

```

Ensure tablespace datafile required for further import should be preserved.

```
<<Transport THE FILE to LOCAL DB DIRECTORY DUMP_DIR like connected to asmcmd and
copied the file from cp cm_xt039_p201701_tbs_ar.553.913864937 /tugbu_perf_02/
BACKUPS/test_verification/>>
```

- Drop the partition, partition tablespace and archive tablespace (as it is already exported).

```
ALTER TABLE CISADM.CI_TD_ENTRY DROP PARTITION P2017JAN UPDATE
INDEXES;
DROP TABLESPACE CM_XT039_P2017JAN INCLUDING CONTENTS AND
DATAFILES;
DROP TABLESPACE CM_XT039_P2017JAN_ARC INCLUDING CONTENTS AND
DATAFILES;
```

## Restore Partition

- Create separate tablespace to restore the partition.

```
CREATE BIGFILE TABLESPACE CM_XT039_P2017JAN DATAFILE '+DATA'
SIZE 50M AUTOEXTEND ON MAXSIZE UNLIMITED DEFAULT ROW STORE
COMPRESS ADVANCED;
```

- Add partition using split operation on next greater value partition

```
ALTER TABLE CISADM.CI_TD_ENTRY SPLIT PARTITION P2017FEB AT
(TO_DATE('2017-02-01 00:00:01','SYYYY-MM-DD HH24:MI:SS'))
INTO
(
PARTITION P2017JAN TABLESPACE CM_XT039_P2017JAN , PARTITION
P2017FEB
)
UPDATE INDEXES;
```

In case table contains LOBS like F1\_SYNC\_REQ\_IN, there will be additional statement in split partition DDL indicating tablespace on which LOB should go.

```
ALTER TABLE CISADM.F1_SYNC_REQ_IN SPLIT PARTITION P2017FEB AT
(TO_DATE('2017-02-01 00:00:01','SYYYY-MM-DD HH24:MI:SS'))
INTO
(
PARTITION P2017JAN TABLESPACE CM_F1T191_P2017JAN
LOB(BO_DATA_AREA,PRE_TRN_INIT_BO_DATA_AREA,PRE_TRN_FIN_BO_DATA_
AREA,POST_TRN_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE
IN ROW COMPRESS MEDIUM CACHE TABLESPACE
CM_F1T191_P2017JAN)
, PARTITION P2017FEB
)
UPDATE INDEXES;
```

- Enable advanced compression after SPLIT partition as it will disable the compression.

```
ALTER TABLE CISADM.CI_TD_SRTKEY ROW STORE COMPRESS ADVANCED;
ALTER TABLE CISADM.CI_TD_MSG_PARM ROW STORE COMPRESS ADVANCED;
ALTER TABLE CISADM.CI_TD_DRLKEY ROW STORE COMPRESS ADVANCED;
ALTER TABLE CISADM.CI_TD_ENTRY_CHA ROW STORE COMPRESS ADVANCED;
ALTER TABLE CISADM.CI_TD_LOG ROW STORE COMPRESS ADVANCED;
```

- Import tablespace using TRANSPORT\_TABLESPACES method.



```
impdp system/manager DIRECTORY=DUMP_DIR DUMPFILE=
CM_XT039_P2017JAN_ARC.DMP PARTITION_OPTIONS=DEPARTITION
LOGFILE=IMP_CM_XT039_P2017JAN_ARC.LOG TRANSPORT_DATAFILES=/
tugbu_perf_02/BACKUPS/test_verification/
cm_xt039_p201701jan_ar.553.913864937
```

5. Load data into parent table first from the staging table

```
ALTER SESSION ENABLE PARALLEL DML;
```

```
INSERT /*+ APPEND PARALLEL */ INTO CISADM.CI_TD_ENTRY SELECT /
/*+ PARALLEL */ * FROM CM_XT039_P2017JAN_ARC;
COMMIT;
```

6. Load data into child table from the staging table

For each Child IN LIST OF CHILD TABLES, perform the following:

```
INSERT /*+ APPEND PARALLEL */ INTO CISADM.CI_TD_ENTRY_CHA
SELECT /*+ PARALLEL */ * FROM CM_XT701_P2017JAN_ARC;
COMMIT;
INSERT /*+ APPEND PARALLEL */ INTO CISADM.CI_TD_MSG_PARM
SELECT /*+ PARALLEL */ * FROM CM_XT04_P2017JAN_ARC;
COMMIT;
```

```
INSERT /*+ APPEND PARALLEL */ INTO CISADM.CI_TD_LOG SELECT /*+
PARALLEL */ * FROM CM_XT721_P2017JAN_ARC;
COMMIT;
```

```
INSERT /*+ APPEND PARALLEL */ INTO CISADM.CI_TD_SRTKEY SELECT
/*+ PARALLEL */ * FROM CM_XT041_P2017JAN_ARC;
COMMIT;
```

```
INSERT /*+ APPEND PARALLEL */ INTO CISADM.CI_TD_DRLKEY SELECT
/*+ PARALLEL */ * FROM CM_XT037_P2017JAN_ARC;
COMMIT;
```

7. Drop the archive tablespace after import is import and data loading is successful.

```
DROP TABLESPACE CM_XT039_P2017JAN_ARC INCLUDING CONTENTS AND
DATAFILES;
```

# Appendix E

---

## Partitioning and Compression Recommendations

This section specifies the partitioning and compression strategies recommended for an initial Oracle Utilities Smart Grid Gateway database configuration. It includes the following topics:

- [Partitioning Recommendations](#)
- [Compression Recommendations](#)

**Note:** If Information Lifecycle Management is part of your implementation, refer to **Appendix 6: Information Lifecycle Management and Data Archiving in SGG** in this guide for instructions on partitioning objects when using ILM.

# Partitioning Recommendations

In general, the recommendation is for a minimum of 'n' partitions for selective database objects, where 'n' is number of RAC nodes. The specific table level partitioning recommendations are as follows:

- The Table Partitioning scheme for Transaction tables is focused primarily on tables associated with Measurement MO, Measurement Log MO and Initial-Measurement-Data MO.
- D1\_MSRMT, D1\_MSRMT\_CHAR, D1\_MSRMT\_LOG, D1\_MSRMT\_LOG\_PARM tables can be partitioned by MSRMT\_DTTM. Bi-monthly partitions is a good start. Subpartition these tables by MEASR\_COMP\_ID (8 subpartitions should be a good number to start with).
- D1\_INIT\_MSRMT\_DATA table can be partitioned by D1\_TO\_DTTM. Bi-monthly partitions is a good start. Subpartition D1\_INIT\_MSRMT\_DATA table by MEASR\_COMP\_ID (8 subpartitions should be a good number to start with).
- D1\_INIT\_MSRMT\_DATA\_CHAR, D1\_INIT\_MSRMT\_DATA\_LOG, D1\_INIT\_MSRMT\_DATA\_LOG\_PARM tables are reference partitioned to the parent table.
- D1\_INIT\_MSRMT\_DATA\_K table can be partitioned by INIT\_MSRMT\_DATA\_ID (8 sub partitions should be a good number to start with).

The following sections gives partition recommendation and can be used as reference. Create one tablespace per partition as needed. It includes the following:

- [D1\\_MSRMT](#)
- [D1\\_MSRMT\\_CHAR](#)
- [D1\\_MSRMT\\_LOG](#)
- [D1\\_MSRMT\\_LOG\\_PARM](#)
- [D1\\_INIT\\_MSRMT\\_DATA](#)
- [D1\\_INIT\\_MSRMT\\_DATA\\_CHAR](#)
- [D1\\_INIT\\_MSRMT\\_DATA\\_K](#)
- [D1\\_INIT\\_MSRMT\\_DATA\\_LOG](#)
- [D1\\_INIT\\_MSRMT\\_DATA\\_LOG\\_PARM](#)

## D1\_MSRMT

```
CREATE BIGFILE TABLESPACE CM_D1T298_P2011JAN DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
CREATE BIGFILE TABLESPACE CM_D1T298_P2011MAR DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
CREATE BIGFILE TABLESPACE CM_D1T298_P2011MAY DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
CREATE BIGFILE TABLESPACE CM_D1T298_P2011JUL DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
CREATE BIGFILE TABLESPACE CM_D1T298_P2011SEP DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
```

```

CREATE BIGFILE TABLESPACE CM_D1T298_P2011NOV DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
CREATE BIGFILE TABLESPACE CM_D1T298_PMAX DATAFILE '+DATADG' SIZE 50M
AUTOEXTEND ON MAXSIZE UNLIMITED ;

CREATE TABLE D1_MSRMT (

MEASR_COMP_ID CHAR(12) NOT NULL ENABLE, MSRMT_DTTM DATE NOT NULL
ENABLE,
BO_STATUS_CD CHAR(12) DEFAULT ' ' NOT NULL ENABLE, MSRMT_COND_FLG
CHAR(6 BYTE) DEFAULT ' ' NOT NULL ENABLE, MSRMT_USE_FLG CHAR(4)
DEFAULT ' ' NOT NULL ENABLE, MSRMT_LOCAL_DTTM DATE NOT NULL ENABLE,
MSRMT_VAL NUMBER(16,6) DEFAULT 0 NOT NULL ENABLE,
ORIG_INIT_MSRMT_ID CHAR(14) DEFAULT ' ' NOT NULL ENABLE,
PREV_MSRMT_DTTM DATE,
MSRMT_VAL1 NUMBER(16,6) DEFAULT 0 NOT NULL ENABLE, MSRMT_VAL2
NUMBER(16,6) DEFAULT 0 NOT NULL ENABLE, MSRMT_VAL3 NUMBER(16,6)
DEFAULT 0 NOT NULL ENABLE, MSRMT_VAL4 NUMBER(16,6) DEFAULT 0 NOT NULL
ENABLE, MSRMT_VAL5 NUMBER(16,6) DEFAULT 0 NOT NULL ENABLE, MSRMT_VAL6
NUMBER(16,6) DEFAULT 0 NOT NULL ENABLE, MSRMT_VAL7 NUMBER(16,6)
DEFAULT 0 NOT NULL ENABLE, MSRMT_VAL8 NUMBER(16,6) DEFAULT 0 NOT NULL
ENABLE, MSRMT_VAL9 NUMBER(16,6) DEFAULT 0 NOT NULL ENABLE,
MSRMT_VAL10 NUMBER(16,6) DEFAULT 0 NOT NULL ENABLE, BUS_OBJ_CD
CHAR(30) DEFAULT ' ' NOT NULL ENABLE, CRE_DTTM DATE NOT NULL ENABLE,
STATUS_UPD_DTTM DATE NOT NULL ENABLE,
USER_EDITED_FLG CHAR(4) DEFAULT ' ' NOT NULL ENABLE, VERSION
NUMBER(5,0) DEFAULT 1 NOT NULL ENABLE,
LAST_UPDATE_DTTM DATE, READING_VAL NUMBER(16,6), COMBINED_MULTIPLIER
NUMBER(18,6), READING_COND_FLG CHAR(6)
) ENABLE ROW MOVEMENT
PARTITION BY RANGE (MSRMT_DTTM) SUBPARTITION BY range (MEASR_COMP_ID)
SUBPARTITION TEMPLATE (
subpartition S01 values less than (124999999999),
subpartition S02 values less than (249999999999),
subpartition S03 values less than (374999999999),
subpartition S04 values less than (499999999999),
subpartition S05 values less than (624999999999),
subpartition S06 values less than (744999999999),
subpartition S07 values less than (874999999999),
subpartition SMAX values less than (maxvalue)
)
(
PARTITION "P2011JAN" VALUES LESS THAN (TO_DATE('2011-02-01 00:00:01',
'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLESPACE CM_D1T298_P2011JAN,
PARTITION "P2011MAR" VALUES LESS THAN (TO_DATE('2011-04-01 00:00:01',
'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLESPACE CM_D1T298_P2011MAR,
PARTITION "P2011MAY" VALUES LESS THAN (TO_DATE('2011-06-01 00:00:01',
'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLESPACE CM_D1T298_P2011MAY,
PARTITION "P2011JUL" VALUES LESS THAN (TO_DATE('2011-08-01 00:00:01',
'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLESPACE CM_D1T298_P2011JUL,
PARTITION "P2011SEP" VALUES LESS THAN (TO_DATE('2011-10-01 00:00:01',
'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLESPACE CM_D1T298_P2011SEP,
PARTITION "P2011NOV" VALUES LESS THAN (TO_DATE('2011-12-01 00:00:01',
'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLESPACE CM_D1T298_P2011NOV,
PARTITION "PMAX" VALUES LESS THAN (MAXVALUE)
TABLESPACE CM_D1T298_PMAX

```

```
);

CREATE UNIQUE INDEX D1T298P0 ON D1_MSRMT(MEASR_COMP_ID, MSRMT_DTTM)
LOCAL COMPRESS ADVANCED LOW;

ALTER TABLE D1_MSRMT ADD CONSTRAINT D1T298P0 PRIMARY
KEY(MEASR_COMP_ID, MSRMT_DTTM) USING INDEX;
```

## D1\_MSRMT\_CHAR

```
CREATE BIGFILE TABLESPACE CM_D1T299_P2011JAN DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
CREATE BIGFILE TABLESPACE CM_D1T299_P2011MAR DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
CREATE BIGFILE TABLESPACE CM_D1T299_P2011MAY DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
CREATE BIGFILE TABLESPACE CM_D1T299_P2011JUL DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
CREATE BIGFILE TABLESPACE CM_D1T299_P2011SEP DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
CREATE BIGFILE TABLESPACE CM_D1T299_P2011NOV DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
CREATE BIGFILE TABLESPACE CM_D1T299_PMAX DATAFILE '+DATADG' SIZE 50M
AUTOEXTEND ON MAXSIZE UNLIMITED ;

CREATE TABLE D1_MSRMT_CHAR (
MEASR_COMP_ID CHAR(12) NOT NULL ENABLE, MSRMT_DTTM DATE NOT NULL
ENABLE,
CHAR_TYPE_CD CHAR(8) NOT NULL ENABLE, SEQ_NUM NUMBER(3,0) NOT
NULL ENABLE,
CHAR_VAL CHAR(16) DEFAULT ' ' NOT NULL ENABLE, ADHOC_CHAR_VAL
VARCHAR2(254) DEFAULT ' ' NOT NULL ENABLE, CHAR_VAL_FK1 VARCHAR2(50)
DEFAULT ' ' NOT NULL ENABLE, CHAR_VAL_FK2 VARCHAR2(50) DEFAULT ' '
NOT NULL ENABLE, CHAR_VAL_FK3 VARCHAR2(50) DEFAULT ' ' NOT NULL
ENABLE, CHAR_VAL_FK4 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK5 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE, SRCH_CHAR_VAL
VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE, VERSION NUMBER(5,0)
DEFAULT 1 NOT NULL ENABLE,
LAST_UPDATE_DTTM DATE, READING_VAL NUMBER(16,6), COMBINED_MULTIPLIER
NUMBER(18,6), READING_COND_FLG CHAR(6)
) ENABLE ROW MOVEMENT
PARTITION BY RANGE (MSRMT_DTTM) SUBPARTITION BY range (MEASR_COMP_ID)
SUBPARTITION TEMPLATE (
subpartition S01 values less than (124999999999),
subpartition S02 values less than (249999999999),
subpartition S03 values less than (374999999999),
subpartition S04 values less than (499999999999),
subpartition S05 values less than (624999999999),
subpartition S06 values less than (744999999999),
subpartition S07 values less than (874999999999),
subpartition SMAX values less than (maxvalue)
)
(
PARTITION "P2011JAN" VALUES LESS THAN (TO_DATE('2011-02-01 00:00:01',
'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLESPACE CM_D1T299_P2011JAN,
PARTITION "P2011MAR" VALUES LESS THAN (TO_DATE('2011-04-01 00:00:01',
'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLESPACE CM_D1T299_P2011MAR,
```

```

PARTITION "P2011MAY" VALUES LESS THAN (TO_DATE('2011-06-01 00:00:01',
'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLESPACE CM_D1T299_P2011MAY,
PARTITION "P2011JUL" VALUES LESS THAN (TO_DATE('2011-08-01 00:00:01',
'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLESPACE CM_D1T299_P2011JUL,
PARTITION "P2011SEP" VALUES LESS THAN (TO_DATE('2011-10-01 00:00:01',
'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLESPACE CM_D1T299_P2011SEP,
PARTITION "P2011NOV" VALUES LESS THAN (TO_DATE('2011-12-01 00:00:01',
'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLESPACE CM_D1T299_P2011NOV,
PARTITION "PMAX" VALUES LESS THAN (MAXVALUE)
TABLESPACE CM_D1T299_PMAX
);

CREATE BIGFILE TABLESPACE CM_D1T299_IND DATAFILE '+DATADG' SIZE 50M
AUTOEXTEND ON MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;

CREATE UNIQUE INDEX D1T299P0 ON D1_MSRMT_CHAR (
MEASR_COMP_ID, MSRMT_DTTM, CHAR_TYPE_CD, SEQ_NUM
) LOCAL COMPRESS ADVANCED LOW;

ALTER TABLE D1_MSRMT_CHAR ADD CONSTRAINT D1T299P0 PRIMARY KEY
(MEASR_COMP_ID, MSRMT_DTTM, CHAR_TYPE_CD, SEQ_NUM) USING INDEX ;

CREATE INDEX D1T299S1 ON D1_MSRMT_CHAR(SRCH_CHAR_VAL)
GLOBAL PARTITION BY HASH(SRCH_CHAR_VAL)
(
PARTITION P1 TABLESPACE CM_D1T299_IND,
PARTITION P2 TABLESPACE CM_D1T299_IND,
PARTITION P3 TABLESPACE CM_D1T299_IND,
PARTITION P4 TABLESPACE CM_D1T299_IND,
PARTITION P5 TABLESPACE CM_D1T299_IND,
PARTITION P6 TABLESPACE CM_D1T299_IND,
PARTITION P7 TABLESPACE CM_D1T299_IND,
PARTITION P8 TABLESPACE CM_D1T299_IND
)
TABLESPACE CM_D1T304_IND;

```

## D1\_MSRMT\_LOG

```

CREATE BIGFILE TABLESPACE CM_D1T300_P2011JAN DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
CREATE BIGFILE TABLESPACE CM_D1T300_P2011MAR DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
CREATE BIGFILE TABLESPACE CM_D1T300_P2011MAY DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
CREATE BIGFILE TABLESPACE CM_D1T300_P2011JUL DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
CREATE BIGFILE TABLESPACE CM_D1T300_P2011SEP DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
CREATE BIGFILE TABLESPACE CM_D1T300_P2011NOV DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
CREATE BIGFILE TABLESPACE CM_D1T300_PMAX DATAFILE '+DATADG' SIZE 50M
AUTOEXTEND ON MAXSIZE UNLIMITED ;

CREATE TABLE D1_MSRMT_LOG (
MEASR_COMP_ID CHAR(12), MSRMT_DTTM DATE,
SEQNO NUMBER(5,0),

```

```

ORIG_INIT_MSRMT_ID CHAR(14) DEFAULT ' ' NOT NULL ENABLE, BUS_OBJ_CD
CHAR(30) DEFAULT ' ' NOT NULL ENABLE,
CHAR_TYPE_CD CHAR(8) DEFAULT ' ' NOT NULL ENABLE, CHAR_VAL
CHAR(16) DEFAULT ' ' NOT NULL ENABLE, ADHOC_CHAR_VAL VARCHAR2(254)
DEFAULT ' ' NOT NULL ENABLE, CHAR_VAL_FK1 VARCHAR2(50) DEFAULT ' '
NOT NULL ENABLE, CHAR_VAL_FK2 VARCHAR2(50) DEFAULT ' ' NOT NULL
ENABLE, CHAR_VAL_FK3 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK4 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE, CHAR_VAL_FK5
VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE, DESCRLONG VARCHAR2(4000)
DEFAULT ' ' NOT NULL ENABLE, LOG_DTTM DATE NOT NULL ENABLE,
MESSAGE_CAT_NBR NUMBER(5,0) DEFAULT 0 NOT NULL ENABLE,
MESSAGE_NBR NUMBER(5,0) DEFAULT 0 NOT NULL ENABLE,
USER_ID CHAR(8) DEFAULT ' ' NOT
NULL ENABLE,
VERSION NUMBER(5,0) DEFAULT 1 NOT NULL ENABLE,
MSRMT_LOG_ENTRY_TYPE_FLG CHAR(4) DEFAULT ' ' NOT NULL ENABLE,
BO_DATA_AREA CLOB
)
LOB (BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW
COMPRESS MEDIUM CACHE)
ENABLE ROW MOVEMENT
PARTITION BY RANGE (MSRMT_DTTM) SUBPARTITION BY range (MEASR_COMP_ID)
SUBPARTITION TEMPLATE (
subpartition S01 values less than (124999999999),
subpartition S02 values less than (249999999999),
subpartition S03 values less than (374999999999),
subpartition S04 values less than (499999999999),
subpartition S05 values less than (624999999999),
subpartition S06 values less than (744999999999),
subpartition S07 values less than (874999999999),
subpartition SMAX values less than (maxvalue)
)
(
PARTITION "P2011JAN" VALUES LESS THAN (TO_DATE('2011-02-01 00:00:01',
'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
LOB (BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS
MEDIUM CACHE TABLESPACE CM_D1T300_P2011JAN)
TABLESPACE CM_D1T300_P2011JAN,
PARTITION "P2011MAR" VALUES LESS THAN (TO_DATE('2011-04-01 00:00:01',
'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
LOB (BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS
MEDIUM CACHE TABLESPACE CM_D1T300_P2011MAR)
TABLESPACE CM_D1T300_P2011MAR,
PARTITION "P2011MAY" VALUES LESS THAN (TO_DATE('2011-06-01 00:00:01',
'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
LOB (BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS
MEDIUM CACHE TABLESPACE CM_D1T300_P2011MAY)
TABLESPACE CM_D1T300_P2011MAY,
PARTITION "P2011JUL" VALUES LESS THAN (TO_DATE('2011-08-01 00:00:01',
'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
LOB (BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS
MEDIUM CACHE TABLESPACE CM_D1T300_P2011JUL)
TABLESPACE CM_D1T300_P2011JUL,
PARTITION "P2011SEP" VALUES LESS THAN (TO_DATE('2011-10-01 00:00:01',
'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
LOB (BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS
MEDIUM CACHE TABLESPACE CM_D1T300_P2011SEP)
TABLESPACE CM_D1T300_P2011SEP,
PARTITION "P2011NOV" VALUES LESS THAN (TO_DATE('2011-12-01 00:00:01',
'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
LOB (BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS
MEDIUM CACHE TABLESPACE CM_D1T300_P2011NOV)

```

```

TABLESPACE CM_D1T300_P2011NOV,
PARTITION "PMAX" VALUES LESS THAN (MAXVALUE)
LOB (BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW COMPRESS
MEDIUM CACHE TABLESPACE CM_D1T300_PMAX)
TABLESPACE CM_D1T300_PMAX
);

CREATE UNIQUE INDEX D1T300P0 ON D1_MSRMT_LOG (
MEASR_COMP_ID, MSRMT_DTTM, SEQNO
) LOCAL COMPRESS ADVANCED LOW;

ALTER TABLE D1_MSRMT_LOG ADD CONSTRAINT D1T300P0 PRIMARY KEY
(MEASR_COMP_ID, MSRMT_DTTM, SEQNO) USING INDEX ;

```

## D1\_MSRMT\_LOG\_PARM

```

CREATE BIGFILE TABLESPACE CM_D1T301_P2011JAN DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
CREATE BIGFILE TABLESPACE CM_D1T301_P2011MAR DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
CREATE BIGFILE TABLESPACE CM_D1T301_P2011MAY DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
CREATE BIGFILE TABLESPACE CM_D1T301_P2011JUL DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
CREATE BIGFILE TABLESPACE CM_D1T301_P2011SEP DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
CREATE BIGFILE TABLESPACE CM_D1T301_P2011NOV DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
CREATE BIGFILE TABLESPACE CM_D1T301_PMAX DATAFILE '+DATADG' SIZE 50M
AUTOEXTEND ON MAXSIZE UNLIMITED ;

CREATE TABLE D1_MSRMT_LOG_PARM (
MEASR_COMP_ID CHAR(12), MSRMT_DTTM DATE,
SEQNO NUMBER(5,0), PARM_SEQ NUMBER(3,0),
MSG_PARM_VAL VARCHAR2(30) DEFAULT ' ' NOT NULL ENABLE,
MSG_PARM_TYP_FLG CHAR(4) DEFAULT ' ' NOT NULL ENABLE, VERSION
NUMBER(5,0) DEFAULT 1 NOT NULL ENABLE
)
ENABLE ROW MOVEMENT
PARTITION BY RANGE (MSRMT_DTTM) SUBPARTITION BY range (MEASR_COMP_ID)
SUBPARTITION TEMPLATE (
subpartition S01 values less than (124999999999),
subpartition S02 values less than (249999999999),
subpartition S03 values less than (374999999999),
subpartition S04 values less than (499999999999),
subpartition S05 values less than (624999999999),
subpartition S06 values less than (744999999999),
subpartition S07 values less than (874999999999),
subpartition SMAX values less than (maxvalue)
)
(
PARTITION "P2011JAN" VALUES LESS THAN (TO_DATE('2011-02-01 00:00:01',
'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLESPACE CM_D1T301_P2011JAN,
PARTITION "P2011MAR" VALUES LESS THAN (TO_DATE('2011-04-01 00:00:01',
'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLESPACE CM_D1T301_P2011MAR,
PARTITION "P2011MAY" VALUES LESS THAN (TO_DATE('2011-06-01 00:00:01',
'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))

```



```

TABLESPACE CM_D1T301_P2011MAY,
PARTITION "P2011JUL" VALUES LESS THAN (TO_DATE('2011-08-01 00:00:01',
'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLESPACE CM_D1T301_P2011JUL,
PARTITION "P2011SEP" VALUES LESS THAN (TO_DATE('2011-10-01 00:00:01',
'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLESPACE CM_D1T301_P2011SEP,
PARTITION "P2011NOV" VALUES LESS THAN (TO_DATE('2011-12-01 00:00:01',
'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
TABLESPACE CM_D1T301_P2011NOV,
PARTITION "PMAx" VALUES LESS THAN (MAXVALUE)
TABLESPACE CM_D1T301_PMAx
);
CREATE UNIQUE INDEX D1T301P0 ON D1_MSRMT_LOG_PARM (
MEASR_COMP_ID, MSRMT_DTTM, SEQNO, PARM_SEQ
) INDEX LOCAL COMPRESS ADVANCED LOW;

ALTER TABLE D1_MSRMT_LOG_PARM ADD CONSTRAINT D1T301P0 PRIMARY KEY
(MEASR_COMP_ID, MSRMT_DTTM, SEQNO, PARM_SEQ) USING INDEX;

```

## D1\_INIT\_MSRMT\_DATA

```

CREATE BIGFILE TABLESPACE CM_D1T304_P2011JAN DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011MAR DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011MAY DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011JUL DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011SEP DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
CREATE BIGFILE TABLESPACE CM_D1T304_P2011NOV DATAFILE '+DATADG' SIZE
50M AUTOEXTEND ON MAXSIZE UNLIMITED ;
CREATE BIGFILE TABLESPACE CM_D1T304_PMAx DATAFILE '+DATADG' SIZE 50M
AUTOEXTEND ON MAXSIZE UNLIMITED ;

CREATE TABLE D1_INIT_MSRMT_DATA
(
INIT_MSRMT_DATA_ID CHAR(14) NOT NULL ENABLE,
MEASR_COMP_ID CHAR(12) DEFAULT ' ' NOT NULL ENABLE,
D1_FROM_DTTM DATE,
D1_TO_DTTM DATE,
DATA_SRC_FLG CHAR(4) DEFAULT ' ' NOT NULL ENABLE,
TIME_ZONE_CD CHAR(10) DEFAULT ' ' NOT NULL ENABLE,
BUS_OBJ_CD CHAR(30) DEFAULT ' ' NOT NULL ENABLE,
BO_STATUS_CD CHAR(12) DEFAULT ' ' NOT NULL ENABLE,
BO_STATUS_REASON_CD VARCHAR2(30) DEFAULT ' ' NOT NULL ENABLE,
IMD_BO_DATA_AREA CLOB,
STATUS_UPD_DTTM DATE NOT NULL ENABLE,
CRE_DTTM DATE NOT NULL ENABLE,
VERSION NUMBER(5,0) DEFAULT 1 NOT NULL ENABLE,
IMD_EXT_ID VARCHAR2(120),
PREVEE_BO_DATA_AREA CLOB,
POSTVEE_BO_DATA_AREA CLOB,
TRACE_BO_DATA_AREA CLOB,
RAW_BO_DATA_AREA CLOB,
LAST_UPDATE_DTTM DATE,
ILM_DT DATE,
ILM_ARCH_SW CHAR(1),

```

```

 RETENTION_PERIOD NUMBER(5,0) DEFAULT 99999 NOT NULL ENABLE
)
 ENABLE ROW MOVEMENT
 LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW
 COMPRESS MEDIUM CACHE)
 LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW
 COMPRESS MEDIUM CACHE)
 LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW
 COMPRESS MEDIUM CACHE)
 LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW
 COMPRESS MEDIUM CACHE)
 LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (ENABLE STORAGE IN ROW
 COMPRESS MEDIUM CACHE)
 PARTITION BY RANGE (D1_TO_DTTM)
 SUBPARTITION BY range (MEASR_COMP_ID)
 SUBPARTITION TEMPLATE (
 SUBPARTITION S01 VALUES LESS THAN (124999999999),
 SUBPARTITION S02 VALUES LESS THAN (249999999999),
 SUBPARTITION S03 VALUES LESS THAN (374999999999),
 SUBPARTITION S04 VALUES LESS THAN (499999999999),
 SUBPARTITION S05 VALUES LESS THAN (624999999999),
 SUBPARTITION S06 VALUES LESS THAN (749999999999),
 SUBPARTITION S07 VALUES LESS THAN (874999999999),
 SUBPARTITION SMAX VALUES LESS THAN (MAXVALUE)
)
 (
 PARTITION "P2011JAN" VALUES LESS THAN (TO_DATE('2011-02-01 00:00:01',
 'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
 LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
 CM_D1T304_P2011JAN)
 LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
 CM_D1T304_P2011JAN)
 LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
 CM_D1T304_P2011JAN)
 LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
 CM_D1T304_P2011JAN)
 LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
 CM_D1T304_P2011JAN)
 TABLESPACE CM_D1T304_P2011JAN,
 PARTITION "P2011MAR" VALUES LESS THAN (TO_DATE('2011-04-01 00:00:01',
 'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
 LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
 CM_D1T304_P2011MAR)
 LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
 CM_D1T304_P2011MAR)
 LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
 CM_D1T304_P2011MAR)
 LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
 CM_D1T304_P2011MAR)
 LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
 CM_D1T304_P2011MAR)
 TABLESPACE CM_D1T304_P2011MAR,
 PARTITION "P2011MAY" VALUES LESS THAN (TO_DATE('2011-06-01 00:00:01',
 'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
 LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
 CM_D1T304_P2011MAY)
 LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
 CM_D1T304_P2011MAY)
 LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
 CM_D1T304_P2011MAY)
 LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
 CM_D1T304_P2011MAY)
)

```

```

 LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
CM_D1T304_P2011MAY)
TABLESPACE CM_D1T304_P2011MAY,
PARTITION "P2011JUL" VALUES LESS THAN (TO_DATE('2011-08-01 00:00:01',
'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
 LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
CM_D1T304_P2011JUL)
 LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
CM_D1T304_P2011JUL)
 LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
CM_D1T304_P2011JUL)
 LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
CM_D1T304_P2011JUL)
 LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
CM_D1T304_P2011JUL)
TABLESPACE CM_D1T304_P2011JUL,
PARTITION "P2011SEP" VALUES LESS THAN (TO_DATE('2011-10-01 00:00:01',
'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
 LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
CM_D1T304_P2011SEP)
 LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
CM_D1T304_P2011SEP)
 LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
CM_D1T304_P2011SEP)
 LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
CM_D1T304_P2011SEP)
 LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
CM_D1T304_P2011SEP)
TABLESPACE CM_D1T304_P2011SEP,
PARTITION "P2011NOV" VALUES LESS THAN (TO_DATE('2011-12-01 00:00:01',
'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
 LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
CM_D1T304_P2011NOV)
 LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
CM_D1T304_P2011NOV)
 LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
CM_D1T304_P2011NOV)
 LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
CM_D1T304_P2011NOV)
 LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
CM_D1T304_P2011NOV)
TABLESPACE CM_D1T304_P2011NOV,
PARTITION "PMAX" VALUES LESS THAN (MAXVALUE)
 LOB (PREVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
CM_D1T304_PMAX)
 LOB (POSTVEE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
CM_D1T304_PMAX)
 LOB (TRACE_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
CM_D1T304_PMAX)
 LOB (RAW_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
CM_D1T304_PMAX)
 LOB (IMD_BO_DATA_AREA) STORE AS SECUREFILE (TABLESPACE
CM_D1T304_PMAX)
TABLESPACE CM_D1T304_PMAX
);

CREATE BIGFILE TABLESPACE CM_D1T304_IND DATAFILE '+DATA' SIZE 50M
AUTOEXTEND ON MAXSIZE UNLIMITED DEFAULT ROW STORE COMPRESS ADVANCED;

CREATE UNIQUE INDEX D1T304P0 ON D1_INIT_MSRMT_DATA (
INIT_MSRMT_DATA_ID
) TABLESPACE CM_D1T304_IND

```

```

GLOBAL PARTITION BY RANGE (INIT_MSRMT_DATA_ID)
(PARTITION P1 values less than (12499999999999),
PARTITION P2 values less than (24999999999999),
PARTITION P3 values less than (37499999999999),
PARTITION P4 values less than (49999999999999),
PARTITION P5 values less than (62499999999999),
PARTITION P6 values less than (74499999999999),
PARTITION P7 values less than (87499999999999),
PARTITION P8 values less than (maxvalue));

ALTER TABLE D1_INIT_MSRMT_DATA ADD CONSTRAINT D1T304P0 PRIMARY KEY
(INIT_MSRMT_DATA_ID) USING INDEX ;

CREATE INDEX D1T304S1 ON D1_INIT_MSRMT_DATA (MEASR_COMP_ID,
D1_TO_DTTM) TABLESPACE CM_D1T304_IND
GLOBAL PARTITION BY RANGE (MEASR_COMP_ID)
(PARTITION P1 VALUES LESS THAN ('124999999999'),
PARTITION P2 VALUES LESS THAN ('249999999999'),
PARTITION P3 VALUES LESS THAN ('374999999999'),
PARTITION P4 VALUES LESS THAN ('499999999999'),
PARTITION P5 VALUES LESS THAN ('624999999999'),
PARTITION P6 VALUES LESS THAN ('749999999999'),
PARTITION P7 VALUES LESS THAN ('874999999999'),
PARTITION P8 VALUES LESS THAN (MAXVALUE)
)
COMPRESS ADVANCED LOW;

```

## D1\_INIT\_MSRMT\_DATA\_CHAR

```

CREATE TABLE D1_INIT_MSRMT_DATA_CHAR
(
INIT_MSRMT_DATA_ID CHAR(14) NOT NULL ENABLE,
CHAR_TYPE_CD CHAR(8) NOT NULL ENABLE,
SEQ_NUM NUMBER(3,0) NOT NULL ENABLE,
CHAR_VAL CHAR(16) DEFAULT ' ' NOT NULL ENABLE,
ADHOC_CHAR_VAL VARCHAR2(254) DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK1 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK2 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK3 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK4 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK5 VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
SRCH_CHAR_VAL VARCHAR2(50) DEFAULT ' ' NOT NULL ENABLE,
VERSION NUMBER(5,0) DEFAULT 1 NOT NULL ENABLE,
LAST_UPDATE_DTTM DATE,
CONSTRAINT D1_INIT_MSRMT_DATA_CHAR_FK FOREIGN
KEY(INIT_MSRMT_DATA_ID) REFERENCES D1_INIT_MSRMT_DATA ON DELETE
CASCADE)
PARTITION BY REFERENCE (D1_INIT_MSRMT_DATA_CHAR_FK)
ENABLE ROW MOVEMENT;

CREATE UNIQUE INDEX D1T305P0 ON
D1_INIT_MSRMT_DATA_CHAR(INIT_MSRMT_DATA_ID, CHAR_TYPE_CD, SEQ_NUM)
TABLESPACE CM_D1T304_IND
GLOBAL PARTITION BY RANGE(INIT_MSRMT_DATA_ID)
(
PARTITION P1 VALUES LESS THAN ('12499999999999'),
PARTITION P2 VALUES LESS THAN ('24999999999999'),
PARTITION P3 VALUES LESS THAN ('37499999999999'),
PARTITION P4 VALUES LESS THAN ('49999999999999'),
PARTITION P5 VALUES LESS THAN ('62499999999999'),
PARTITION P6 VALUES LESS THAN ('74999999999999'),

```

```

PARTITION P7 VALUES LESS THAN ('8749999999999'),
PARTITION P8 VALUES LESS THAN (MAXVALUE)
) COMPRESS ADVANCED LOW;

```

```

ALTER TABLE D1_INIT_MSRMT_DATA_CHAR ADD CONSTRAINT D1T305P0 PRIMARY
KEY (INIT_MSRMT_DATA_ID, CHAR_TYPE_CD, SEQ_NUM) USING INDEX ;

```

```

CREATE INDEX D1T305S1 ON D1_INIT_MSRMT_DATA_CHAR(SRCH_CHAR_VAL)
GLOBAL PARTITION BY HASH(SRCH_CHAR_VAL)
(
PARTITION P1 TABLESPACE CM_D1T304_IND,
PARTITION P2 TABLESPACE CM_D1T304_IND,
PARTITION P3 TABLESPACE CM_D1T304_IND,
PARTITION P4 TABLESPACE CM_D1T304_IND,
PARTITION P5 TABLESPACE CM_D1T304_IND,
PARTITION P6 TABLESPACE CM_D1T304_IND,
PARTITION P7 TABLESPACE CM_D1T304_IND,
PARTITION P8 TABLESPACE CM_D1T304_IND
);

```

## D1\_INIT\_MSRMT\_DATA\_K

```

CREATE TABLE D1_INIT_MSRMT_DATA_K (
INIT_MSRMT_DATA_ID CHAR(14),
ENV_ID NUMBER(6,0) NOT NULL ENABLE,
CONSTRAINT D1T314P0 PRIMARY KEY (INIT_MSRMT_DATA_ID, ENV_ID) ENABLE
)
ORGANIZATION INDEX ENABLE ROW MOVEMENT
PARTITION BY RANGE (INIT_MSRMT_DATA_ID)
(PARTITION P1 values less than (12499999999999),
PARTITION P2 values less than (24999999999999),
PARTITION P3 values less than (37499999999999),
PARTITION P4 values less than (49999999999999),
PARTITION P5 values less than (62499999999999),
PARTITION P6 values less than (74499999999999),
PARTITION P7 values less than (87499999999999),
PARTITION P8 values less than (maxvalue))
TABLESPACE CM_D1T314_IND ;

```

## D1\_INIT\_MSRMT\_DATA\_LOG

```

CREATE TABLE D1_INIT_MSRMT_DATA_LOG
(
INIT_MSRMT_DATA_ID CHAR(14) NOT NULL ENABLE,
SEQNO NUMBER(5,0) NOT NULL ENABLE,
BO_STATUS_CD CHAR(12) DEFAULT ' ' NOT NULL ENABLE,
BO_STATUS_REASON_CD VARCHAR2(30 BYTE) DEFAULT ' ' NOT NULL ENABLE,
CHAR_TYPE_CD CHAR(8) DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL CHAR(16) DEFAULT ' ' NOT NULL ENABLE,
ADHOC_CHAR_VAL VARCHAR2(254 BYTE) DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK1 VARCHAR2(50 BYTE) DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK2 VARCHAR2(50 BYTE) DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK3 VARCHAR2(50 BYTE) DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK4 VARCHAR2(50 BYTE) DEFAULT ' ' NOT NULL ENABLE,
CHAR_VAL_FK5 VARCHAR2(50 BYTE) DEFAULT ' ' NOT NULL ENABLE,
DESCRLONG VARCHAR2(4000) DEFAULT ' ' NOT NULL ENABLE,
LOG_DTTM DATE NOT NULL ENABLE,
LOG_ENTRY_TYPE_FLG CHAR(4) DEFAULT ' ' NOT NULL ENABLE,

```

```

 MESSAGE_CAT_NBR NUMBER(5,0) DEFAULT 0 NOT NULL ENABLE,
 MESSAGE_NBR NUMBER(5,0) DEFAULT 0 NOT NULL ENABLE,
 USER_ID CHAR(8) DEFAULT ' ' NOT NULL ENABLE,
 VERSION NUMBER(5,0) DEFAULT 1 NOT NULL ENABLE,
 LAST_UPDATE_DTTM DATE,
 CONSTRAINT D1_INIT_MSRMT_DATA_LOG_FK FOREIGN
KEY(INIT_MSRMT_DATA_ID) REFERENCES D1_INIT_MSRMT_DATA ON DELETE
CASCADE)
PARTITION BY REFERENCE (D1_INIT_MSRMT_DATA_LOG_FK)
ENABLE ROW MOVEMENT;

CREATE UNIQUE INDEX D1T306P0 ON D1_INIT_MSRMT_DATA_LOG
(INIT_MSRMT_DATA_ID, SEQNO) TABLESPACE CM_D1T304_IND
GLOBAL PARTITION BY RANGE(INIT_MSRMT_DATA_ID)
(
PARTITION P1 VALUES LESS THAN ('12499999999999'),
PARTITION P2 VALUES LESS THAN ('24999999999999'),
PARTITION P3 VALUES LESS THAN ('37499999999999'),
PARTITION P4 VALUES LESS THAN ('49999999999999'),
PARTITION P5 VALUES LESS THAN ('62499999999999'),
PARTITION P6 VALUES LESS THAN ('74999999999999'),
PARTITION P7 VALUES LESS THAN ('87499999999999'),
PARTITION P8 VALUES LESS THAN (MAXVALUE)
) COMPRESS ADVANCED LOW;

ALTER TABLE D1_INIT_MSRMT_DATA_LOG ADD CONSTRAINT D1T306P0 PRIMARY KEY
(INIT_MSRMT_DATA_ID, SEQNO) USING INDEX ;

```

## D1\_INIT\_MSRMT\_DATA\_LOG\_PARM

```

CREATE TABLE D1_INIT_MSRMT_DATA_LOG_PARM
(
INIT_MSRMT_DATA_ID CHAR(14) NOT NULL ENABLE,
SEQNO NUMBER(5,0) NOT NULL ENABLE,
PARAM_SEQ NUMBER(3,0) NOT NULL ENABLE,
MSG_PARAM_VAL VARCHAR2(30) DEFAULT ' ' NOT NULL ENABLE,
MSG_PARAM_TYP_FLG CHAR(4) DEFAULT ' ' NOT NULL ENABLE,
VERSION NUMBER(5,0) DEFAULT 1 NOT NULL ENABLE,
LAST_UPDATE_DTTM DATE,
CONSTRAINT D1_INIT_MSRMT_DATA_LOG_PARM_FK FOREIGN
KEY(INIT_MSRMT_DATA_ID) REFERENCES D1_INIT_MSRMT_DATA ON DELETE
CASCADE)
PARTITION BY REFERENCE (D1_INIT_MSRMT_DATA_LOG_PARM_FK)
ENABLE ROW MOVEMENT;

CREATE UNIQUE INDEX D1T307P0 ON
D1_INIT_MSRMT_DATA_LOG_PARM(INIT_MSRMT_DATA_ID, SEQNO, PARAM_SEQ)
TABLESPACE CM_D1T304_IND
GLOBAL PARTITION BY RANGE(INIT_MSRMT_DATA_ID)
(
PARTITION P1 VALUES LESS THAN ('12499999999999'),
PARTITION P2 VALUES LESS THAN ('24999999999999'),
PARTITION P3 VALUES LESS THAN ('37499999999999'),
PARTITION P4 VALUES LESS THAN ('49999999999999'),
PARTITION P5 VALUES LESS THAN ('62499999999999'),
PARTITION P6 VALUES LESS THAN ('74999999999999'),
PARTITION P7 VALUES LESS THAN ('87499999999999'),
PARTITION P8 VALUES LESS THAN (MAXVALUE)
) COMPRESS ADVANCED LOW;

```

```
ALTER TABLE D1_INIT_MSRMT_DATA_LOG_PARM ADD CONSTRAINT D1T307P0
PRIMARY KEY (INIT_MSRMT_DATA_ID, SEQNO, PARM_SEQ) USING INDEX ;
```

## Compression Recommendations

It is highly recommended to use the following guidelines with regard to compression.

1. For all transactional data tables including ILM enabled tables (except D1\_MSRMT\* tables):
  - a. For easier operational manageability, it is recommended to enable the compression at tablespace level while creating separate tablespaces for each logical unit of archival (like a parent table partition and the corresponding referenced child table partitions).
  - b. Use securefile medium compression for LOBs.
  - c. On Oracle database 12c:
    - Use advanced compression for table data compression.
    - Compress indexes using advanced low compression (using 'compress advanced low' clause).
  - d. On Oracle database 11g:
    - Use OLTP compression for table data and compression indexes using default compression.
2. For D1\_MSRMT\* tables:
  - a. Keep current table partitions uncompressed for D1\_MSRMT. Other D1\_MSRMT\* tables should use compressed tablespaces for all partitions.
  - b. For the D1\_MSRMT table- Periodically (recommended monthly), compress the data by reloading into a staging table followed by partition exchange. It is highly recommended to use bulk load CTAS operation with parallel clause during the reload.
    - Use 'QUERY HIGH' compression for Exadata implementations.
    - For non-Exadata implementations, on 12c use 'row store compress advanced' and on 11g use OLTP compression.
  - c. For indexes
    - On Oracle database 12c, Compress indexes using advanced low compression (using 'compress advanced low' clause).
    - On Oracle database 11g, use default index compression.

# Appendix F

---

## Database Changes in Oracle Utilities Smart Grid Gateway

This chapter provides the database changes in specific releases of Oracle Utilities Smart Grid Gateway.

- [Upgrading from Oracle Utilities Smart Grid Gateway V2.3.0.2.0 to V2.4.0.0.0](#)
- [Upgrading from Oracle Utilities Smart Grid Gateway V2.3.0.2.3 to V2.4.0.0.0](#)



# Upgrading from Oracle Utilities Smart Grid Gateway V2.3.0.2.0 to V2.4.0.0.0

## New Tables

| Table_Name                     |
|--------------------------------|
| D1_ACTIVITY_TYPE_FLD_TASK_TYPE |
| D1_DVC_EQPMNT                  |
| D1_FACILITY_SP                 |
| D1_SP_TYPE_FIELD_TASK_TYPE     |

## New Views

| View_Name               |
|-------------------------|
| D1_ACT_DVC_SP_VW        |
| D1_BI_DYN_AGG_VW        |
| D2_MEASR_QTY_AGR_MV     |
| D2_MEASR_QTY_MV         |
| D2_QUALITY_CNT_AGR_MV   |
| D2_QUALITY_CNT_MV       |
| D2_TIMELINESS_CNT_MV    |
| D2_TIMELINESS_QTY_MV    |
| D2_TIMELINES_CNT_AGR_MV |
| D2_TIMELINES_QTY_AGR_MV |

## Primary Key Change

| Table_Name                    | Added_Column  |
|-------------------------------|---------------|
| D1_MEASR_COMP_SET_PARTICIPANT | MEASR_COMP_ID |

## Primary Key Change

| Table_Name                    | Deleted_Column |
|-------------------------------|----------------|
| D1_MEASR_COMP_SET_PARTICIPANT | SEQNO          |

**New Primary Keys**

| Table_Name | Primary Key | Column_Names |
|------------|-------------|--------------|
| D1_US_QTY  | D1M576P0    | D1_US_QTY_ID |

**Added Columns (Table)**

| Table_Name       | Column_Name                            |
|------------------|----------------------------------------|
| D1_ACTIVITY_TYPE | SP_DVC_COND_SRC_FLG CHAR(4)            |
| D1_CONS_EXT_TYPE | INTERVAL_SCALAR_FLG CHAR(4)            |
| D1_DVC           | ACCESS_GRP_CD CHAR(12)                 |
| D1_DVC_TYPE      | ATTACHED_TO_FLG CHAR(4)                |
| D1_MEASR_COMP    | ACCESS_GRP_CD CHAR(12)                 |
| D1_SP_EQPMNT     | EQPMNT_CNT NUMBER(11,2)                |
| D1_SP_EQPMNT     | BO_DATA_AREA CLOB                      |
| D1_MDSI          | VERSION NUMBER(5,0) DEFAULT 1 NOT NULL |
| D1_MDSS          | VERSION NUMBER(5,0) DEFAULT 1 NOT NULL |

**Updated Column Sequences**

| Table_Name                    | Old_Column_Sequence        | New_Column_Sequence  |
|-------------------------------|----------------------------|----------------------|
| D1_MEASR_COMP_SET_PARTICIPANT | 1. MEASR_COMP_ID           | 1. MEASR_COMP_SET_CD |
|                               | 2. MEASR_COMP_SET_CD       | 2. SEQNO             |
|                               | 3. INITIATED_BY_MC_SET_FLG | 3. BO_XML_DATA_AREA  |
|                               | 4. BO_XML_DATA_AREA        | 4. VERSION           |
|                               | 5. VERSION                 | 5. MEASR_COMP_ID     |
|                               | 6. SEQNO                   | 6. MC_ID_TYPE_FLG    |
|                               | 7. MC_ID_TYPE_FLG          | 7. ID_VALUE          |

**Updated Columns**

| Table_Name                    | Column_Name | From         | To            |
|-------------------------------|-------------|--------------|---------------|
| D1_CONTACT_EMAIL              | EMAIL_VALUE | VARCHAR2(70) | VARCHAR2(254) |
| D1_MEASR_COMP_SET_PARTICIPANT | SEQNO       | NOT NULL     | NULL          |

| Table_Name                    | Column_Name   | From | To       |
|-------------------------------|---------------|------|----------|
| D1_MEASR_COMP_SET_PARTICIPANT | MEASR_COMP_ID | NULL | NOT NULL |

**Added Columns (View)**

| Table_Name          | Column_Name |
|---------------------|-------------|
| D1_DVC_SP_VW        | DIVISION_CD |
| D1_INBOUND_COMM_VW  | DIVISION_CD |
| D1_OUTBOUND_COMM_VW | DIVISION_CD |

**Added Indexes**

| Table_Name                     | Index_Name |
|--------------------------------|------------|
| D1_ACTIVITY_TYPE_FLD_TASK_TYPE | D1C631P0   |
| D1_DVC_EQPMNT                  | D1M629P0   |
| D1_DVC_EQPMNT                  | D1M629S0   |
| D1_FACILITY_SP                 | D1T633P0   |
| D1_SP_TYPE_FIELD_TASK_TYPE     | D1M630P0   |
| D1_USAGE_EXCP                  | D1T443S1   |
| D1_US_QTY                      | D1M576P0   |

**Index Changes (Columns Added)**

| Table_Name                    | Index_Name | Added_Column  |
|-------------------------------|------------|---------------|
| D1_MEASR_COMP_SET_PARTICIPANT | D1C478P0   | MEASR_COMP_ID |

**Index Changes (Columns Deleted)**

| Table_Name            | Index_Name | Deleted_Column      |
|-----------------------|------------|---------------------|
| D1_CONTACT_IDENTIFIER | D1C183S0   | CONTACT_ID_TYPE_FLG |
| CONTACT_ID_TYPE_FLG   | D1C183S1   | CONTACT_ID_TYPE_FLG |
| D1_SP_IDENTIFIER      | D1M104S0   | SP_ID_TYPE_FLG      |
| D1_SP_IDENTIFIER      | D1M104S1   | SP_ID_TYPE_FLG      |
| D1_US_IDENTIFIER      | D1M213S0   | US_ID_TYPE_FLG      |

| Table_Name                    | Index_Name | Deleted_Column       |
|-------------------------------|------------|----------------------|
| D1_US_IDENTIFIER              | D1M213S1   | US_ID_TYPE_FLG       |
| D1_DVC_IDENTIFIER             | D1M236S0   | DVC_ID_TYPE_FLG      |
| D1_DVC_IDENTIFIER             | D1M236S1   | DVC_ID_TYPE_FLG      |
| D1_MEASR_COMP_IDENTIFIER      | D1M253S0   | MC_ID_TYPE_FLG       |
| D1_MEASR_COMP_IDENTIFIER      | D1M253S1   | MC_ID_TYPE_FLG       |
| D1_ACTIVITY_IDENTIFIER        | D1T330S0   | ACTIVITY_ID_TYPE_FLG |
| D1_ACTIVITY_IDENTIFIER        | D1T330S1   | ACTIVITY_ID_TYPE_FLG |
| D1_COMM_OUT_IDENTIFIER        | D1T385S0   | COMM_ID_TYPE_FLG     |
| D1_COMM_OUT_IDENTIFIER        | D1T385S1   | COMM_ID_TYPE_FLG     |
| D1_COMM_IN_IDENTIFIER         | D1T391S0   | COMM_ID_TYPE_FLG     |
| D1_COMM_IN_IDENTIFIER         | D1T391S1   | COMM_ID_TYPE_FLG     |
| D1_DVC_EVT_IDENTIFIER         | D1T405S0   | DVC_EVT_ID_TYPE_FLG  |
| D1_DVC_EVT_IDENTIFIER         | D1T405S1   | DVC_EVT_ID_TYPE_FLG  |
| D1_MEASR_COMP_SET_PARTICIPANT | D1C478P0   | SEQNO                |
| D1_INIT_MSRMT_DATA            | D1T304S1   | BO_STATUS_CD         |
| D1_INIT_MSRMT_DATA            | D1T304S1   | BUS_OBJ_CD           |
| D1_INIT_MSRMT_DATA            | D1T304S1   | D1_FROM_DTTM         |

### Index Organized Tables

The following tables are converted from normal table to Index Organized tables (only on-premises).

|                      |
|----------------------|
| D1_ATTR_VAL_K        |
| D1_ADS_K             |
| D1_INIT_MSRMT_SNAP_K |
| D1_MDSI_K            |
| D1_MDSS_K            |
| D1_MKT_AWARD_K       |
| D1_MKT_CONTRACT_K    |
| D1_MKT_PRODUCT_K     |
| D1_MKT_RP_K          |

D1\_SP\_QTY\_K

D1\_USAGE\_EXCP\_K

D1\_US\_MKT\_PART\_K

D1\_US\_QTY\_K

# Upgrading from Oracle Utilities Smart Grid Gateway V2.3.0.2.3 to V2.4.0.0.0

## New Tables

| Table_Name     |
|----------------|
| D1_FACILITY_SP |

## New Columns

| Table_Name     | Column_Name   | Data Type | Data Length |
|----------------|---------------|-----------|-------------|
| D1_DVC         | ACCESS_GRP_CD | CHAR      | 12          |
| D1_MEASR_COMP  | ACCESS_GRP_CD | CHAR      | 12          |
| D1_FACILITY_SP | FACILITY_ID   | CHAR      | 14          |
| D1_FACILITY_SP | D1_SP_ID      | CHAR      | 12          |
| D1_FACILITY_SP | VERSION       | NUMBER    | 5           |

## Updated Columns

| Table_Name                    | Old_Column_Sequence        | New_Column_Sequence        |
|-------------------------------|----------------------------|----------------------------|
| D1_MEASR_COMP_SET_PARTICIPANT | 1. MEASR_COMP_ID           | 1. MEASR_COMP_SET_CD       |
|                               | 2. MEASR_COMP_SET_CD       | 2. SEQNO                   |
|                               | 3. INITIATED_BY_MC_SET_FLG | 3. BO_XML_DATA_AREA        |
|                               | 4. BO_XML_DATA_AREA        | 4. VERSION                 |
|                               | 5. VERSION                 | 5. MEASR_COMP_ID           |
|                               | 6. SEQNO                   | 6. MC_ID_TYPE_FLG          |
|                               | 7. MC_ID_TYPE_FLG          | 7. ID_VALUE                |
|                               | 8. ID_VALUE                | 8. INITIATED_BY_MC_SET_FLG |

## New Primary Keys

| Table_Name     | Primary_Key | Columns                  |
|----------------|-------------|--------------------------|
| D1_FACILITY_SP | D1T633P0    | FACILITY_ID,<br>D1_SP_ID |

**Updated Primary Keys**

| Table_Name                    | Old Primary Key Column      | New Primary Key Column              |
|-------------------------------|-----------------------------|-------------------------------------|
| D1_MEASR_COMP_SET_PARTICIPANT | MEASR_COMP_SET_CD,<br>SEQNO | MEASR_COMP_ID,<br>MEASR_COMP_SET_CD |

**New Indexes**

| Table_Name     | Index_Name |
|----------------|------------|
| D1_FACILITY_SP | D1T633P0   |
| D1_USAGE_EXCP  | D1T443S1   |

**Index Changes**

| Table_Name                    | Index_Name | Old Column                                   | New Column          |
|-------------------------------|------------|----------------------------------------------|---------------------|
| D1_CONTACT_IDENTIFI<br>TIFIER | D1C183S0   | CONTACT_ID_TYPE_F<br>LG,<br>ID_VALUE         | ID_VALUE            |
| D1_CONTACT_IDENTIFI<br>TIFIER | D1C183S1   | CONTACT_ID_TYPE_F<br>LG,<br>UPPER(ID_VALUE)  | UPPER<br>(ID_VALUE) |
| D1_SP_IDENTIFIER              | D1M104S0   | SP_ID_TYPE_FLG,<br>ID_VALUE                  | ID_VALUE            |
| D1_SP_IDENTIFIER              | D1M104S1   | SP_ID_TYPE_FLG,<br>UPPER(ID_VALUE)           | UPPER(ID_VAL<br>UE) |
| D1_US_IDENTIFIER              | D1M213S0   | US_ID_TYPE_FLG,ID_<br>VALUE                  | ID_VALUE            |
| D1_US_IDENTIFIER              | D1M213S1   | US_ID_TYPE_FLG,<br>UPPER(ID_VALUE)           | UPPER(ID_VAL<br>UE) |
| D1_DVC_IDENTIFIER             | D1M236S0   | DVC_ID_TYPE_FLG,<br>ID_VAL UE                | ID_VALUE            |
| D1_DVC_IDENTIFIER             | D1M236S1   | DVC_ID_TYPE_FLG,<br>UPPER(ID_VALUE)          | UPPER(ID_VAL<br>UE) |
| D1_MEASR_COMP_<br>IDENTIFIER  | D1M253S0   | MC_ID_TYPE_FLG,<br>ID_VALUE                  | ID_VALUE            |
| D1_MEASR_COMP_<br>IDENTIFIER  | D1M253S1   | MC_ID_TYPE_FLG,<br>UPPER(ID_VALUE)           | UPPER(ID_VAL<br>UE) |
| D1_ACTIVITY_<br>IDENTIFIER    | D1T330S0   | ACTIVITY_ID_TYPE_<br>FLG,<br>ID_VALUE        | ID_VALUE            |
| D1_ACTIVITY_<br>IDENTIFIER    | D1T330S1   | ACTIVITY_ID_TYPE_<br>FLG,<br>UPPER(ID_VALUE) | UPPER(ID_VAL<br>UE) |

| Table_Name                    | Index_Name | Old Column                                                                    | New Column                          |
|-------------------------------|------------|-------------------------------------------------------------------------------|-------------------------------------|
| D1_COMM_OUT_IDENTIFIER        | D1T385S0   | COMM_ID_TYPE_FLG,<br>ID_VALUE                                                 | ID_VALUE                            |
| D1_COMM_OUT_IDENTIFIER        | D1T385S1   | COMM_ID_TYPE_FLG,<br>UPPER(ID_VALUE)                                          | UPPER(ID_VALUE)                     |
| D1_COMM_IN_IDENTIFIER         | D1T391S0   | COMM_ID_TYPE_FLG,<br>ID_VALUE                                                 | ID_VALUE                            |
| D1_COMM_IN_IDENTIFIER         | D1T391S1   | COMM_ID_TYPE_FLG,<br>UPPER(ID_VALUE)                                          | UPPER(ID_VALUE)                     |
| D1_DVC_EVT_IDENTIFIER         | D1T405S0   | DVC_EVT_ID_TYPE_<br>FLG,<br>ID_VALUE                                          | ID_VALUE                            |
| D1_DVC_EVT_IDENTIFIER         | D1T405S1   | DVC_EVT_ID_TYPE_<br>FLG,<br>UPPER(ID_VALUE)                                   | UPPER(ID_VALUE)                     |
| D1_MEASR_COMP_SET_PARTICIPANT | D1C478P0   | MEASR_COMP_SET_<br>CD,<br>SEQ NO                                              | MEASR_COMP_ID,<br>MEASR_COMP_SET_CD |
| D1_INIT_MSRMT_DATA*           | D1T304S1   | MEASR_COMP_ID,<br>BO_STATUS_CD,<br>BUS_OBJ_CD,<br>D1_TO_DTTM,<br>D1_FROM_DTTM | MEASR_COMP_ID,<br>D1_TO_DTTM        |

**\*Note that** within this upgrade, the Initial Measurement Data secondary index columns have been changed as outlined in the table.

Please be aware that implementing this change can take longer than expected depending on the amount of data present in the Initial Measurement Data table.

To manually create this index outside of the upgrade process, it is recommended to follow these steps:

1. CREATE INDEX ZZ\_D1T304S1 ON D1\_INIT\_MSRMT\_DATA (MEASR\_COMP\_ID,D1\_TO\_DTTM);

Include tablespace, compression, partitioning, and parallelism as required.

2. ALTER INDEX D1T304S1 RENAME TO ZZ\_D1T304S1\_ORIG;
3. ALTER INDEX ZZ\_D1T304S1 RENAME TO D1T304S1;
4. ALTER INDEX D1T304S1 NOPARALLEL;

(Optional: If parallelism was used to create the index)

5. DROP INDEX ZZ\_D1T304S1\_ORIG;

After the index is created with the new columns, the upgrade process will not recreate it.



### Updated Views

| View_Name          |
|--------------------|
| D1_DVC_SP_VW       |
| D1_INBOUND_COMM_VW |
| D1_INBOUND_COMM_VW |

# Appendix G

---

## 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.

The section provides information on upgrading the Oracle Utilities Application Framework Database including:

- [Upgrading from Oracle Utilities Application Framework v4.3.0.1.0 to v4.3.0.4.0](#)
- [Upgrading from Oracle Utilities Application Framework v4.3.0.2.0 to v4.3.0.4.0](#)
- [Upgrading from Oracle Utilities Application Framework v4.3.0.3.0 to v4.3.0.4.0](#)
- [Upgrading from Oracle Utilities Application Framework v4.3.0.4.0 to v4.3.0.5.0](#)
- [Upgrading from Oracle Utilities Application Framework v4.3.0.5.0 to v4.3.0.6.0](#)
- [Upgrading from Oracle Utilities Application Framework v4.3.0.6.0 to v4.4.0.0.0](#)
- [Upgrading from Oracle Utilities Application Framework v4.4.0.0.0 to v4.4.0.2.0](#)
- [Upgrading from Oracle Utilities Application Framework v4.4.0.2.0 to v4.4.0.3.0](#)

# Upgrading from Oracle Utilities Application Framework v4.3.0.1.0 to v4.3.0.4.0

## New Tables

| Table                    | Type of Table                         |
|--------------------------|---------------------------------------|
| F1_BUS_FLG               | Business Flag                         |
| F1_BUS_FLG_CHAR          | Business Flag Characteristic          |
| F1_BUS_FLG_K             | Business Flag Key Table               |
| F1_BUS_FLG_LOG           | Business Flag Log                     |
| F1_BUS_FLG_LOG_PARM      | Business Flag Log Parameter           |
| F1_BUS_FLG_REL           | Business Flag Relationship            |
| F1_BUS_FLG_REL_OBJ       | Business Flag Related Object          |
| F1_BUS_FLG_TYPE          | Business Flag Type                    |
| F1_BUS_FLG_TYPE_ALG      | Business Flag Type Algorithm          |
| F1_BUS_FLG_TYPE_BUS_PROC | Business Flag Type / Business Process |
| F1_BUS_FLG_TYPE_CHAR     | Business Flag Type Characteristic     |
| F1_BUS_FLG_TYPE_L        | Business Flag Type Language           |
| F1_ETL_MP_CTRL           | ETL Mapping Control                   |

## New Views

None

## Dropped Tables

None

## Unsupported Tables

None

## Added Columns

| Table                 | Column                | Required |
|-----------------------|-----------------------|----------|
| CI_BATCH_THD          | LOG_FILE_NAME         |          |
| CI_MD_TBL             | CHAR_ENTITY_FLG       | N        |
| F1_EXTSYS_OUTMSG_PROF | JSON_CONVRSN_METH_FLG | N        |
| F1_EXTSYS_OUTMSG_PROF | REQ_SCHEMA_NAME       | N        |
| F1_EXTSYS_OUTMSG_PROF | RES_SCHEMA_NAME       | N        |
| F1_MIGR_OBJ_ALG       | ALG_PROC_TYPE_FLG     | N        |

### Dropped Columns

| Table         | Column        |
|---------------|---------------|
| CI_MD_TBL     | TBL_USAGE_FLG |
| CI_MD_TBL_FLD | FLD_USAGE_FLG |

### Unsupported Table Columns

None

### Column Format Change

None

# Upgrading from Oracle Utilities Application Framework v4.3.0.2.0 to v4.3.0.4.0

## 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

# Upgrading from Oracle Utilities Application Framework v4.3.0.3.0 to 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_ENTTTY | COMMENT_LONG       | N        |
| F1_MIGR_REQ_INSTR_ENTTTY | 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_<br>VAL_CHAR | F1_EXT_LOOKUP_<br>VALUE | VARCHAR2<br>(30) | VARCHAR2<br>(254) |

**Primary Key Change**

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



# Upgrading from Oracle Utilities Application Framework v4.3.0.4.0 to v4.3.0.5.0

## New Tables

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

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

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

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

### New Views

None

### Dropped Tables

| Table                |
|----------------------|
| F1_IWS_ANN_CHAR      |
| F1_IWS_ANN_TYPE_CHAR |

### Unsupported Tables

None

### Added Columns

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

---

| Table            | Column    | Required |
|------------------|-----------|----------|
| F1_OUTMSG_TYPE_L | OWNER_FLG | N        |

---

**Dropped Columns**

None

**Unsupported Table Columns**

None

**Column Format Change**

None

**Primary Key Change**

None

**Index Changes**

Index S1C675S1 for table F1\_EXT\_LOOKUP\_VAL\_CHAR has been renamed to F1C675S1.

# Upgrading from Oracle Utilities Application Framework v4.3.0.5.0 to 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_CRYPT_KEY_RING_LINK |

### Added Columns

| Table        | Column   | Required |
|--------------|----------|----------|
| CI_BATCH_RUN | END_DTTM | N        |

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

**Dropped Columns**

None

**Column Format Change**

None

**Primary Key Change**

None

**Index Changes**

None

# Upgrading from Oracle Utilities Application Framework v4.3.0.6.0 to 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)

# Upgrading from Oracle Utilities Application Framework v4.4.0.0.0 to 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) |



| <b>Table</b>            | <b>Description</b>                       | <b>Table Type</b>  |
|-------------------------|------------------------------------------|--------------------|
| 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"

# Upgrading from Oracle Utilities Application Framework v4.4.0.2.0 to 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) ;
```

# Appendix H

---

## Oracle Application Framework System Table Guide

This section lists the system tables owned by the Oracle Utilities Application Framework V4.4.0.3.0 and explains the data standards of the system tables. The data standards are required for the installation of Oracle Utilities Application Framework, development within the Oracle Utilities Application Framework, and the configuration and customization of Oracle Utilities products. Adhering to the data standards is a prerequisite for seamless upgrade to future releases.

This section includes:

- [About the Application Framework System Tables](#)
- [System Table Standards](#)
- [Guidelines for System Table Updates](#)
- [System Table List](#)

# About the Application Framework System Tables

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

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

## System Table Standards

System table standards must be observed for the following reasons:

- The product installation and upgrade process and customer modification data extract processes depend on the data prefix and owner flag values to determine the system data owned by each product.
- The standards ensure that there will be no data conflict in the product being developed and the future Oracle Utilities Application Framework release. They also ensure that there will be no data conflict between customer modifications and future Oracle Utilities product releases.
- The data prefix is used to prevent test data from being released to production.

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

# Guidelines for System Table Updates

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

## Business Configuration Tables

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

### Application Security and User Profile

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

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

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

### Currency Code

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

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

### Display Profile

The Display Profile Code is referenced in the User (SC\_USER) table.

| Properties | Description                  |
|------------|------------------------------|
| Tables     | CI_DISP_PROF, CI_DISP_PROF_L |

| Properties   | Description                                                                                                                                                                                                                                                                                                                   |
|--------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Initial Data | North America (NORTHAM) and Europe (EURO) and HIJRI Format (HIJRI)<br><br><b>Configuration Note:</b> In order to use HIJRI Format display profile, additional configuration is needed to define the mappings between Hijri dates and Gregorian dates.<br><br>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 |
| Initial Data | Option 1111                                                                        |

**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 Universal Time) is provided. Implementations should define the appropriate time zone and update the installation option value accordingly.

| Properties   | Description                  |
|--------------|------------------------------|
| Tables       | CI_TIME_ZONE, CI_TIME_ZONE_L |
| Initial Data | UTC                          |

## To Do Priority and Role

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

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

## Development and Implementation System Tables

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

### Standards

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

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

Note that the system supports a system upgrade rule called Override Owner flag. If duplicate data rows (data row with same primary key values) are found at the time of upgrade, the owner flag values will get overridden. The lower level application system data will override the upper level system data. For example, F1 overrides C1, F1&C1 override CM, and so on. This rule will be applied to the following tables: CI\_CHAR\_ENTITY, CI\_MD\_MO\_ALG, C1\_PORTAL\_OPT, F1\_BUS\_OBJ\_ALG, F1\_BUS\_OBJ\_STATUS\_ALG, CI\_MD\_MO\_OPT, F1\_BUS\_OBJ\_OPT, F1\_BUS\_OBJ\_STATUS\_OPT, F1\_BUS\_OBJ\_STATUS, F1\_BUS\_OBJ\_STATUS\_L

## Algorithm Type

| Properties            | Description                                                    |
|-----------------------|----------------------------------------------------------------|
| Tables                | CI_ALG_TYPE, CI_ALG_TYPE_I, CI_ALG_TYPE_PRM, CI_ALG_TYPE_PRM_I |
| Standard Data Fields  | Algorithm Type (ALG_TYPE_CD)                                   |
| Customer Modification | None                                                           |

## Algorithm

| Properties            | Description                               |
|-----------------------|-------------------------------------------|
| Tables                | CI_ALG, CI_ALG_I, 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_I, 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_I, CI_BATCH_CTRL_P, CI_BATCH_CTRL_P_I |
| Standard Data Fields | Batch Process (BATCH_CD), Program Name (PROGRAM_NAME)              |

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

## Business Object

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

## Business Service

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

## Characteristics

| Properties            | Description                                                                                                                                                                                                                                                                |
|-----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Tables                | CI_CHAR_TYPE, CI_CHAR_TYPE_L, CI_CHAR_ENTITY, CI_CHAR_VAL, CI_CHAR_VAL_L                                                                                                                                                                                                   |
| Standard Data Fields  | Characteristic Type (CHAR_TYPE_CD)<br>Characteristic Value (CHAR_VAL) on CI_CHAR_VAL<br><br>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_INSTR_L, F1_MIGR_REQ_INSTR_ENTITY, F1_MIGR_REQ_INCL_REQ |
| Standard Data Fields  | Migration Plan Code (MIGR_PLAN_CD), Migration Request Code (MIGR_REQ_CD)                                                                                                                                           |
| Customer Modification | None                                                                                                                                                                                                               |

## Data Area

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

## Deployment Part

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

## Display Icon

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

## Extendable Lookup

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

## Foreign Key Reference

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

## Inbound Web Service

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

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

## Unsupported Metadata

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

## Lookup

| Properties            | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|-----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Tables                | CI_LOOKUP_FIELD, CI_LOOKUP_VAL, CI_LOOKUP_VAL_L                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| Standard Data Fields  | <p>Field Name (FIELD_NAME)</p> <ul style="list-style-type: none"> <li>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.</li> </ul> <p>Field Value (FIELD_VALUE)</p> <ul style="list-style-type: none"> <li>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.</li> <li>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.</li> </ul> |
| 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 |

---

| <b>Properties</b>     | <b>Description</b> |
|-----------------------|--------------------|
| Standard Data Fields  | UI Map (MAP_CD)    |
| Customer Modification | None               |

---

## **Managed Content**

---

| <b>Properties</b>     | <b>Description</b>                   |
|-----------------------|--------------------------------------|
| Tables                | F1_MANAG_CONTENT, F1_MANAG_CONTENT_L |
| Standard Data Fields  | Managed Content (MANAG_CONTENT_CD)   |
| Customer Modification | None                                 |

---

## **Messages**

---

| <b>Properties</b> | <b>Description</b>                                      |
|-------------------|---------------------------------------------------------|
| Tables            | CI_MSG_CATEGORY, CI_MSG_CATEGORY_L, CI_MSG,<br>CI_MSG_L |

---

| Properties            | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|-----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Standard Data Fields  | <p data-bbox="846 218 1308 249">Message Category (MESSAGE_CAT_NBR)</p> <ul data-bbox="894 254 1507 554" style="list-style-type: none"> <li data-bbox="894 254 1507 422">• 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.</li> <li data-bbox="894 436 1507 506">• Implementer Message Categories are 80000 and 90000</li> <li data-bbox="894 520 1247 554">• Reserved for Tests - 99999</li> </ul> <p data-bbox="846 573 1479 604">Message Number (MESSAGE_NBR) for message categories</p> <ul data-bbox="894 609 1479 705" style="list-style-type: none"> <li data-bbox="894 609 1479 705">• Message numbers below 1000 are reserved for common messages. Implementers must not use message numbers below 1000.</li> </ul> <p data-bbox="846 724 1419 785">Message Number (MESSAGE_NBR) for Java message categories</p> <ul data-bbox="894 789 1507 1514" style="list-style-type: none"> <li data-bbox="894 789 1507 821">• Subsystem Standard Messages - 00001 thru 02000</li> <li data-bbox="894 835 1268 867">• Reserved - 02001 thru 09999</li> <li data-bbox="894 882 1390 913">• Published Messages - 10001 thru 11000</li> <li data-bbox="894 928 1370 959">• Package Messages - 10001 thru 90000</li> <li data-bbox="894 974 1268 1005">• Reserved - 90001 thru 99999</li> <li data-bbox="894 1020 1507 1104">• Each package is allocated 100 message numbers, each starting from 101.</li> <li data-bbox="894 1119 1507 1398">• 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</li> <li data-bbox="894 1413 1507 1514">• Reserved message number ranges are for future use and therefore must not be used by all products.</li> </ul> |
| Customer Modification | Override Description (DESCRLONG_OVRD), Message Text Override (MESSAGE_TEXT_OVRD)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |



## Meta Data - Table and Field

| Properties            | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|-----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Tables                | CI_MD_TBL, CI_MD_TBL_FLD, CI_MD_TBL_L, CI_MD_TBL_FLD_L, CI_MD_FLD, CI_MD_FLD_L, F1_DB_OBJECTS_REPO                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| Standard Data Fields  | <p>Table Name (TBL_NAME)</p> <ul style="list-style-type: none"> <li>Table names must match with the physical table name or view name in the database.</li> <li>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.</li> <li>F1_DB_OBJECTS_REPO Table stores information about Indexes, Sequences, Triggers and other database objects excluding Tables and Fields (as they are already stored in the other Metadata tables)</li> </ul> |
| Customer Modification | <p>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.</p>                                                                                                                                                                                |

## Meta Data - Constraints

| Properties            | Description                                                                                                                                                                                                                                                         |
|-----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Tables                | CI_MD_CONST, CI_MD_CONST_FLD                                                                                                                                                                                                                                        |
| Standard Data Fields  | <p>Constraint Id (CONST_ID)</p> <ul style="list-style-type: none"> <li>Index Name for Primary Constraints</li> <li>&lt;Index Name&gt;Rnn for Foreign Key Constraints Where <ul style="list-style-type: none"> <li>nn: integer, 01 through 99</li> </ul> </li> </ul> |
| 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\_I) determines the order the menu lines appear. Within the same sequence, alphabetic sorting is used.

| Properties            | Description                                                                                      |
|-----------------------|--------------------------------------------------------------------------------------------------|
| Tables                | CI_MD_MENU, CI_MD_MENU_I, CI_MD_MENU_ITEM, CI_MD_MENU_ITEM_I, CI_MD_MENU_LINE, CI_MD_MENU_LINE_I |
| 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_I)                      |

### Meta Data - Program, Location and Services

| Properties            | Description                                                                                                                                                                     |
|-----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Tables                | CI_MD_PRG_COM, CI_MD_PRG_LOC, CI_MD_SVC, CI_MD_SVC_I, 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_I, 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_I, CI_MD_WRK_TBLFLD, CI_MD_MO_WRK |
| Standard Data Fields  | Work Table Name (WRK_TBL_NAME)                                 |
| Customer Modification | None                                                           |

## Meta Data - Search Object

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

## Mobile Component

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

## Navigation Option

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

## Outbound Message Type

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

## Portal and Zone

| Properties            | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|-----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Tables                | CI_PORTAL, CI_PORTAL_L, CI_PORTAL_ZONE, CI_PORTAL_OPT, CI_ZONE, CI_ZONE_L, CI_ZONE_PRM, CI_ZONE_HDL, CI_ZONE_HDL_L, CI_ZONE_HDL_PRM, CI_ZONE_HDL_PRM_L, CI_UI_ZONE                                                                                                                                                                                                                                                                                                                                |
| Standard Data Fields  | Portal Code (PORTAL_CD), Zone Code (ZONE_CD), Zone Type Code (ZONE_HDL_CD) <ul style="list-style-type: none"> <li>A new Zone can be added to the Product owned Portal Pages.</li> <li>The existing Zones cannot be removed from the Product owned Portal Pages.</li> </ul>                                                                                                                                                                                                                        |
| 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<br>F1_PROC_DEFN_L<br>F1_PROC_NEXT_PANEL<br>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)<br>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_PRMP, CI_SCR_PRMP_L, CI_SCR_STEP, CI_SCR_STEP_L |
| Standard Data Fields  | Script (SCR_CD)                                                                                                                                   |
| Customer Modification | None                                                                                                                                              |

## To Do Type

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

## Web Service Category

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

## XAI Configuration

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

## XAI Services

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

## Inbound Web Service Operations Parameter

| Properties            | Description                        |
|-----------------------|------------------------------------|
| Tables                | F1_IWS_SVC_OPER_PARM               |
| Standard Data Fields  | Inbound Service Name (IN_SVC_NAME) |
| Customer Modification | None                               |

## Market Configuration

| Properties            | Description                           |
|-----------------------|---------------------------------------|
| Tables                | F1_MKTCFG                             |
| Standard Data Fields  | Market Configuration Code (MKTCFG_CD) |
| Customer Modification | None                                  |

## File Integration

| Properties            | Description                                                                                         |
|-----------------------|-----------------------------------------------------------------------------------------------------|
| Tables                | F1_FILE_INT_REC, F1_FILE_INT_REC_L,<br>F1_FILE_INT_REC_ALG, F1_FILE_INT_TYPE,<br>F1_FILE_INT_TYPE_L |
| Standard Data Fields  | File Integration Record (FILE_INT_REC_CD), File Integration<br>Type (FILE_INT_TYPE_CD)              |
| Customer Modification | None                                                                                                |

## Meta Data - BI

| Properties            | Description                                       |
|-----------------------|---------------------------------------------------|
| Tables                | F1_MD_BI_TBL, F1_MD_BI_TBL_L,<br>F1_MD_BI_TBL_FLD |
| Standard Data Fields  | Table Name (BI_TBL_NAME)                          |
| Customer Modification | None                                              |

## Data Export Control

| Properties            | Description                                                                      |
|-----------------------|----------------------------------------------------------------------------------|
| Tables                | F1_DATA_EXPORT_CTRL,<br>F1_DATA_EXPORT_CTRL_LOG,<br>F1_DATA_EXPORT_CTRL_LOG_PARM |
| Standard Data Fields  | F1_DATA_EXPORT_CTRL (DATA_EXPORT_CD)                                             |
| Customer Modification | None                                                                             |

## System Table List

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

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

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

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

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

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



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

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

| Table Name        | Upgrade Action | Description                    |
|-------------------|----------------|--------------------------------|
| CI_PORTAL         | MG             | Portal                         |
| CI_PORTAL_L       | MG             | Portal Language                |
| 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_PRMPPT     | MG             | Script Prompt                  |
| CI_SCR_PRMPPT_L   | MG             | Script Prompt Language         |
| CI_SCR_STEP       | MG             | Script Step                    |
| CI_SCR_STEP_L     | MG             | Script Step Language           |
| CI_SEQ            | MG             | Sequence                       |
| CI_TD_DRLKEY_TY   | MG             | To Do Type Drill Key           |
| CI_TD_SRTKEY_TY   | MG             | To Do Type Sort Key            |
| CI_TD_SRTKEY_TY_L | MG             | To Do Type Sort Key Language   |
| CI_TD_TYPE        | MG             | To Do Type                     |
| CI_TD_TYPE_L      | MG             | To Do Type Language            |
| CI_UI_ZONE        | MG             | Context Sensitive Zone         |
| CI_USR_NAV_LINK   | MG             | User Favorite Links            |
| CI_XAI_ADAPTER    | MG             | XAI Adapter                    |
| CI_XAI_ADAPTER_L  | MG             | XAI Adapter Lang               |
| CI_XAI_CLASS      | MG             | Message Class                  |
| CI_XAI_CLASS_L    | MG             | Message Class Language         |
| CI_XAI_ENV_HNDL   | MG             | XAI Envelope Handler           |
| CI_XAI_ENV_HNDL_L | MG             | XAI Envelope Handler Language  |
| CI_XAI_IN_SVC     | MG             | XAI Inbound Service            |
| CI_XAI_IN_SVC_L   | MG             | XAI Inbound Service Language   |
| CI_XAI_SVC_PARM   | MG             | XAI Inbound Service Parameters |

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

| Table Name               | Upgrade Action | Description                                    |
|--------------------------|----------------|------------------------------------------------|
| F1_IWS_ANN_L             | MG             | Web Service Annotation Language                |
| F1_IWS_ANN_PARM          | MG             | Web Service Annotation Parameter               |
| F1_IWS_ANN_TYPE          | MG             | Web Service Annotation Type                    |
| F1_IWS_ANN_TYPE_L        | MG             | Web Service Annotation Type Language           |
| F1_IWS_ANN_TYPE_PARM     | MG             | Web Service Annotation Type Parm               |
| F1_IWS_ANN_TYPE_PARM_L   | MG             | Web Service Annotation Type Parameter Language |
| F1_IWS_SVC               | MG             | Inbound Web Service                            |
| F1_IWS_SVC_L             | MG             | Inbound Web Service Language                   |
| F1_IWS_SVC_OPER          | MG             | Inbound Web Service Operations                 |
| F1_IWS_SVC_OPER_PARM     | MG             | Inbound Web Service Operations Parameter       |
| F1_MKTCFG                | MG             | Market Configuration                           |
| F1_MANAG_CONTENT         | MG             | Managed Content                                |
| F1_MANAG_CONTENT_L       | MG             | Managed Content Language                       |
| F1_MAP                   | MG             | UI Map                                         |
| F1_MAP_L                 | MG             | UI Map Language                                |
| F1_MIGR_PLAN             | MG             | Migration Plan                                 |
| F1_MIGR_PLAN_INSTR       | MG             | Migration Plan Instruction                     |
| F1_MIGR_PLAN_INSTR_ALG   | MG             | Migration Plan Instruction Algorithm           |
| F1_MIGR_PLAN_INSTR_L     | MG             | Migration Plan Instruction Language            |
| F1_MIGR_PLAN_L           | MG             | Migration Plan Language                        |
| F1_MIGR_REQ              | MG             | Migration Request                              |
| F1_MIGR_REQ_INCL_REQ     | MG             | Migration Request Grouping                     |
| F1_MIGR_REQ_INSTR        | MG             | Migration Request Instruction                  |
| F1_MIGR_REQ_INSTR_ENTITY | MG             | Migration Request Instruction Entity           |
| F1_MIGR_REQ_INSTR_L      | MG             | Migration Request Instruction Language         |
| F1_MIGR_REQ_L            | MG             | Migration Request Language                     |
| F1_MOBILE_COMPONENT      | MG             | Mobile Component                               |
| F1_MOBILE_COMPONENT_L    | MG             | Mobile Component Language                      |
| F1_MOB_COMP_CHAR         | MG             | Mobile Component Characteristics               |

| Table Name          | Upgrade Action | Description                             |
|---------------------|----------------|-----------------------------------------|
| F1_MOB_COMP_CNT     | MG             | Mobile Component Content                |
| F1_OUTMSG_TYPE      | MG             | Outbound Message Type                   |
| F1_OUTMSG_TYPE_L    | MG             | Outbound Message Type Language          |
| F1_PROC_DEFN        | MG             | Process Flow Type                       |
| F1_PROC_DEFN_L      | MG             | Process Flow Type Language              |
| F1_PROC_NEXT_PANEL  | MG             | Next Panel                              |
| F1_PROC_PANEL       | MG             | Process Panels                          |
| F1_SCHEMA           | MG             | Schema                                  |
| F1_WEB_CAT          | MG             | Web Service Category                    |
| F1_WEB_CAT_L        | MG             | Web Service Category Language           |
| F1_WEB_CAT_INCL_SVC | MG             | Web Service Category Included Services  |
| F1_FILE_INT_REC     | MG             | File Integration Record Table           |
| F1_FILE_INT_REC_L   | MG             | File Integration Record Language Table  |
| F1_FILE_INT_REC_ALG | MG             | File Integration Record Algorithm Table |
| F1_FILE_INT_TYPE    | MG             | File Integration Type                   |
| F1_FILE_INT_TYPE_L  | MG             | File Integration Language Table         |
| F1_MD_BI_TBL        | MG             | MD Table for BI                         |
| F1_MD_BI_TBL_L      | MG             | MD Language Table for BI                |
| F1_MD_BI_TBL_FLD    | MG             | MD Field Table for BI                   |
| SC_ACCESS_CNTL      | MG             | User Group Access Control               |
| SC_APP_SERVICE      | MG             | Application Service                     |
| SC_APP_SERVICE_L    | MG             | Application Service Language            |
| SC_USR_GRP_PROF     | MG             | User Group Profile                      |
| CI_CURRENCY_CD      | KP             | Currency Code                           |
| CI_CURRENCY_CD_L    | KP             | Currency Code Language                  |
| CI_DISP_PROF        | KP             | Display Profile                         |
| CI_DISP_PROF_L      | KP             | Display Profile Language                |
| CI_TIME_ZONE        | KP             | Time Zone                               |
| CI_TIME_ZONE_L      | KP             | Time Zone Language                      |
| CI_USR_PORTAL       | KP             | User Portal                             |
| CI_XAI_JNDI_SVR     | KP             | XAI JNDI Server                         |
| CI_XAI_JNDI_SVR_L   | KP             | XAI JNDI Server Language                |

| Table Name                 | Upgrade Action | Description                               |
|----------------------------|----------------|-------------------------------------------|
| CI_XAI_OPTION              | KP             | Message Option                            |
| CI_XAI_SENDER              | KP             | Message Sender                            |
| CI_XAI_SENDER_L            | KP             | Message Sender Language                   |
| CI_XAI_SNDR_CTX            | KP             | Message Sender Context                    |
| F1_BUS_OBJ_STATUS_RSN_CHAR | KP             | Status Reason Characteristic              |
| F1_INSTALLATION            | KP             | Installation Option - Framework           |
| SC_USER                    | KP             | User                                      |
| SC_USER_CHAR               | KP             | User Characteristic                       |
| SC_USER_GROUP              | KP             | User Group                                |
| SC_USER_GROUP_L            | KP             | User Group Language                       |
| SC_USR_GRP_USR             | KP             | User Group User                           |
| CI_MD_ATT_TY               | RF             | MD Element Attribute Type                 |
| CI_MD_AT_DTL               | RF             | MD Element Attribute Type Detail          |
| CI_MD_AT_DTL_L             | RF             | MD Element Attribute Type Detail Language |
| CI_MD_CTL                  | RF             | Generator Control                         |
| CI_MD_CTL_L                | RF             | Generator Control Language                |
| CI_MD_CTL_TMPL             | RF             | Generator Control Template                |
| CI_MD_ELTY                 | RF             | MD Element Type                           |
| CI_MD_ELTY_AT              | RF             | Element Type Attributes                   |
| CI_MD_ELTY_L               | RF             | Element Type Language                     |
| CI_MD_LOOKUP_F             | RF             | MD Lookup Field                           |
| CI_MD_MSG                  | RF             | MD Message                                |
| CI_MD_MSG_L                | RF             | MD Message Language                       |
| CI_MD_PDF                  | RF             | Predefined Fields                         |
| CI_MD_PDF_VAL              | RF             | Predefined Values                         |
| CI_MD_SRC_TYPE             | RF             | Source Type                               |
| CI_MD_SRC_TYPE_L           | RF             | Source Type Language                      |
| CI_MD_TMPL                 | RF             | Template                                  |
| CI_MD_TMPL_ELTY            | RF             | Template Element Types                    |
| CI_MD_TMPL_L               | RF             | Template Language                         |
| CI_MD_TMPL_VAR             | RF             | Template Variable                         |

---

| <b>Table Name</b> | <b>Upgrade Action</b> | <b>Description</b>         |
|-------------------|-----------------------|----------------------------|
| CI_MD_TMPL_VAR_L  | RF                    | Template Variable Language |
| CI_MD_VAR         | RF                    | Variable                   |
| CI_MD_VAR_DTL     | RF                    | Variable Detail            |
| CI_MD_VAR_DTL_L   | RF                    | Variable Detail Language   |
| CI_XAI_EXECUTER   | RF                    | XAI Executer               |
| CI_XAI_EXECUTER_L | RF                    | XAI Executer Language      |
| CI_XAI_FORMAT     | RF                    | XAI Format                 |
| CI_XAI_FORMAT_L   | RF                    | XAI Format Language        |
| F1_LGCY_OBJ       | RF                    | Unsupported Metadata       |

---