DB Installation Guide

Oracle Financial Services Lending and Leasing

Release 14.12.0.0.0

Part No. F53373-01

March 2022

ORACLE®

Send Us Your Comments

Oracle Financial Services Lending and Leasing User Documentation

Oracle welcomes customers' comments and suggestions on the quality and usefulness of this document. Your feedback is important, and helps us to best meet your needs as a user of our products. For example:

- Are the implementation steps correct and complete?
- Did you understand the context of the procedures?
- Did you find any errors in the information?
- Does the structure of the information help you with your tasks?
- Do you need different information or graphics? If so, where, and in what format?
- Are the examples correct? Do you need more examples?

If you find any errors or have any other suggestions for improvement, then please tell us your name, the name of the company who has licensed our products, the title and part number of the documentation and the chapter, section, and page number (if available).

Note: Before sending us your comments, you might like to check that you have the latest version of the document and if any concerns are already addressed. To do this, access the Oracle Documentation available at www.docs.oracle.com which contains the most current Documentation Library plus all documents revised or released recently.

Send your comments using the electronic mail address: financialservices_ww@oracle.com

Please give your name, address, electronic mail address, and telephone number (optional).

If you need assistance with Oracle software, then please contact your support representative or Oracle Support Services.

If you require training or instruction in using Oracle software, then please contact your Oracle local office and inquire about our Oracle University offerings. A list of Oracle offices is available on our Web site at www.oracle.com.

Table of Contents

1. Preface

This document contains notes and installation steps involved in installation and setup of Oracle Financial Services Lending and Leasing. Oracle Financial Services Lending and Leasing relies on several pieces of Oracle software in order to run and this document is in no way meant to replace Oracle documentation supplied with these Oracle products or available via Oracle technical support. The purpose of this document is only meant to supplement the Oracle documentation and to provide Oracle Financial Services Lending and Leasing specific installation instructions.

For recommendations on security configuration, refer Security Configuration Guide.

Note

It is assumed that anyone installing Oracle Financial Services Lending and Leasing will have a thorough knowledge and understanding of Oracle Database Administration19c.

1.1 Pre-requisites

The following software is required to install Oracle Financial Services Lending and Leasing application database. They are available from the following source:

● Oracle Software Delivery Cloud

  – http://edelivery.oracle.com/

● Oracle Technology Network (OTN)

● Following database versions are supported and can be downloaded from http://www.oracle.com/technetwork/database/enterprise-edition/downloads/index.html

  – Oracle Database 19c (version 19.12.0.0.0)

Database installation is a two step process:

1. Creating Database

2. Installing the Application Database

After installing the application database, you need to:

1. Modifying Application System Parameters

2. Set the Oracle JVM File Permissions

3. Fine-Grained Access to Network Services in Oracle Database

In additional, you can configure the following:

1. Optimize PS_TXN Table in Fusion Application

2. Perform Online Application Upgrade Using EBR

3. Machine Learning for Servicing Queue Creation

1.2 Audience

This document is intended for system administrators or application developers who install Oracle Financial Services Lending and Leasing Application.

ORACLE®

1.3 Conventions Used

Term	Refers to
Home Directory/ $OFSLL_HOME	Oracle Financial Services Lending and Leasing Home Directory
Application	Oracle Financial Services Lending and Leasing

ORACLE®

2. Setup and Configuration of the Database Server

2.1 Creating Database

The first step in the database creation is the creation of Database Server Users and Groups. The below sections are a guide to:

1. Creating the Database Server Users and Groups
2. Creating Database

2.1.1 Creating the Database Server Users and Groups

The software and processes on the database server can be broken into three groups of ownership. These are; the Oracle processes, the Oracle external procedure processes, and the home directory. Some of this information applies only to UNIX server installations. The typical Windows server Application installation does not make use of separate users and groups, for the various processes and instead runs everything as the system user.

Process Group	Description
Oracle	The Oracle processes are the database processes (excluding the listener process) responsible for database operations. Normally, the Oracle processes are owned by a user named oracle and a group called dba. Some of the processes use an internal Oracle PL/SQL utility called UTL_FILE to read and/or write to files in the home directory. The UTL_FILE program will perform its functions as the oracle user, hence, add the oracle user to the Oracle Financial Services Lending and Leasing user's group.
Oracle External Procedures	Oracle Financial Services Lending and Leasing uses external procedures to perform tasks that cannot (or does not want to) be performed inside the database. Such as, interfacing with the credit bureaus, writing files, or converting images. Oracle implements external procedures through its listener process. Multiple listeners can be defined (one for "normal" database connections and one for external procedures) or run a single listener that handles both types of requests. Regardless of what is done, the external procedures will need access to files and directories that are within the home directory. It is recommended that the owner of the listener process handling external procedures is either the Oracle Financial Services Lending and Leasing user or a part of the same group.

ORACLE®

Process Group	Description
Oracle Financial Services Lending and Leasing home directory	The home directory is a top-level directory under which some application files are stored. For example, log files, document templates, email templates, temporary files, and output files. There are no processes or programs that run (outside of the Oracle external procedure processes) as the Oracle Financial Services Lending and Leasing user, this is the location where the server side application files reside. Oracle Financial Services Lending and Leasing administrator can log into the server and perform the administrative functions without needing any special system superuser or administrative privileges. The home directory is needed for each application instance (production, test, development) installed on the server.

Keeping this information in mind, here are the recommended users and groups for the server:

User	Group(s)	Description
ofsll	ofsll	Owns the home directory. Set up the directory with, group write privileges so that other application related processes can access the files and directories below it.
oracle	dba, ofsll	Owns the Oracle database processes.
ofsllext	ofsll	Owns the Oracle listener process or just the external procedure listener process. If the latter, run the "normal" SQL*Net listener process as the oracle user.

2.1.2   Creating the Database

Create a database as per corporate standards and/or Oracle best practices with the following recommendations/guidelines.

- Set NLS Characterset as AL32UTF8.

- Ensure that the external procedure listener has been configured. Set the EXTPROC_DLLS environment variable so that the external procedure listener can access the Oracle Financial Services Lending and Leasing external procedure shared libraries. A section from a *listener.ora* file and a *tnsnames.ora* file is given below as an example. This sets up the external procedure listener as a separate listener named LEXT. See the Oracle Net Services documentation for more information on setting up an external procedure listener.

ORACLE®

listener.ora

For 12c database, Unix

```
LEXT =
  (DESCRIPTION_LIST =
      (DESCRIPTION =
          (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC0))
          (PROTOCOL_STACK =
              (PRESENTATION = TTC)
              (SESSION = NS)
          )
      )
  )

SID_LIST_LEXT =
(SID_LIST =
    (SID_DESC =
        (SID_NAME = PLSExtProc)
        (ORACLE_HOME = /opt/app/oracle/product/12.1.0)
        (PROGRAM = extproc)

    (ENVS="EXTPROC_DLLS=ANY,LD_LIBRARY_PATH=/opt/app/oracle/product/12.1.0/lib")
        )
)
```

For 19c database, go to $ORACLE_HOME/hs/admin/extproc.ora and set the parameter:-
SET EXTPROC_DLLS=ANY

2.2 Installing the Application Database

Download and unzip the Application Database file (ofslldb.zip) to a staging folder.

The application database software consists of tables, indexes, types, directories, libraries, views, and packages. A script program creates the application home directory structure, database user, the required tablespaces, database objects and loads the seed data. Snap shots of the script are provided after each step, to enable easy understanding. The steps involved in installing the application database are:

1. Running the installation script

2. Selecting the Install Type

3. Setting up the Installation Environment

4. Creating Application Home directory

5. Creating Application Tablespaces

6. Creating the Application Owner User and Grant Tablespace Quotas

7. Installing Database Objects and Seed Data

8. Installing Library Objects

9. Installing Directory Objects

2.2.1 Running the installation script

From the unzipped 'Application Database file' folder, run the install script from a UNIX shell prompt and follow the on screen prompts to install the Oracle Financial Services Lending and Leasing infrastructure database objects.

ORACLE®

On Unix:

$ ./installofslldb.sh

```
Oracle Financial Services Lending and Leasing DB Installer

    Important Note:
    --------------------------------------------------------------------------------

    It is recommended to have a OS level user id created on DB server (say ofsll)
    before running the installer.

    The installer will create the required Tablespace and application schema user.

    The default set of subdirectories underneath the home directory creates a couple
    of command files needed by the application.

    The tables are grouped as follows:

    GROUP               DESCRIPTION
    -----               ---------------------
     org                origination
     svc                servicing
     cmn                common
     txn                transaction
     api                import/conversion
     adm                setup/administration
     arc                archive

Continue? [y/n]: y
```

When the script prompts for confirmation, enter 'y' to continue.

2.2.2   Selecting the Install Type

The script displays install option..

```
Oracle Financial Services Lending and Leasing DB Installer

    Important Note:
    ------------------------------------------------------------------------------

    It is recommended to have a OS level user id created on DB server (say ofsll)
    before running the installer.

    The installer will create the required Tablespace and application schema user.

    The default set of subdirectories underneath the home directory creates a couple
    of command files needed by the application.

    The tables are grouped as follows:

    GROUP               DESCRIPTION
    -----               ---------------------
     org                origination
     svc                servicing
     cmn                common
     txn                transaction
     api                import/conversion
     adm                setup/administration
     arc                archive

Continue? [y/n]: y

        Installer Options              Description
        ------------------------------------------------------------------------------------------------------------
        1.  New Installation           Full installation of OFSLL Product
        2.  Upgrade Installation       Upgrade existing installation of OFSLL Product
        3.  Install DB Objects only    Already the DB Schema and tablespaces are available install only DB Objects

Choose an installer option? [1-3]: 1
```

The script prompts to 'Choose an installer option? [1-3]'.

● Enter '1' for complete installation of Product. When entered, the script runs installer for the complete installation of the product. The below sections guide you on the process, in detail.

2-4

ORACLE®

- Enter '2' for upgrade installation. When entered, the script runs installer to upgrade the existing version of the product. [Installing Upgrade](#) section guides you on the upgrade installation.
- Enter '3' for only DB Object installation. When entered, the script runs installer to install all the OFSLL database objects. The script assumes that the user (OFSLL schema owner) and the tablespaces are created and all the required privileges are granted to the schema.

2.2.3 <u>Setting up the Installation Environment</u>

The installation script requires a properly set up environment in order to run. The script requests for few details. Enter the values as listed below. The script requests confirmation after each entry, enter 'y' to confirm.

```
Enter the Oracle Financial Services Lending and Leasing
Home Path? (usually /home/ofsll): /scratch/work_area/DEV/OFSLLREL

OFSLLHOME=/scratch/work_area/DEV/OFSLLREL
Okay? [y/n]:

Enter the Oracle Financial Services Lending and Leasing
Home Path? (usually /home/ofsll): /scratch/work_area/DEV/OFSLLREL

OFSLLHOME=/scratch/work_area/DEV/OFSLLREL
Okay? [y/n]: y

Enter the Oracle DB Home Path? /scratch/app/database19c

ORAHOME=/scratch/app/database19c
Okay? [y/n]: y

Enter the Oracle SID? OLLDB
INSTANCENAME=OLLDB
Okay? [y/n]: y

    Important Note:
    ----------------------------------------------------------------------

    Here is a list of CRITICAL environment variables and their settings:

    PATH=/usr/lib64/qt-3.3/bin:/bin:/usr/bin:/usr/dev_infra/platform/bin:/usr/dev_infra/generic/bin:/usr/local/bin
app/database19c/bin
    ORACLE_HOME=/scratch/app/database19c
    ORACLE_SID=OLLDB
    OFSLL_HOME=/scratch/work_area/DEV/OFSLLREL

    If the above environment variables are not correct, correct the environment
    and restart the script to continue.

Continue? [y/n]: y
```

Script Prompts	Description and Action Required
Oracle Financial Services Lending and Leasing Home Path	Enter the path to the application home directory. This is referred to as $OFSLL_HOME.
Oracle DB Home Path	Enter the path to the Oracle DB home directory. This is referred to as $ORACLE_HOME
Oracle SID	Enter the Name of Oracle Instance. In case of multitenant environment, enter the PDB name'
Path	Sets the path that includes $ORACLE_HOME/bin as the installation script requires SQL*Plus utility

When the script prompts for the confirmation, enter 'y' to continue.

ORACLE®

2.2.4   Creating Application Home directory

The *install* script will create a proper directory structure for the application below the user's home directory and will set the proper permissions on the directories.

$OFSLL_HOME

Variable	Description
/api	Stores the captured Applications/Account data from any third party Origination/Servicing System and setup the account in OFSLL
/acct_doc_load	Input load directory for account documents
bin	Contains executable scripts
/cor_storage	Top level directory for generated correspondences
/doc_templates	Correspondence document template load directory
/email_templates	Contains email message templates
/dot_storage	Top level directory for account documents
/cus_dot_storage	Top level directory for customer/business documents
/images	Contains fax / document images uploaded to OFSLL screen
/lib	Contains external procedure shared libraries
/logs	Contains all Oracle Financial Services Lending and Leasing log files
/output	Destination for Oracle Financial Services Lending and Leasing output data files.
/rs_archive	Top level directory for archived reports
/sql	Directory contain SQL scripts used for creating database objects, recompiling packages, and various utility functions.
/tmp	Temporary directory used by some external procedures.
/input	Repository for data files needed as input to Oracle Financial Services Lending and Leasing processes
/input/lockbox	Contains files for lockbox processing.
/input/lockbox/processed	Contains files that are already processed.
/input/adr	Contains incoming adr files.
/input/adr/processed	Contains files that are already processed.
/input/cac	Contains incoming call activity files.
/input/cac/processed	Contains files that are already processed.
/input/ifd	Contains incoming lien title tracking files.

ORACLE®

Variable	Description
/input/ifd/processed	Contains files that are already processed.
/input/itu	Contains incoming ITU files.
/input/itu/processed	Contains files that are already processed.
/input/ivr	Contains incoming IVR files.
/input/ivr/processed	Contains files that are already processed.
/input/wfp	Contains incoming wholesale floor planning files.
/input/wfp/processed	Contains files that are already processed.
/input/ibn	Contains incoming BANKO_NEW files.
/input/ibn/processed	Contains files that are already processed.
/input/ibu	Contains incoming BANKO_UPDATE files.
/input/ibu/processed	Contains files that are already processed.
/input/ice	Contains currency exchange files.
/input/ice/processed	Contains files that are already processed.
/input/icl	Contains cure letter files.
/input/icl/processed	Contains files that are already processed.
/input/ipr	Contains AP Transaction History files.
/input/ipr/processed	Contains files that are already processed.
/input/ist	Contains files for Input Sale Transfer.
/input/ist/processed	Contains files that are already processed.
/input/ipu	Contains files for Payment Upload.
/input/ipu/processed	Contains files that are already processed.
/input/ipi	Contains files for Personally Identifiable Information (PII).
/input/ipi/processed	Contains files that are already processed.
input/iuh	Contains files for Input Usage History.
input/iuh/processed	Contains files that are already processed.
input/ifc	Contains files for Offline Cross Upsell Activity Posting.
input/ifc/processed	Contains files that are already processed.
input/icp	Contains files for Customer Based Payment Upload.
input/icp/processed	Contains files that are already processed.
input/icc	Contains files for Collateral Management Upload.

Variable	Description
input/icc/processed	Contains files that are already processed.
input/isc	Contains files for Securitization Pool Upload.
input/isc/processed	Contains files that are already processed.
input/iar	Contains Asset Billing Rate Setup Create/Update file upload.
input/iar/processed	Contains files that are already processed.
input/api	Contains Upload Legacy Account Information.
input/api/processed	Contains files that are already processed.
input/itr	Contains files for tracking attribute upload process.
input/itr/processed	Contains files that are already processed.
input/ipc	Contains files for Customer Payment Upload.
input/ipc/processed	Contains files that are already processed.
input/gls	Contains files for GL Attributes Upload.
input/gls/processed	Contains files that are already processed.
input/prp	Contains files for Product Pricing Upload.
input/prp/processed	Contains files that are already processed.
input/iuu	Contains files for User Upload.
input/iuu/processed	Contains files that are already processed.
input/ias	Contains files for Assets Upload.
input/ias/processed	Contains files that are already processed.
input/iav	Contains files for Asset Valuations Upload.
input/iav/processed	Contains files that are already processed.
input/iat	Contains files for Asset Tracking Attributes.
input/iat/processed	Contains files that are already processed.
input/iaa	Contains files for Asset Attributes.
input/iaa/processed	Contains files that are already processed.
input/iad	Contains files for Account Dues Upload.
input/iad/processed	Contains files that are already processed.

2.2.5 Creating Application Tablespaces

The Oracle Financial Services Lending and Leasing table and index creation DDL allow for the use of up to 22 different tablespaces --11 for tables and 11 for indexes. The DDL commands reference a set of files that contain storage parameter information. These files

ORACLE®

allow mapping of logical tablespace names to physical tablespaces. Create as many of these tablespaces as necessary depending on the storage resources. Use locally managed tablespaces with a uniform extent policy. The following table describes different tablespaces, their content, and their expected growth pattern.

Tablespace	Storage Parameter File	Default Tablespace Name	Data	Comment
Admin	storage_parms_table_adm.sql storage_parms_index_adm.sql	OFSLL_ADM_DATA OFSLL_ADM_INDEX	setup data	low growth, low change.
Origination	storage_parms_table_org.sql storage_parms_index_org.sql	OFSLL _ORG_DATA OFSLL _ORG_INDEX	loan application data	growth varies by customer, few updates.
Servicing	storage_parms_table_svc.sql storage_parms_index_svc.sql	OFSLL _SVC_DATA OFSLL _SVC_INDEX	non transaction account data	growth varies by customer, few updates.
Transactions	storage_parms_table_txn.sql storage_parms_index_txn.sql	OFSLL _TXN_DATA OFSLL _TXN_INDEX	transaction data	high growth (relative to account and application data table spaces)
API	storage_parms_table_api.sql storage_parms_index_api.sql	OFSLL _API_DATA OFSLL _API_INDEX	application/ account data used during imports/ conversions	high growth, one time usage, data can be removed after loading
Archive	storage_parms_table_arc.sql storage_parms_index_arc.sql	OFSLL _ARC_DATA OFSLL _ARC_INDEX	Archived application/ account data	steady growth, potentially very large
Common	storage_parms_table_cmn.sql storage_parms_index_cmn.sql	OFSLL _CMN_DATA OFSLL_CMN_INDEX	common non-admin data	generally low growth, some tables can be periodically truncated
Input Process	storage_parms_table_ipf.sql storage_parms_index_ipf.sql	OFSLL_IPF_DATA OFSLL_IPF_INDEX	Input processing file	steady growth, potentially very large

ORACLE®

Tablespace	Storage Parameter File	Default Tablespace Name	Data	Comment
Output Process	storage_parms_table_opf.sql storage_parms_index_opf.sql	OFSLL_OPF_DATA OFSLL_OPF_INDEX	Output processing file	steady growth, potentially very large
Logging	storage_parms_table_log.sql storage_parms_index_log.sql	OFSLL_LOG_DATA OFSLL_LOG_INDEX	Error Logging	steady growth
Audit	storage_parms_table_aud.sql storage_parms_index_aud.sql	OFSLL_AUD_DATA OFSLL_AUD_INDEX	Audit Logging	Growth depends on enabling number of audits fields.

ORACLE®

```
     Important Note:
     -------------------------------------------------------------------

     Here is a list of CRITICAL environment variables and their settings:

     PATH=/usr/lib64/qt-3.3/bin:/bin:/usr/bin:/usr/dev_infra/platform/bin:/usr/dev_infra/generic/bin:/usr/local/bin:
app/database19c/bin
     ORACLE_HOME=/scratch/app/database19c
     ORACLE_SID=OLLDB
     OFSLL_HOME=/scratch/work_area/DEV/OFSLLREL                 .

     If the above environment variables are not correct, correct the environment
     and restart the script to continue.


Continue? [y/n]: y

Data already exists in /scratch/work_area/DEV/OFSLLREL,
Continue with override data? [y/n]: y


     Oracle Financial Services Lending and Leasing Default Tablespace Name and Size (default 300MB)

        Default Tablespace Name   Description                       New Name            Size (MB)
        -------------------------------------------------------------------------------------------------

        1.  OFSLL_ADM_DATA        setup data
        2.  OFSLL_ADM_INDEX       setup data index
        3.  OFSLL_ORG_DATA        loan application data
        4.  OFSLL_ORG_INDEX       loan application data index
        5.  OFSLL_SVC_DATA        non transaction account data
        6.  OFSLL_SVC_INDEX       non transaction account data index
        7.  OFSLL_TXN_DATA        transaction data
        8.  OFSLL_TXN_INDEX       transaction data index
        9.  OFSLL_API_DATA        api data
       10.  OFSLL_API_INDEX       api data index
       11.  OFSLL_ARC_DATA        archive data
       12.  OFSLL_ARC_INDEX       archive data index
       13.  OFSLL_CMN_DATA        common non-admin data
       14.  OFSLL_CMN_INDEX       common non-admin data index
       15   OFSLL_IPF_DATA        incoming process file data
       16   OFSLL_IPF_INDEX       incoming process file index
       17   OFSLL_OPF_DATA        outgoing process file data
       18   OFSLL_OPF_INDEX       outgoing process file index
       19   OFSLL_LOG_DATA        log files header data
       20   OFSLL_LOG_INDEX       log files header index
       21   OFSLL_AUD_DATA        Audit data
       22   OFSLL_AUD_INDEX       Audit data index
       23.  Confirm Tablespace Values

     Note:
     -------------------------------------------------------------------------------------------------
          To Change the Default Tablespace Name and Size
              a. Select a number from 1 to 22 to redefine each tablespace name and its size
              b. Select number 23 to confirm the redefined tablespace values

          To Accept the Default Tablespace Name and Size
              a. Select number 23 to confirm the default tablespace values
Redefine the tablespace name?
select a number [1-23]: █
```

When prompted to 'Select a number' to redefine the tablespace name, you have 2 options:

Accept default Tablespace Name and Size: Select # 21

```
          Redefine the tablespace name?
          select a number [1-23]: 23█
```

ORACLE®

Modify the default Tablespace Name and Size: Select the number of the tablespace, between 1–20, you need to modify

```
Oracle Financial Services Lending and Leasing Default Tablespace Name and Size (default 300MB)

   Default Tablespace Name    Description                              New Name            Size (MB)
   ------------------------------------------------------------------------------------------------------
   1.   OFSLL_ADM_DATA         setup data
   2.   OFSLL_ADM_INDEX        setup data index
   3.   OFSLL_ORG_DATA         loan application data
   4.   OFSLL_ORG_INDEX        loan application data index
   5.   OFSLL_SVC_DATA         non transaction account data
   6.   OFSLL_SVC_INDEX        non transaction account data index
   7.   OFSLL_TXN_DATA         transaction data
   8.   OFSLL_TXN_INDEX        transaction data index
   9.   OFSLL_API_DATA         api data
   10.  OFSLL_API_INDEX        api data index
   11.  OFSLL_ARC_DATA         archive data
   12.  OFSLL_ARC_INDEX        archive data index
   13.  OFSLL_CMN_DATA         common non-admin data
   14.  OFSLL_CMN_INDEX        common non-admin data index
   15   OFSLL_IPF_DATA         incoming process file data
   16   OFSLL_IPF_INDEX        incoming process file index
   17   OFSLL_OPF_DATA         outgoing process file data
   18   OFSLL_OPF_INDEX        outgoing process file index
   19   OFSLL_LOG_DATA         log files header data
   20   OFSLL_LOG_INDEX        log files header index
   21   OFSLL_AUD_DATA         Audit data
   22   OFSLL_AUD_INDEX        Audit data index
   23.  Confirm Tablespace Values

   Note:
   ------------------------------------------------------------------------------------------------------
        To Change the Default Tablespace Name and Size
             a. Select a number from 1 to 22 to redefine each tablespace name and its size
             b. Select number 23 to confirm the redefined tablespace values

        To Accept the Default Tablespace Name and Size
             a. Select number 23 to confirm the default tablespace values

Redefine the tablespace name?
select a number [1-23]: 1
```

Script Prompts	Description and Action Required
Enter new tablespace name	Enter tablespace name you need to assign to the default tablespace you have selected.
Enter tablespace size (in MB)	Enter the size (in MB) you need to assign to the new tablespace name you have entered

ORACLE®

Repeat this process for all the tablespace names you need to modify

```
Oracle Financial Services Lending and Leasing Default Tablespace Name and Size (default 300MB)

   Default Tablespace Name   Description                        New Name           Size (MB)
   ---------------------------------------------------------------------------------------------

   1.  OFSLL_ADM_DATA         setup data
   2.  OFSLL_ADM_INDEX        setup data index
   3.  OFSLL_ORG_DATA         loan application data
   4.  OFSLL_ORG_INDEX        loan application data index
   5.  OFSLL_SVC_DATA         non transaction account data
   6.  OFSLL_SVC_INDEX        non transaction account data index
   7.  OFSLL_TXN_DATA         transaction data
   8.  OFSLL_TXN_INDEX        transaction data index
   9.  OFSLL_API_DATA         api data
   10. OFSLL_API_INDEX        api data index
   11. OFSLL_ARC_DATA         archive data
   12. OFSLL_ARC_INDEX        archive data index
   13. OFSLL_CMN_DATA         common non-admin data
   14. OFSLL_CMN_INDEX        common non-admin data index
   15  OFSLL_IPF_DATA         incoming process file data
   16  OFSLL_IPF_INDEX        incoming process file index
   17  OFSLL_OPF_DATA         outgoing process file data
   18  OFSLL_OPF_INDEX        outgoing process file index
   19  OFSLL_LOG_DATA         log files header data
   20  OFSLL_LOG_INDEX        log files header index
   21  OFSLL_AUD_DATA         Audit data
   22  OFSLL_AUD_INDEX        Audit data index
   23. Confirm Tablespace Values

   Note:
   ---------------------------------------------------------------------------------------------
       To Change the Default Tablespace Name and Size
           a. Select a number from 1 to 22 to redefine each tablespace name and its size
           b. Select number 23 to confirm the redefined tablespace values

       To Accept the Default Tablespace Name and Size
           a. Select number 23 to confirm the default tablespace values

Redefine the tablespace name?
select a number [1-23]: 23

Enter Oracle datafile path : /scratch/app/database19c/oradata/OLLDB
```

Enter # 21 to redefine and to apply changes for the tablespaces you modified.

```
Enter Oracle datafile path : /scratch/app/db19c/oradata/ORCL

Oracle datafile path: /scratch/app/db19c/oradata/ORCL
Okay? [y/n]: y

Enter the Oracle userid (schema name) that will own the Oracle Financial Services Lending and Leasing
objects? (usually ofsllprd): OFSLLREL

Enter the password for this userid:

Enter the Oracle sysdba
userid? (usually sys): sys

Enter the password for this userid:
```

Script Prompts	Description and Action Required
Oracle tablespace path	Installed Database path that stores DBF files. Once entered the script prompts for confirmation. Enter 'y', if it is ok
Oracle User ID that will own the Oracle Financial Services Lending and Leasing objects	Valid User ID
Password for this User ID	Valid Password
Oracle sysdba User ID	Valid User ID
Password for sysdba User ID	Valid Password

2.2.6 Creating the Application Owner User and Grant Tablespace Quotas

Oracle Financial Services Lending and Leasing application requires a single Oracle user (or schema) and this user is the application owner. The application owner user owns all of the tables, indexes, views, sequences, packages, etc. that make up the application.

ORACLE®

The user is then granted access to the application tablespaces.

2.2.7 Installing Database Objects and Seed Data

Once the User ID is created, the script initiates the database object installation and lists the objects available for installation.

```
Oracle Financial Services Lending and Leasing Database Object Installation

The following items are available for installation:

                          1.   database libraries      (3)
                          2.   database directories     (59)
                          3.   database tables          (1648)
                          4.   common database views    (5)
                          5.   engine views             (4053)
                          6.   form views               (1333)
                          7.   java views               (421)
                          8.   database types           (298)
                          9.   database package specs   (3678)
                          10.  database package bodies  (3656)
                          11.  database indexes         (1999)
                          12.  System Seed Data         (67935)


Continue with Installation? [y/n] :
```

When the script prompts for the confirmation, enter 'y' to continue.

```
The Oracle Financial Services Lending and Leasing shared libraries are normally installed in a lib directory
below where the $OFSLL home directory is.  For example:
/home/ofsll/lib OR D:\ofsll\lib

Enter name of Oracle Financial Services Lending and Leasing shared library directory: /home/oracle/scratch/OFSLL/OFSLL143/lib


PL/SQL procedure successfully completed.
```

2.2.8 Installing Library Objects

Once the confirmation is received, the script initiates installation of library objects. A sample of the script is given below.

```
The Oracle Financial Services Lending and Leasing shared libraries are normally installed in a lib directory
below where the $OFSLL home directory is.  For example:
/home/ofsll/lib OR D:\ofsll\lib

Enter name of Oracle Financial Services Lending and Leasing shared library directory: /scratch/work_area/DEV/OFSLLREL/lib


PL/SQL procedure successfully completed.


Installing Directory objects...
crt_cordir_cmn.sql

The Oracle Financial Services Lending and Leasing schema owner owns all of the objects.

Enter name of Oracle Financial Services Lending and Leasing schema owner: OFSLLREL

This script will create an Oracle directory object that will be
used by the correspondence engine to determine where to store
document files.  A separate storage directory can be set up for
each company that is configured in Oracle Financial Services Lending and Leasing.  Enter the
company code for the company that you are configuring.  If you
don't know at this time, enter HQ

Enter the company code: HQ
```

ORACLE®

```
Enter name of Oracle Financial Services Lending and Leasing shared library directory: /export/home/ofss/lib

PL/SQL procedure successfully completed.
```

Script Prompts	Description and Action Required
Shared Library Directory	The path for the library directory. The default value is $OFSLL_HOME/lib

2.2.9 Installing Directory Objects

The fax images, correspondence documents, and the account document images are stored as BFILE objects. With BFILE, the image or document is physically stored outside the database on a file system available either on the database server or to the database server. A pointer to the file consisting an Oracle directory object and a file name is stored in the database. Application programs that create the BFILEs use application parameters to determine the specific directory object name. The application parameters are company level parameters, hence, a different directory object can be created for each defined company and branch.

```
The Oracle Financial Services Lending and Leasing shared libraries are normally installed in a lib directory
below where the $OFSLL home directory is.  For example:
/home/ofsll/lib OR D:\ofsll\lib

Enter name of Oracle Financial Services Lending and Leasing shared library directory: /scratch/work_area/DEV/OFSLLREL/lib

PL/SQL procedure successfully completed.

Installing Directory objects...
crt_cordir_cmn.sql
The Oracle Financial Services Lending and Leasing schema owner owns all of the objects.

Enter name of Oracle Financial Services Lending and Leasing schema owner: OFSLLREL

This script will create an Oracle directory object that will be
used by the correspondence engine to determine where to store
document files.  A separate storage directory can be set up for
each company that is configured in Oracle Financial Services Lending and Leasing.  Enter the
company code for the company that you are configuring.  If you
don't know at this time, enter HQ

Enter the company code: HQ

The COR_STORAGE_DIRECTORY is the top-level directory below which
the document files will be stored.
For example: /home/ofsll/cor_storage or d:\ofsll\cor_storage

Enter name of COR_STORAGE_DIRECTORY: /scratch/work_area/DEV/OFSLLREL/cor_storage

Directory created.

crt_dotdir_cmn.sql

The Oracle Financial Services Lending and Leasing schema owner owns all of the objects.

Enter name of Oracle Financial Services Lending and Leasing schema owner: OFSLLREL

This script will create an Oracle directory object that will be
used by the account document load batch job to determine where to
account document files.  Enter the company code for
company that you are configuring.  If you don't know at this
time, enter HQ

Enter the company code: HQ

The DOT_STORAGE_DIRECTORY is the top-level directory below which
the account document files will be stored.
For example: /home/ofsll/dot_storage or d:\ofsll\dot_storage
Enter name of DOT STORAGE DIRECTORY: /scratch/work_area/DEV/OFSLLREL/dot_storage

The DOT_STORAGE_DIRECTORY is the top-level directory below which
the account document files will be stored.
For example: /home/ofsll/dot_storage or d:\ofsll\dot_storage

Enter name of DOT_STORAGE_DIRECTORY: /scratch/work_area/DEV/OFSLLREL/dot_storage

Directory created.

crt_utldir_cmn.sql

The Oracle Financial Services Lending and Leasing schema owner owns all of the objects.

Enter name of Oracle Financial Services Lending and Leasing schema owner: OFSLLREL

This script will create a number of Oracle directory objects that will be
used to grant the UTL_FILE package read and write access to application
directories.

The OFSLL HOME is the top-level directory below which
all application input and output files will be stored.
For example: /home/ofsll

Enter directory path for OFSLL HOME : /scratch/work_area/DEV/OFSLLREL
```

ORACLE®

Choose a name that identifies the usage, company, and branch for the directory object. Since directory objects are for the entire database and not just for the application, it is suggested that the schema name or identifier is used in the name to identify it from other directory objects. Choose a directory path with enough storage space to handle the expected volume of correspondences.

Directory Object Types	Abbreviation for Object Types	Suggested installation Location in $OFSLL_HOME
Correspondence interface directory objects	COR	*cor_storage*
Account documents interface directory objects	DOT	*dot_storage*

Loading Seed Data

The factory shipped seed data is automatically uploaded during installation and once complete, a confirmation message is displayed as indicated below:

```
Generating Seed Data...

Done.
```

2.3   Modifying Application System Parameters

Several system parameters in the seed data require modification to fit the local installation environment. One such parameter must be set manually. You can modify the rest using the application system parameter setup screen (Setup > Administration > System > System Parameter).

Setting the CMN_SERVER_HOME parameter values

As mentioned, the CMN_SERVER_HOME parameters must be set manually before the application screens can be used to set other parameters.

CMN_SERVER_HOME

Set it to the **$OFSLL_HOME** directory.

Setting the remaining system parameters manually or from the setup screen

```
$ sqlplus

SQL*Plus: Release 19.0.0.0.0 Production on Fri Feb 12 15:28:17 2020
Copyright (c) 1982, 2019, Oracle. All rights reserved.

Enter user-name: OFSLL/OFSLL

Connected to:
Oracle Database 19c Enterprise Edition Release 19.3.0.0.0 - 64bit Production
with the Partitioning, OLAP, Advanced Analytics and Real Application Testing
options

SQL> update system_parameters set syp_value = '$OFSLL_HOME' where
syp_parameter_cd = 'CMN_SERVER_HOME';

1 row updated.
SQL> commit;
```

Set the below parameters which are located on the Setup→Administration→System→System Parameters screen. This is not a complete list of system parameters, but is a list of system parameters related to installation details. The other

ORACLE®

parameters are related to application business functionality. The values for many of these parameters are known only after the installation and configuration of application server.

1. CMN_SER_ENVIRONMENT_FILE

   – Set it to the full path of the text file that contains environment settings that will be used when running external commands from the job service. For example, $OFSLL_HOME/ofsllenv

2. CMN_SERVER_TEMP_DIR

   – Set it to the full path of a directory that will be used to store temporary files needed during certain procedures. This is typically a directory named "*tmp*" underneath the $OFSLL_HOME directory (ex. $OFSLL_HOME*tmp* or $OFSLL_HOME*/tmp*).

3. ADMIN_SERVER_URL

   – This parameter is used to enable changing the user-level passwords. The default parameter value will generally have the form:

   Set the t3://hostname:port/console (Host name and Port) where application is installed

4. UIX_DEFAULT_IMAGE_PATH

   – Set the path to the fax / document images that are uploaded to OFSLL screen. Typically, it is $OFSLL_HOME/images.

5. Update system parameters with Oracle directory objects used by the application.

   – Login as an application schema owner and execute the following sql file from the dba_utils folder - 'update_sys_parms.sql'.

Modifying Oracle Analytics Publisher Parameters

1. JSV_REPORTS_SERVER_URL

Set this to the URL to the job service should use when running a report. Reports are all submitted as a background job (either by a user or by the job scheduler). These reports will be sent to a printer/file system as chosen by the user or as configured in the job setup. The parameter value will generally have the form: http://hostname:port/ (Host name and Port number of xmlp server where BIP is installed)

2. JSV_USE_REPORTS_SERVER

Set to 'Yes

3. JSV_REPORT_ARCHIVE_DIRECTORY

Set to a directory that will be used to store OFSLL report PDFs. Typically it is $OFSLL_HOME/rs_archive

Batch reports can be configured to print to a special "archive" printer that will create a PDF of the report and store it below the JSV_REPORT_ARCHIVE_DIRECTORY. This directory must be accessible to the database server machine. All archived reports are stored by a process running on the database server machine and not by a process running on the report server machine.

The report PDFs are stored below the JSV_REPORT_ARCHIVE_DIRECTORY in a subdirectory representing the type of report (report or correspondence), the year, the month, and the day that the report was printed. For example, if a report was printed on December 15, 2013, the PDF file will be located in: JSV_REPORT_ARCHIVE_DIRECTORY/reports/2013/12/15

4. JSV_ARCHIVE_SERVER_URL

ORACLE®

Set this to the URL that the job service should use when running an archived report. Archive reports are all reports submitted as a background job (either by a user or by the job scheduler) to a special printer named ARCHIVE. The parameter value will generally have the form: http://hostname:port/ (Host name and Port number of xmlp server where BIP is installed)

This can have the same value as the JSV_REPORTS_SERVER_URL parameter. This additional parameter is supplied in case there are multiple report servers and it is desirable to send archive requests to a different server than the other report requests.

5. JSV_SMTP_SERVER - smtp mail router of organization

6. JSV_USE_BI_PUBLISHER - Set to YES (Y)

7. JSV_BI_USER - set as the BI Publisher Admin User Name *(Required only if the system parameter 'OUTBOUND_CALL_Q' is set to 'N')*.

8. JSV_BI_PASSWORD - set as the BI Publisher Admin Password *(Required only if the system parameter 'OUTBOUND_CALL_Q' is set to 'N')*.

9. UIX_REPORTS_SERVER_URL

Set this to the URL that user interface should use when running a report. The parameter value will generally have the form: http://hostname:port/ (Host name and Port number of xmlp server where BIP is installed). This can have the same value as the JSV_REPORTS_SERVER_URL parameter.

Setting the correspondence interface company parameters manually or from the setup screen.

If using the correspondence interface, set these additional parameters located on the Setup→Administration→Company→System Parameters screen. These parameters are company level parameters, which means that a correspondence interface can be configured for each company in Oracle Financial Services Lending and Leasing.

COR_STORAGE_DIRECTORY

Set to the name of the Oracle *directory object,* created during installation, which defines where the generated correspondences will be stored. The correspondence files are stored in the database as a BFILE which means that only a reference to an external file is stored in the database and the actual file is stored outside on the operating system's file system in a path defined by an Oracle directory object. This parameter is the name of the Oracle object, not the actual directory path. Typical naming convention used for COR directory objects is - COR_DIR_<application owner name>_company_branch. For example: COR_DIR_OFSLLREL_HQ

Setting the account documents interface company parameters manually or from the setup screen.

If using the account documents interface, set these additional parameters located on the Setup→Administration→Company→System Parameters screen. These parameters are company level parameter, which means that an account documents interface can be configured for each company in Oracle Financial Services Lending and Leasing.

DOT_STORAGE_DIRECTORY

Set to the name of the Oracle *directory object,* created during installation, which defines where the account documents will be stored. The account document files are stored in the database as a BFILE which means that only a reference to an external file is stored in the database and the actual file is stored on the operating system's file system in a path defined by an Oracle directory object. This parameter is the name of the Oracle object, not the actual directory path. Typical naming convention used for DOT directory objects is -

DOT_DIR_<application owner name>_company_branch. For example:
DOT_DIR_OFSLLREL_HQ

2.4   <u>Creating Users and Granting Access</u>

Creating the user 'INTERNAL'. This user is required for batch job process, webservices and to start services.

A script is provided in the distribution media in the dba_utils folder to create an user. Run the script "crt_app_user.sql script" as a OFSLL application owner user.

```
SQL> @crt_app_user.sql;
Enter the name of the OFSLL App user Id you
Want to create user: INTERNAL
Enter the First Name for this user: OFSLL
Enter the Last Name for this user: RELEASE
Enter the Phone Number for this user: 1234567891
Enter the Fax Number for this user: 1234567891

1 row created.


1 row created.


1 row created.
```

<u>Granting OFSLL Screen / Web Service Access to Application Users (New Installation only)</u>

Post user creation, to access all the 'Screens' and 'web service' for a specific responsibility, login as an application schema owner and do the following:

1. Execute the below SQL statement:
   TRUNCATE TABLE FLS_ACCESS_DETAILS;

   ```
   SQL> TRUNCATE TABLE FLS_ACCESS_DETAILS;

   Table truncated
   ```

2. Execute the following db script available in dba_utils folder.

   set_screen_access.sql

In the prompt, enter the user responsibility. You can also specify 'ALL' to provide screen / web service access to all the created users in bulk.

In the next prompt 'source(UI/WS)', enter one of the following:

- 'UI' to provide screen access to either All or specific user responsibility.

ORACLE®

- 'WS' to provide web service access to either All or specific user responsibility.

```
SQL> @set_screen_access.sql;
Enter responsibility_cd to continue or Q to Exit. :ALL
Enter source(UI/WS) to continue or Q to Exit. :UI
Granted access to ALL
No of rows inserted :7

PL/SQL procedure successfully completed.

Enter Commit to save the changes or Rollback if you want to revert changes.
SQL> COMMIT;

Commit complete.
```

Further, you can customized the access through Setup > User > Access screen - 'Screen' and 'Webservice' tabs, as detailed in setup guide.

Granting user access to Reports, Transactions, & Correspondence

Post user creation, to access all the Reports, Transactions, Correspondence for a specific responsibility, login as an application schema owner and execute the following db script available in dba_utils folder.

set_rpt_txn_cor_access.sql

In the prompt, enter the user responsibility. You can also specify 'ALL' to provide screen access to all the created users in bulk.

```
SQL> @set_rpt_txn_cor_access.sql;
Enter responsibility_cd to continue or Q to Exit :ALL
Granted access to Reports, Correspondance and Txn codes.

PL/SQL procedure successfully completed.

Enter Commit to save the changes or Rollback if you want to revert changes.
SQL> commit;

Commit complete.

SQL>
```

Further, you can customized the access through Setup > User > Access screen as detailed in setup guide.

Note

In case where the application schema is created using an export and import utility, please reset the sequences by executing 'upgrade_fix_reset_all_sequences.sql file available in the data fix folder of the media. This helps to reset all the sequences to the current level.

2.5   Set the Oracle JVM File Permissions

The application correspondence interface uses the built-in Oracle Java Virtual Machine (JVM) to create subdirectories below the top-level correspondence/document storage directory (defined by the application's COR_STORAGE_DIRECTORY, DOT_STORAGE_DIRECTORY parameters). In order to do this, the JVM's security policy needs to be updated by the Oracle built-in DBMS_JAVA package to allow directories to be created and accessed. A SQL script named *set_java_perms.sql* has been supplied on the

ORACLE®

installation media in the *dba_utils* directory for this purpose. The script will select all directory object names defined for the application COR_STORAGE_DIRECTORY and DOT_STORAGE_DIRECTORY parameters and will provide read, write, and delete privileges to the directory named by the directory object and all directories below it.

Running the set_java_perms.sql script

After the system and company parameters have been set up, run SQL*Plus as the SYS user and execute the *set_java_perms.sql* script to set the file permissions.

Similarly, in case of any java permission access issues to directories, provide the required read, write or delete privileges.

2.6   Advanced Queues

Queuing feature is used in OFSLL for writing debugs into the XMLTYPE column of LOG_FILES_HEADER table instead of writing into the file system.

Also, in OFSLL, Outbound database calls are routed through application server through AQ JMS bridge. The MDB deployed in Middleware, reads the AQ message remotely which has all information required to make the call to Bureau or RO /DT.

The MDB deployed in Middleware, reads the AQ message ('OFSLL_OUTBOUND_TOPIC') remotely which has information about job set and job set status code, which has been initiated by job scheduler.

To enable alert and debug message queue:

1. Set the system parameter "CMN_DEBUG_METHOD" with syp_value = 4.

    – SQL> update system_parameters set syp_value = '4' where syp_parameter_cd = 'CMN_DEBUG_METHOD';

To enable MDB EJB queue:

2. Set the system parameter and credit bureau parameter in "OUTBOUND_CALL_Q" with syp_value = Y.

    – UPDATE SYSTEM_PARAMETERS SET SYP_VALUE='Y' WHERE SYP_PARAMETER_CD='OUTBOUND_CALL_Q' AND SYP_ENABLED_IND='Y'

    – UPDATE CREDIT_BUREAU_PARAMETERS SET CRP_VALUE='Y' WHERE CRP_PARAMETER_CD='OUTBOUND_CALL_Q'

3. Start the queue services using the sql file

    – SQL>setup_AQ_subscribe.sql.

4. Once the queue services are started, system subscribes and registers the queue for a notification so that whenever a data inserted into the queue it will notify the system.

To switch back to the existing file system mechanism:

1. To unsubscribe, de-register the queue and stops the queue services for notification, run the below sql file:

    – SQL>setup_AQ_unsubscribe.sql

2. Set the system parameter "CMN_DEBUG_METHOD"with syp_value= 1.

    – SQL> update system_parameters set syp_value = '1' where syp_parameter_cd = 'CMN_DEBUG_METHOD';

ORACLE®

Note

    – Please do not delete any queue or queue table from the schema.

    – When you unsubscribe the queue, MDB queue is also stopped. You need to start 'OFSLL_OUTBOUND_Q' and 'OFSLL_OUTBOUND_TOPIC' to continue with MDB flow.

    – Ensure that you have granted 'dbms_aqin' privileges to the schema user.

2.7   Fine-Grained Access to Network Services in Oracle Database

Oracle allows access to external network services using several PL/SQL APIs (UTL_TCP, UTL_SMTP, UTL_MAIL, UTL_HTTP and UTL_INADDR), all of which are implemented using the TCP protocol.

An Access Control Entry can be limited to specific PL/SQL APIs (UTL_TCP, UTL_INADDR, UTL_HTTP, UTL_SMTP, and UTL_MAIL).

In a multitenant environment, Access Control Entries (ACEs) can be created at the CDB or PDB level. For the examples in this article, all the host ACLs and host ACEs will be created at the PDB level. The following code creates two test users in a PDB.

Append an Access Control List (ACE)

Host ACL are never created directly. Instead, they are implicitly created when we append a host Access Control Entry (ACE) using the DBMS_NETWORK_ACL_ADMIN.APPEND_HOST_ACE procedure. If we append a new ACE to a host that has no existing host ACL, a new host ACL is implicitly created. If the host already has an ACL, the new host ACE will be appended to the existing host ACL.

Login to SQL*Plus as the SYS user.

```
BEGIN
  DBMS_NETWORK_ACL_ADMIN.append_host_ace (
    host      => ' 192.168.2.3 ',
    lower_port => 80,
    upper_port => 80,
    ace       => xs$ace_type(privilege_list => xs$name_list('http', 'connect'),
                     principal_name => 'OFSLL',
                     principal_type => xs_acl.ptype_db));
END;
/
```

Append a second host to the existing ACL

```
BEGIN
  DBMS_NETWORK_ACL_ADMIN.APPEND_HOST_ACL (
    host       => 192.168.2.3 ',
    lower_port => 80,
    upper_port => 80,
    acl        => 'NETWORK_ACL_02B9BC669CA6110CE0536638A8C05D8A');
END;
/
```

Parameter Definitions

The parameters used in the procedures and functions above

Parameters	Description
host	Any valid host name or IP address. Wildcards are allowed.

ORACLE®

Parameters	Description
lower_port	Specific port number, or lower part of a range of ports.
upper_port	Upper part of a range of ports. If NULL, it defaults to the lower_port value.
ace	The access control entry, defined using the XS$ACE_TYPE type.

The XS$ACE_TYPE type has the following definition.

```
xs$ace_type(privilege_list => xs$name_list('privilege1', 'privilege2'),
            principal_name => 'user-or-role',
            principal_type => xs_acl.ptype_db)
```

Parameters	Description
privilege_list	The list of privileges available to the ACE.
princi-pal_name	The database user the ACE applies to.
principal_-type	You will always use XS_ACL.PTYPE_DB for these network ACEs as they apply to users and roles.

The privilege_list specifies one or more privileges in a comma separated list. The available privileges are shown below.

Parameters	Description
http	Access restricted to the UTL_HTTP package and the HttpUriType type.
http_proxy	Needed in conjunction with http if HTTP access is via a proxy.
smtp	Access restricted to the UTL_SMTP and UTL_MAIL packages.
resolve	Access restricted to the UTL_INADDR packages.
connect	Opens access to the UTL_TCP, UTL_SMTP, UTL_MAIL, UTL_HTTP, and DBMS_LDAP packages and the HttpUriType type.
jdwp	Enables Java Debug Wire Protocol debugging operations.

2.8   Installing Upgrade

An upgrade is a process of updating an existing version to its higher version. For example, upgrading from Oracle Financial Services Lending and Leasing 14.11.0.0.0 to 14.12.0.0.0.

The following upgrade path is recommended for existing Daybreak customers:

DLS 11.6.0.0.23 > OFSLL 14.0.0.0.0 > OFSLL 14.1.0.0.0 > OFSLL 14.2.0.0.0 > OFSLL 14.3.0.0.0 > OFSLL 14.3.1.0.0 > OFSLL 14.4.0.0.0 > OFSLL 14.5.0.0.0 > OFSLL 14.6.0.0.0 > OFSLL 14.7.0.0.0 > OFSLL 14.8.0.0.0 > OFSLL 14.9.0.0.0 > OFSLL 14.10.0.0.0 > OFSLL 14.11.0.0.0 > 14.12.0.0.0

Note

– Please refer to 'Upgrade Installation Guide' for detailed information.

ORACLE®

- If table compression feature is enabled in OFSLL, disable table compression prior to upgrade.

ORACLE®

3. Security Features

3.1 Enabling Transparent Data Encryption to Secure Stored Data

Oracle Database uses authentication, authorization, and auditing mechanisms to secure data in the database. The operating system data files where data is stored is not used. To protect these data files, Oracle Database provides Transparent Data Encryption (TDE). TDE encrypts sensitive data stored in data files. Encrypted data is transparently decrypted for a database user or application that has access to data.

OFSLL application processes sensitive data. Hence, it is recommended to use a TDE to protect confidential data, such as credit card and social security numbers, stored in table columns.

A script is provided along with the distribution media to encrypt the sensitive columns in the table. You can follow the steps below to enable TDE for column encryption:

To start using TDE, the security administrator must create a wallet and set a master key. The wallet can be the default database wallet shared with other Oracle Database components, or a separate wallet specifically used by TDE. Oracle strongly recommends that you use a separate wallet to store the master encryption key.

Specifying a Wallet Location for Transparent Data Encryption

If you wish to use a wallet specifically for TDE, then you must specify a wallet location in the sqlnet.ora file by using the ENCRYPTION_WALLET_LOCATION parameter. Oracle recommends that you use the ENCRYPTION_WALLET_LOCATION parameter to specify a wallet location for TDE.

ENCRYPTION_WALLET_LOCATION = (SOURCE =

(METHOD = FILE)

(METHOD_DATA =

(DIRECTORY =

 /etc/ORACLE/WALLETS/oracle)))

If no wallet location is specified in the sqlnet.ora file, then the default database wallet location is used. The default database wallet location is ORACLE_BASE/admin/DB_UNIQUE_NAME/wallet or ORACLE_HOME/admin/DB_UNIQUE_NAME/wallet. Here, DB_UNIQUE_NAME is the unique name of the database specified in the initialization parameter file.

Setting the Master Encryption Key

The master encryption key is stored in an external security module, and is used to protect the table keys and tablespace encryption keys. By default, the master encryption key is a random key generated by Transparent Data Encryption (TDE). It can also be an existing key pair from a PKI certificate designated for encryption. To use TDE with PKI key pairs, the issuing certificate authority must be able to issue X.509v3 certificates with the key usage field marked for encryption.

To set the master encryption key, use the following command:

ORACLE®

SQL> ALTER SYSTEM SET ENCRYPTION KEY IDENTIFIED BY 'password'

where,

- password is the mandatory wallet password for the security module, with no default setting. It is case sensitive. Enclose the password string in double quotation marks (" ").
- The wallet location specified by the ENCRYPTION_WALLET_LOCATION parameter, in the sqlnet.ora parameter file, is used to create the master encryption key. If the ENCRYPTION_WALLET_LOCATION parameter is not present in the sqlnet.ora file, then the WALLET_LOCATION value is used. A new wallet is created in case of no wallet.
- If no wallet location is specified in the sqlnet.ora file, then the default database wallet location is used. The default database wallet location is ORACLE_BASE/admin/ DB_UNIQUE_NAME/wallet or ORACLE_HOME/admin/DB_UNIQUE_NAME/wallet. Here, DB_UNIQUE_NAME is the unique name of the database specified in the initialization parameter file.
- If an existing auto login wallet is present at the expected wallet location, then a new wallet is not created.

Resetting the Master Encryption Key

Reset/Regenerate the master encryption key only if it has been compromised or as per the security policies of the organization. You should back up the wallet before resetting the master encryption key.

Use the ALTER SYSTEM command to set or reset (rekey) the master encryption key.

Opening and Closing the Encrypted Wallet

The database must load the master encryption key into memory before it can encrypt or decrypt columns/tablespaces. Opening the wallet allows the database to access the master encryption key. Use the following ALTER SYSTEM command to explicitly open the wallet:

SQL> ALTER SYSTEM SET ENCRYPTION WALLET OPEN IDENTIFIED BY "password"; where password is the password to open the wallet. You should enclose the password string in double quotation marks (" ").

Once the wallet has been opened, it remains open until you shut down the database instance, or close it explicitly by issuing the following command:

SQL> ALTER SYSTEM SET ENCRYPTION WALLET CLOSE IDENTIFIED BY "password"

Encrypting Columns in Existing Tables

To add an encrypted column to an existing table, or to encrypt or decrypt an existing column, you use the ALTER TABLE SQL command with the ADD or MODIFY clause.

SQL> ALTER TABLE applicants MODIFY (apl_gender_cd ENCRYPT USING 'AES256');

Encrypting the indexed columns:

SQL> ALTER TABLE applicants MODIFY (apl_ssn ENCRYPT USING 'AES256' NO SALT);

Disabling Encryption on a Column

You may want to disable encryption for reasons of compatibility or performance. To disable column encryption, use the ALTER TABLE MODIFY command with the DECRYPT clause.

ORACLE®

Example 7-11 Turning Off Column Encryption:

SQL> ALTER TABLE applicants MODIFY (apl_gender_cd DECRYPT);

A Set of scripts are provided on the installation media in the dba_utils directory to encrypt the recommended columns in OFSLL.

upgrade_tb_tde_enable_ofsll.sql to encrypt base table columns.

upgrade_tb_opur_tde_enable_ofsll.sql to encrypt archive table columns.

upgrade_tb_api_tde_enable_ofsll.sql to encrypt api table columns.

Login as an OFSLL user and execute the scripts to encrypt the columns.

Note

Refer Oracle® Database Advanced Security Administrator's Guide, section on Securing Stored Data Using Transparent Data Encryption for details.

3.2   <u>Data Redaction for Data Masking</u>

Data Redaction is one of the new features available in 19c. Data Redaction is in Advanced Security option of enterprise edition.

Oracle Advanced Security Data Redaction provides selective, on-the-fly redaction of sensitive data in SQL query results prior to display by applications so that unauthorized users cannot view the sensitive data.

OFSLL application processes sensitive data. Hence, it is recommended to use a Data Redaction to protect confidential data, such as credit card and social security numbers, stored in table columns.

There are different types of redaction; full, partial, regexp, random and none. Please refer Oracle® Database Advanced Security Administrator's Guide, section on Configuring Oracle Data Redaction Policies for details.

ORACLE®

4. Optimize PS_TXN Table in Fusion Application

Oracle Fusion Applications use the PS_TXN table to store the intermediate processing state. When there are many concurrent users, this table receives a high number of inserts and could suffer from concurrency issues.

Follow the steps outlined in note ID 1444959.1 in My Oracle Support to alleviate the contention.

ORACLE®

5. Perform Online Application Upgrade Using EBR

EBR (Edition-Based Redefinition) is a feature supported in Oracle Database which facilitates to upgrade the database component of an application while it is in use, thereby minimizing or eliminating downtime.

Upgrading by this process means, copying the database objects that comprise the application and redefining the copied objects in isolation. These changes do not affect application users and they continue to run the unchanged application. When you are sure that the changes are correct, you can make the upgraded application available to all users.

Note

Enabling EBR involves a set of configurations which are to be performed by system administrator who is well-versed with installation and upgrade of OFSLL environment. Since this is an irreversible process, ensure that you are familiar with EBR and take extra caution before you proceed. The information provided in this section is specific to EBR configuration process in OFSLL and for detailed information about EBR, refer to https://docs.oracle.com/database/121/ADFNS/adfns_editions.htm#ADFNS020

If you require EBR (Edition Based Redefinition) to be enabled, the same can be done after successful application database installation. To configure OFSLL with EBR, follow the below configuration process.

Before you begin,

- Place the executable file (config_ebr.sh) in staging folder.
- The procedures directory consists of sql files which are executed by the installation scripts to create procedures in the schema. Hence, create a directory for installation to write the sql files which will be used in subsequent steps.
- Ensure that the above mentioned directories have read, write and execute permissions before installation.

To Configure EBR on OFSLL Schema

1. Navigate to staging directory and execute the installer script config_ebr.sh

   ```
   -bash-4.1$ ./config_ebr.sh
   ```

2. On seeing the below prompt, enter OFSLL_HOME path. When the script prompts for confirmation, enter 'y' to continue.

   ```
   Enter the Oracle Financial Services Lending and Leasing
   Home Path? (This is usually /home/ofsll): /scratch/work_area/DEV/OFSLLREL
   OFSLLHOME=/scratch/work_area/DEV/OFSLLREL
   Okay [y/n]?: Y
   ```

3. On seeing the below prompt, enter Oracle DB Home Path. When the script prompts for confirmation, enter 'y' to continue.

   ```
   Enter the Oracle DB Home Path? /scratch/app/db12c/product/12.1.0/dbhome_1

   ORAHOME=/scratch/app/db12c/product/12.1.0/dbhome_1
   Okay [y/n]?: Y
   ```

ORACLE®

4. On seeing the below prompt, enter Oracle SID. When the script prompts for confirmation, enter 'y' to continue.

```
Enter the Oracle SID? OFSLLDB

INSTANCENAME=OFSLLDB
Okay [y/n]?: Y
```

5. Verify the critical environment variables and confirm 'Y' to continue.

```
Important Note:
------------------------------------------------------------------------
-------
Here is a list of CRITICAL environment variables and their settings:

PATH=/scratch/db/db12102/app/product/12.1.0/dbhome_1/bin:/scratch/db/db12102/app/p
roduct/12.1.0/dbhome_1/bin:/usr/lib64/qt-3.3/bin:/usr/kerberos/sbin:/usr/kerberos/
bin:/bin:/usr/bin:/usr/dev_infra/platform/bin:/usr/dev_infra/generic/bin:/usr/loca
l/bin:/usr/X11R6/bin:/usr/local/ade/bin:/scratch/app/db12c/product/12.1.0/dbhome_1
/bin
ORACLE_HOME=/scratch/app/db12c/product/12.1.0/dbhome_1
ORACLE_SID=OFSLLDB
OFSLL_HOME=/scratch/work_area/DEV/OFSLLREL

If the above environment variables are not correct, correct the environment and re
start
the script to continue.

Okay to continue? [y/n]:
```

6. Enter Oracle sysdba user id and press 'Enter'. You are prompted to enter password of Oracle sysdba. Enter the password and press 'Enter'.

```
Enter the Oracle sysdba userid (usually sys) of OFSLLDB : sys
 You entered: sys
Enter the password of Oracle sysdba :
```

7. Enter User ID who owns OFSLL objects and press 'Enter'. You are prompted to enter password for the user. Enter the password and press 'Enter'.

```
Enter the Oracle OFSS application username for which EBR need to be enabled (usual
ly OFSLL schemaname) : OFSLLREL
You entered: OFSLLREL

Enter the password for this OFSLLREL
```

8. Enter the staging area path where EBR installation scripts reside and press 'Enter'. The below screen is provided as an example.

```
Enter directory where EBR scripts exists.

/scratch/ebr_configuration
```

9. The installer creates required procedures in the schema.

```
Creating procedure alter_type_nonedition

Procedure created.
```

10. When prompted, enter the path of directory for procedures to write sql files and press 'Enter'. This is the same directory which was created before starting the installation process.

```
Enter directory path where procedure generates a sql file, for EBR e.g  /scratc
h/app/db12c/backup/sql_op :

/scratch/ebr_configuration/sql_ebr
You entered: /scratch/ebr_configuration/sql_ebr
```

ORACLE®

11. Enter name of the directory object that is to be created to access the above path and press 'Enter'.

```
/scratch/ebr_configuration/sql_ebr
You entered: /scratch/ebr_configuration/sql_ebr
ENTER DIRECTORY NAME  e.g SQL_OP_DIR : SQL_EBR
```

12. The installation process starts and the installer creates the required procedures and executes them.

```
Creating procedure create_edview.

Procedure created.

 Creating procedure recompile_objects to recompile_objects.

Procedure created.

new create edition view script file can now be created
```

The installation completes after recompiling all the invalid objects.

ORACLE®

6. Machine Learning for Servicing Queue Creation

OFSLL is equipped to leverage Oracle Data Mining capability to give additional Machine Learning features. Oracle Data Mining provides a powerful, state-of-the-art data mining capability within Oracle Database. Machine Learning capability is leveraged to identify the Queue/Segmentation that can be created for the Account data. Intelligent Segmentation feature provides list of Clusters/Queues using the Machine Learning Algorithm for a given account condition.

Orthogonal Partitioning Clustering (O-Cluster), an Oracle-proprietary Clustering algorithm, has been used to create Intelligent Segments/Clusters for a given condition. Clustering Algorithm will discover natural groupings in given data.

This helps to automate the manual process of queue creation which is otherwise done by identifying different segments of Accounts and assigning day to day Customer Service Activities.

The Intelligent Segmentation screen in OFSLL UI is developed using the Oracle JavaScript Extension Toolkit (Oracle JET) frame work. For information on usability, refer to 'Intelligent Segmentation' section in setup guide.

To enable Machine Learning Service Queue creation in OFSLL:

1. Provide Grant Mining Model privilege to the schema user.

   GRANT CREATE MINING MODEL TO <Schema>;

2. Execute the sql script 'upgrade_ML_dataset.sql' (Available in data fix folder of release bundle) to load service account's data on to ML table.

Following is a known issue:

- External tables are not applicable in ATP-D

Note

Ensure that customer service accounts exist in the system so that based on the data present in accounts, account conditions and assets tables gets loaded on to ML table.

---

3. Enable menu access for 'Intelligence Segmentation' screen with key column data 'FLL.SET.JET.INTELLIGENTSEGMENTATION.MENU|SETUP_USER_MENU' from seed data setup screen.

For more information on Oracle Data Mining, refer to

https://www.oracle.com/database/technologies/advanced-analytics/odm.html

ORACLE®

7. Oracle Autonomous Database for Transaction Processing

An autonomous database leverages AI and machine learning to provide full, end-to-end automation for provisioning, security, updates, availability, performance, change management, and error prevention. This is a Self-Driving, Self-Securing, Self-Repairing database service.

OFSLL can be deployed on Oracle Autonomous Transaction Processing Dedicated (ATP-D)

However, the following challenges / limitations exist:

- Conversion (Data Migration of Legacy Data)
- SQL loader utility is required to install the application schema seed data.
- Interface with Credit Bureaus.
- To facilitate Application/Account Document upload

7.1 Pre- requisites

Following are the pre-requisites before installation:

- It is assumed that the ATP-D is provisioned and have access to database 'Admin credentials.
- An application VM with Oracle instant client installed. SQL loader utility available along with the client.

7.2 Installation

Download and unzip OFSLL Release Bundle from e-delivery.

On Unix:

- – $ unzip atp-d_install.zip
- – $ ./installofsllatpdb.sh

For detailed information on database installation, refer to Running the installation script section and follow the steps up to section 2.2.7. Installing Database Objects and Seed Data.

For installing webservice database objects, refer to section '2. Installing WebServices Database Objects' in Web Service installation guide.

For more information on Autonomous Transaction Processing, refer to

https://docs.oracle.com/en/cloud/paas/atp-cloud/index.html.

ORACLE®