

Database Practices Guide  
Oracle FLEXCUBE Investor Servicing  
Release 14.0.0.0.0  
[May] [2018]



# Table of Contents

<b>1. INTRODUCTION .....</b>	<b>1-1</b>
<b>2. DATABASE INITIALIZATION PARAMETERS.....</b>	<b>2-1</b>
2.1 DB_WRITER_PROCESSES.....	2-1
2.2 CURSOR_SHARING.....	2-1
2.3 DB_CACHE_ADVICE .....	2-2
2.4 FAST_START_MTTR_TARGET .....	2-2
2.5 FILESYSTEM_IO_OPTIONS .....	2-3
2.6 JOB_QUEUE_PROCESSES .....	2-3
2.7 NLS_DATE_FORMAT.....	2-4
2.8 OPEN_CURSORS.....	2-4
2.9 OPTIMIZER_DYNAMIC_SAMPLING.....	2-5
2.10 OPTIMIZER_INDEX_CACHING.....	2-5
2.11 OPTIMIZER_INDEX_COST_ADJ .....	2-6
2.12 PLSQL_CODE_TYPE .....	2-6
2.13 PLSQL_OPTIMIZE_LEVEL .....	2-7
2.14 PROCESSES .....	2-7
2.15 QUERY_REWRITE_ENABLED .....	2-8
2.16 REMOTE_DEPENDENCIES_MODE.....	2-8
2.17 RESULT_CACHE_MAX_SIZE .....	2-9
2.18 CLIENT_RESULT_CACHE_LAG.....	2-9
2.19 CLIENT_RESULT_CACHE_SIZE .....	2-10
2.20 SESSION_CACHED_CURSORS.....	2-10
2.21 SKIP_UNUSABLE_INDEXES .....	2-11
2.22 UNDO_RETENTION .....	2-11
2.23 UTL_FILE_DIR .....	2-12
2.24 DEFERRED_SEGMENT_CREATION .....	2-12
2.25 LOG_BUFFER .....	2-12
2.26 ALLOW_LEVEL_WITHOUT_CONNECT_BY .....	2-13
2.27 PGA_AGGREGATE_LIMIT .....	2-13
2.28 OPTIMIZER_ADAPTIVE_FEATURES .....	2-13
2.29 PARALLEL_ADAPTIVE_MULTI_USER.....	2-14
2.30 MEMORY_TARGET .....	2-14
2.31 MEMORY_MAX_TARGET .....	2-14
2.32 TRANSPARENT DATABASE ENCRYPTION .....	2-15
<b>3. REDO LOG FILES .....</b>	<b>3-1</b>
<b>4. PLSQL OPTIMIZER LEVEL .....</b>	<b>4-1</b>
<b>5. ORACLE FLEXCUBE DATABASE STORAGE RECOMMENDATIONS .....</b>	<b>5-1</b>
5.1 KEY BENEFITS OF ASM .....	5-1
<b>6. ORACLE FLEXCUBE DATABASE BACKUP RECOMMENDATIONS.....</b>	<b>6-1</b>
6.1 RMAN Vs CONVENTIONAL BACKUP .....	6-1
6.2 BENEFITS OF USING RMAN .....	6-2
6.3 BACKUP STRATEGY RECOMMENDATION .....	6-3
<b>7. TABLESPACE LAYOUT AND MOVING TABLES TO RESPECTIVE TABLESPACES .....</b>	<b>7-1</b>
<b>8. TABLE AND INDEX PARTITIONING .....</b>	<b>8-1</b>
<b>9. SEQUENCE CACHING .....</b>	<b>9-1</b>

<b>10.</b>	<b>STATISTICS COLLECTION FOR FCIS SCHEMA (RECOMMENDED METHOD)</b> .....	<b>10-1</b>
10.1	CUSTOMIZING DEFAULT STATISTICS COLLECTION SCHEDULE .....	10-1
10.2	CUSTOMIZING STATISTICS GATHERING FOR FCIS .....	10-3
10.2.1	<i>Statistics Histograms</i> .....	<i>10-3</i>
10.2.2	<i>Sample Size of Statistics</i> .....	<i>10-3</i>
<b>11.</b>	<b>FCIS PERIODIC TABLE MAINTENANCE</b> .....	<b>11-1</b>
<b>12.</b>	<b>FCIS FUNCTIONALITY RELATED PERFORMANCE CHANGES</b> .....	<b>12-1</b>
<b>13.</b>	<b>APPENDIX</b> .....	<b>13-1</b>
13.1	ORACLE 12C DB BENEFITS OF CONSOLIDATING APPLICATION WORKLOADS .....	13-1
13.2	SCRIPT TO CHECK HISTOGRAMS ON FCIS SCHEMA.....	13-1
13.3	SCRIPT TO REMOVE HISTOGRAMS ON FCIS SCHEMA .....	13-2

---

# 1. Introduction

This document contains FCIS Database best practices.

---

## 2. Database Initialization Parameters

Oracle FLEXCUBE standard database initialization parameters have been derived after performing the required benchmark tests (Performance Load tests).

**Note:** Since some of the initialization parameters values are specific to customer volume, parameters should be derived using [FLEXCUBE-Disk-Layouts-initparams-12c.xlsm](#) excel sheet base lined along with this document.

Following are the Parameters with the details and its relevance to Oracle FLEXCUBE:

### 2.1 DB WRITER PROCESSES

This is useful for systems that modify data heavily. It specifies the initial number of database writer processes for an instance.

Property	Description
Parameter type	Integer
Default value	1 or CPU_COUNT / 8, whichever is greater
Range of Values	1 to 20
Recommended	Refer <a href="#">FLEXCUBE-Disk-Layouts-initparams-12c.xlsm</a>

#### Oracle FLEXCUBE relevance

Multiple DB writer processes helps faster flushing of data to disk. To arrive right value, refer [FLEXCUBE-Disk-Layouts-initparams-12c.xlsm](#) excel.

### 2.2 CURSOR SHARING

Determines what kind of SQL statements can share the same cursors.

Property	Description
Parameter type	String
Default value	EXACT
Recommended Value	Force

### Oracle FLEXCUBE relevance

Some of the Oracle FLEXCUBE sql statements are generated dynamically. So they contain literal values in the WHERE clause conditions. This results in large numbers of nearly identical statements with separate parse trees in Oracle's library cache, which can slow performance and cause latch problems.

By setting cursor\_sharing to FORCE database convert literals to bind variables before parsing the statement.

## **2.3 DB CACHE ADVICE**

This enables or disables statistics gathering used for predicting behaviour with different cache sizes through the V\$DB\_CACHE\_ADVICE performance view.

Property	Description
Parameter type	String
Syntax	DB_CACHE_ADVICE = { ON   READY   OFF }
Default value	If STATISTICS_LEVEL is set to TYPICAL / ALL, then ON If STATISTICS_LEVEL is set to BASIC, then OFF
Recommended Value	OFF (Should be ON while Performance Monitoring)

### Oracle FLEXCUBE relevance

Turning ON advisory will have an extra overhead. Please note it should be ON, only during performance monitoring.

## **2.4 FAST START MTTR TARGET**

This enables you to specify the number of seconds the database takes to perform crash recovery of a single instance. When specified, FAST\_START\_MTTR\_TARGET is overridden by LOG\_CHECKPOINT\_INTERVAL.

Property	Description
Parameter type	Integer
Default value	0
Range of values	0 to 3600 seconds
Recommended Values	300

### Oracle FLEXCUBE relevance

If FAST\_START\_MTTR\_TARGET is not set to 300 then run time performance for write/redo generation intensive workloads will not be optimized. This will reduce checkpoint writes from DBWR processes, making more room for LGWR IO. To optimize run time performance for write/redo generation intensive workloads, increase the FAST\_START\_MTTR\_TARGET initialization parameter to 300.

## 2.5 FILESYSTEM IO OPTIONS

This specifies the IO operation for file system files.

Property	Description
Parameter type	String
Default value	There is no default value
Range of values	NONE,SETALL,DIRECTIO,ASYNC
Recommended Values	SETALL

### Oracle FLEXCUBE relevance

By setting parameter value to SETALL, Oracle can take advantage of direct I/O and asynchronous I/O on supported platforms. However, this parameter will not have any effect if ASM is being used.

## 2.6 JOB QUEUE PROCESSES

This specifies the maximum number of processes that can be created for the execution of jobs. It specifies the number of job queue processes per instance (J000, J999).

Property	Description
Parameter type	Integer
Default value	0
Range of values	0 to 1000
Recommended Values	Refer <a href="#">FLEXCUBE-Disk-Layouts-initparams-12c.xlsm</a>

### Oracle FLEXCUBE relevance

This parameter has to be set with respect to the maximum number of jobs (dbms\_jobs). To arrive at the right value, refer [FLEXCUBE-Disk-Layouts-initparams-12c.xlsm](#) excel.

## 2.7 NLS\_DATE\_FORMAT

This specifies the default date format to use with the TO\_CHAR and TO\_DATE functions.

Property	Description
Parameter type	String
Syntax	NLS_DATE_FORMAT = "format"
Default value	Derived from NLS_TERRITORY
Recommended Values	DD-MON-RRRR

### Oracle FLEXCUBE relevance

Oracle FLEXCUBE standard date format

## 2.8 OPEN\_CURSORS

This specifies the maximum number of open cursors (handles to private SQL areas) a session can have at once. You can use this parameter to prevent a session from opening an excessive number of cursors.

Property	Description
Parameter type	Integer
Default value	50
Modifiable	ALTER SYSTEM
Range of values	1 to 4294967295 (4 GB -1)
Recommended Values	5000

### Oracle FLEXCUBE relevance

It is important to set the value of OPEN\_CURSORS high enough to prevent FCIS application from running out of open cursors (ORA-01000: maximum open cursors exceeded).



## 2.9 OPTIMIZER\_DYNAMIC\_SAMPLING

This controls the level of dynamic sampling performed by the optimizer.

Property	Description
Parameter type	Integer
Default value	If OPTIMIZER_FEATURES_ENABLE is set to 10.0.0 or higher, then 2 If OPTIMIZER_FEATURES_ENABLE is set to 9.2.0, then 1 If OPTIMIZER_FEATURES_ENABLE is set to 9.0.1 or lower, then 0
Recommended Values	1
Range of values	0 to 10

### Oracle FLEXCUBE relevance

Dynamic Sampling is a method of gathering additional statistics during optimization by recursively sampling statements. When dynamic sampling is enabled, queries are recursively generated by Oracle to test various selectivity based upon real values in order to improve their accuracy. This can result in the production of better explain plans.

Value 1 Sample all tables that have not been analyzed that meet certain criteria.

## 2.10 OPTIMIZER\_INDEX\_CACHING

This lets you adjust the behaviour of cost-based optimization to favor nested loops joins and IN-list iterators.

Property	Description
Parameter type	Integer
Default value	0
Recommended Values	90
Range of values	0 to 100

### Oracle FLEXCUBE relevance

The cost of executing an index using IN-list iterators or of executing nested loops join when an index is used to access the inner table depends on the caching of that index in the buffer cache. Oracle FLEXCUBE favors nested loop joins by setting optimizer\_index\_caching to 90.

## 2.11 OPTIMIZER\_INDEX\_COST\_ADJ

This lets you tune optimizer behaviour for access path selection to be more or less index friendly - that is, to make the optimizer more or less prone to selecting an index access path over a full table scan.

Property	Description
Parameter type	Integer
Default value	100
Recommended Values	50
Range of values	1 to 10000

### Oracle FLEXCUBE relevance

Oracle FLEXCUBE favors index read over full table scan as it is very useful when optimizer favors to give a lower cost to index scans over full-table scans.

## 2.12 PLSQL\_CODE\_TYPE

This specifies the compilation mode of the PL/SQL units.

Property	Description
Parameter type	String
Default value	INTERPRETED
Recommended values	NATIVE
Range of values	INTERPRETED, NATIVE

### Oracle FLEXCUBE relevance

The PL SQL interpreter overhead will be minimal when set to NATIVE.

## 2.13 PLSQL\_OPTIMIZE\_LEVEL

This specifies the optimization level that will be used to compile PL/SQL library units. The higher the setting of this parameter, the more effort the compiler makes to optimize PL/SQL library units.

Property	Description
Parameter type	Integer
Default value	2
Recommended values	2
Range of values	0 to 3

### Oracle FLEXCUBE relevance

This applies a wide range of modern optimization techniques beyond those of level 1 including changes which may move source code relatively far from its original location.

## 2.14 PROCESSES

This specifies the maximum number of operating system user processes that can simultaneously connect to Oracle. Its value should allow for all background processes such as locks, job queue processes, and parallel execution processes.

Property	Description
Parameter type	Integer
Default value	100
Range of values	6 to operating system dependent
Recommended values	Refer <a href="#">FLEXCUBE-Disk-Layouts-initparams-12c.xlsm</a>

### Oracle FLEXCUBE relevance

This parameter can set be set with respect to maximum no of sessions connected to DB.

## 2.15 QUERY\_REWRITE\_ENABLED

Allows you to enable or disable query rewriting globally for the database.

Property	Description
Parameter type	String
Syntax	QUERY_REWRITE_ENABLED = { false   true   force }
Default value	If OPTIMIZER_FEATURES_ENABLE is set to 10.0.0 or higher, then true If OPTIMIZER_FEATURES_ENABLE is set to 9.2.0 or lower, then false
Recommended values	FALSE

### Oracle FLEXCUBE relevance

Oracle FLEXCUBE does not use function-based indexes.

## 2.16 REMOTE\_DEPENDENCIES\_MODE

Specifies how Oracle should handle dependencies upon remote PL/SQL stored procedures.

Property	Description
Parameter type	String
Syntax	REMOTE_DEPENDENCIES_MODE = { TIMESTAMP   SIGNATURE }
Default value	TIMESTAMP
Recommended values	SIGNATURE

### Oracle FLEXCUBE relevance

Oracle allows the procedure to execute as long as the signatures are considered safe. This setting allows client PL/SQL applications to be run without recompilation.

## 2.17 RESULT\_CACHE\_MAX\_SIZE

RESULT\_CACHE\_MAX\_SIZE specifies the maximum amount of SGA memory (in bytes) that can be used by the Result Cache.

Property	Description
Parameter type	Big integer
Syntax	RESULT_CACHE_MAX_SIZE = integer [K   M   G]
Default value	Derived from the values of SHARED_POOL_SIZE, SGA_TARGET, and MEMORY_TARGET
Recommended values	0.5% of SGA

### Oracle FLEXCUBE relevance

Automatic memory management: If you are using the MEMORY\_TARGET initialization parameter to specify memory allocation, Oracle Database allocates 0.25% of the value of the MEMORY\_TARGET parameter to the result cache.

Automatic shared memory management: If you are managing the size of the shared pool using the SGA\_TARGET initialization parameter, Oracle Database allocates 0.50% of the value of the SGA\_TARGET parameter to the result cache.

Manual shared memory management: If you are managing the size of the shared pool using the SHARED\_POOL\_SIZE initialization parameter, then Oracle Database allocates 1% of the shared pool size to the result cache.

## 2.18 CLIENT\_RESULT\_CACHE\_LAG

CLIENT\_RESULT\_CACHE\_LAG specifies the maximum time (in milliseconds) since the last round trip to the server, before which the OCI client query execute makes a round trip to get any database changes related to the queries cached on the client.

Property	Description
Parameter type	Big integer
Syntax	CLIENT_RESULT_CACHE_LAG = integer
Default value	3000
Recommended values	10,800,000(3 hours)

## 2.19 CLIENT RESULT CACHE SIZE

CLIENT\_RESULT\_CACHE\_SIZE specifies the maximum size of the client per-process result set cache (in bytes). All OCI client processes inherit this maximum size. Setting a nonzero value enables the client query cache feature. This can be overridden by the client configuration parameter OCI\_RESULT\_CACHE\_MAX\_SIZE.

Property	Description
Parameter type	Big integer
Syntax	CLIENT_RESULT_CACHE_SIZE=integer [K M G]
Default value	0
Recommended values	32K

## 2.20 SESSION CACHED CURSORS

Specifies the number of session cursors to cache. Repeated parse calls of the same SQL statement cause the session cursor for that statement to be moved into the session cursor cache. Subsequent parse calls will find the cursor in the cache and do not need to reopen the cursor. Oracle uses a least recently used algorithm to remove entries in the session cursor cache to make room for new entries when needed.

Property	Description
Parameter type	Integer
Default value	50
Recommended values	400
Range of values	0 to operating system-dependent

### **Oracle FLEXCUBE relevance**

This helps to cache the cursor thus avoid parsing of the cursor which heavy CPU intensive particularly in batch.

## 2.21 SKIP\_UNUSABLE\_INDEXES

Enables or disables the use and reporting of tables with unusable indexes or index partitions.

Property	Description
Parameter type	Boolean
Default value	true
Recommended values	FALSE
Range of values	true / false

### Oracle FLEXCUBE relevance

TRUE enables error reporting of indexes marked UNUSABLE. This setting does not allow inserts, deletes, and updates on tables with unusable indexes or index partitions. IT is set to false, because Oracle FLEXCUBE application should throw error if any of the indexes become UNUSABLE.

## 2.22 UNDO\_RETENTION

This specifies (in seconds) the low threshold value of undo retention. For AUTOEXTEND undo tablespaces, the system retains undo for at least the time specified in this parameter, and automatically tunes the undo retention period to satisfy the undo requirements of the queries. For fixed- size undo tablespaces, the system automatically tunes for the maximum possible undo retention period, based on undo tablespace size and usage history, and ignores UNDO\_RETENTION unless retention guarantee is enabled.

The UNDO\_RETENTION parameter can only be honoured if the current undo tablespace has enough space. If an active transaction requires undo space and the undo tablespace does not have available space, then the system starts reusing unexpired undo space. This action can potentially cause some queries to fail with a "snapshot too old" message.

Property	Description
Parameter type	Integer
Default value	900
Range of values	0 to 231 – 1
Recommended values	1800

### Oracle FLEXCUBE relevance

Increased value along with automatic undo management helps to avoid "snapshot too old error".

## 2.23 UTL\_FILE\_DIR

Lets you specify one or more directories that Oracle should use for PL/SQL file I/O. If you are specifying multiple directories, you must repeat the UTL\_FILE\_DIR parameter for each directory on separate lines of the initialization parameter file.

Property	Description
Parameter type	String
Syntax	UTL_FILE_DIR = pathname
Default value	There is no default value
Recommended values	/tmp/Flexcube

### Oracle FLEXCUBE relevance

Security recommends to create one single directory for writing all the DEBUG related files. This should be in sync with the DEBUG related parameter values mentioned in CSTB\_PARAM table. If any other components require to write to a different location, the same needs to be updated in UTL\_FILE\_DIR as well.

## 2.24 DEFERRED SEGMENT CREATION

Specifies the semantics of deferred segment creation. If set to true, then segments for non-partitioned tables and their dependent objects (LOBs, indexes) will not be created until the first row is inserted into the table.

Property	Description
Parameter type	Boolean
Default value	True
Modifiable	ALTER SESSION, ALTER SYSTEM
Recommended value	False

### Oracle FLEXCUBE relevance

All Oracle FLEXCUBE tables should be imported / created even though there is no record in the table.

## 2.25 LOG\_BUFFER

Recommended Value: 100M



### Oracle FLEXCUBE relevance

The default log buffer size is too small as Oracle FLEXCUBE performs heavy DML during batch processing.

## 2.26 ALLOW LEVEL WITHOUT CONNECT BY

Recommended Value: TRUE

This parameter is set to avoid following error,

After Upgrading To Oracle 10g, Getting ORA-01788 When Running a Query That Includes

The LEVEL Pseudo Column [ID 455953.1]

## 2.27 PGA AGGREGATE LIMIT

Recommended Value: 0

### Oracle FLEXCUBE Relevance

Setting this parameter limits the PGA consumed by the instance, hence might cause failure to few of the running processes.

## 2.28 Optimizer Adaptive Features

Property	Description
Parameter type	Boolean
Default value	True
Modifiable	ALTER SESSION, ALTER SYSTEM
Recommended value	False

### Oracle FLEXCUBE relevance

Some of Oracle FLEXCUBE sql statements are generated dynamically. So they contain literal values in the WHERE clause conditions. This results in large numbers of nearly identical statements with separate parse trees in Oracle's library cache, which can slow performance and cause latch problems. Also this can lead to a wrong explain plan.

## 2.29 Parallel Adaptive Multi User

Property	Description
Parameter type	Boolean
Default value	True
Modifiable	ALTER SESSION, ALTER SYSTEM
Recommended value	False

### Oracle FLEXCUBE relevance

When set to true enables an adaptive algorithm designed to improve performance in multiuser environments that use parallel execution.

## 2.30 Memory Target

Property	Description
Parameter type	Integer
Default value	0
Modifiable	ALTER SYSTEM
Recommended value	0

### Oracle FLEXCUBE relevance

It specifies the Oracle system-wide usable memory. As the database tunes the memory either reducing or increasing SGA and PGA default value is retained. If value is explicitly set, it will negatively impact the performance.

## 2.31 Memory Max Target

Property	Description
Parameter type	Integer
Default value	0
Modifiable	ALTER SYSTEM
Recommended value	0

## Oracle FLEXCUBE relevance

It specifies the Oracle system-wide usable max memory. If value is explicitly set, it should be greater than Memory\_Target.

## 2.32 Transparent Database Encryption

Transparent Data Encryption (TDE) enables you to encrypt data so that only an authorized recipient can read it. Use encryption to protect sensitive data in a potentially unprotected environment, such as data you placed on backup media that is sent to an off-site storage location. You can encrypt individual columns in a database table, or you can encrypt an entire tablespace.

To use Transparent Data Encryption, you do not need to modify your applications. TDE enables your applications to continue working seamlessly as before. It automatically encrypts data when it is written to disk, and then automatically decrypts the data when your applications access it. Key management is built-in, eliminating the complex task of managing and securing encryption keys.

To enable TDE,

Login to SYS user in Container Database

### Creating a Password-Based Software Keystore

Compile the below command to create password protected keystore.

```
ADMINISTER KEY MANAGEMENT CREATE KEYSTORE '<keystore_location>' IDENTIFIED BY  
<key_store_password>;
```

In the above command,

**keystore\_location:** keystore\_location is the path to the keystore directory location of the password-based keystore for which you want to create the auto-login keystore

**key\_store\_password:** password of the keystore that you, the security administrator, creates.

Above command creates, ewallet.p12 file in the keystore\_location

### Open the Password-protected Keystore

Run the below command to open the Password protected keystore.

```
ADMINISTER KEY MANAGEMENT SET KEYSTORE OPEN IDENTIFIED BY <key_store_password>  
[<CONTAINER = ALL | CURRENT>];
```

In the above command,

**CONTAINER:** Enter **ALL** to set the keystore in all of the PDBs in this CDB, or **CURRENT** for the current PDB.

### Creating Master Encryption Key

Creating and managing the TDE master encryption key can be done for password protected keystore either from the root or the PDB (SYS user of corresponding PDB)

Login to SYS user of either CDB to PDB to create the key

**ADMINISTER KEY MANAGEMENT SET KEY IDENTIFIED BY <key\_store\_password> with BACKUP [using '<Key\_backup>'];**

In the above command,

*Key\_backup*: This identifier is appended to the named keystore file

When compiled in PDB, TDE gets enabled in the PDB.

### **Creating Auto-Login Keystore**

Alternatively, auto-login software keystore can be created for Password protected keystore by following below commands

**ADMINISTER KEY MANAGEMENT CREATE [LOCAL] AUTO\_LOGIN KEYSTORE FROM KEYSTORE '<keystore\_location>' IDENTIFIED BY <key\_store\_password>;**

In the above command,

**LOCAL** enables you to create a local auto-login software keystore. Otherwise, omit this clause if you want the keystore to be accessible by other computers.

*keystore\_location*: Password protected Keystore location

*key\_store\_password*: Password protected Keystore password

The auto-login software keystore can be opened from different computers from the computer where this keystore resides, but the local auto-login software keystore can only be opened from the computer on which it was created. Both the auto-login and local auto-login keystores are created from the password-based software keystores.

Note: Transparent Data Encryption uses an auto login keystore only if it is available at the correct location (**ENCRYPTION\_WALLET\_LOCATION**, **WALLET\_LOCATION**, or the default keystore location), and the SQL statement to open an encrypted keystore has not already been executed. (Auto-login keystores are encrypted, because they have system-generated passwords.)

### **Creating Tablespace with ENCRYPTION.**

After opening the keystore and setting Management Key add the below clause to CREATE TABLESPACE command to create with ENCRYPTION

**ENCRYPTION USING 'AES256' ENCRYPT**

Above command specifies the encryption algorithm and the key length for the encryption. The **ENCRYPT** clause encrypts the tablespace. Enclose this setting in single quotation marks (' '). The key lengths are included in the names of the algorithms. If you do not specify an encryption algorithm, then the default encryption algorithm, **AES128**, is used. For IS Installation **AES256** should be mentioned as algorithm.

---

## 3. Redo Log Files

The default redo log files groups and size is inadequate to run Oracle FLEXCUBE. Hence, the recommended are:

- 6 redo log groups
- Redo log file size
  - 1 GB each for the DB size up to 1 TB
  - 2 GB each for DB size more than 1 TB

---

## 4. PLSQL Optimizer Level

The `plsql_optimize_level` value for all the pl/sql units should be same which would be the value set in `plsql_optimize_level` init parameter.

Following sql gives the PLSQL optimizer level for Oracle FLEXCUBE schema plsql units:

```
Select PLSQL_OPTIMIZE_LEVEL,type,count(*) "Count" from  
user_plsql_object_settings group by PLSQL_OPTIMIZE_LEVEL,type;
```

PLSQL\_OPTIMIZE\_LEVEL for all the objects should be same which should be value set in `plsql_optimize_level` init parameter. If there is a difference then the objects should be recompiled. This can be done using `dbms_utility.compile_schema` procedure.

Eg: - `exec dbms_utility.compile_schema('FCISBM2')` Here, 'FCISBM2' refers to Oracle FLEXCUBE schema.

**Note:** The '`dbms_utility.compile_schema`' procedure invalidates and recompiles all the plsql units.

---

## 5. Oracle FLEXCUBE Database Storage Recommendations

Oracle database 10g release 2 onwards, Automatic Storage Management (ASM) is the recommended storage option for Oracle FLEXCUBE database. ASM is an integrated cluster aware volume manager and a file system designed and optimized for managing Oracle database files. ASM is the realization of the Oracle Stripe and Mirror Everything (SAME) storage management methodology researched and established as best practices for Oracle database environment over many years.

**Note:** For configuring ASM refer Automatic storage management best practice document provided by Oracle for your database version.

### 5.1 Key benefits of ASM

- I/O is spread evenly across all available disk drives to prevent hot spots and maximize performance.
- ASM eliminates the need for over provisioning and maximizes storage resource utilization facilitating database consolidation.
- Inherent large file support.
- Performs automatic online redistribution after the incremental addition or removal of storage capacity.
- Maintains redundant copies of data to provide high availability, or leverage 3rd party RAID functionality.
- Supports Oracle Database 12c as well as Oracle Real Application Clusters (RAC). Capable of leveraging 3rd party multipathing technologies.
- For simplicity and easier migration to ASM, an Oracle Database 12c database can contain ASM and non-ASM files. Any new files can be created as ASM files whilst existing files can also be migrated to ASM.
- RMAN commands enable non-ASM managed files to be relocated to an ASM disk group. Oracle Database 12c Enterprise Manager can be used to manage ASM disk and file management activities.

---

## 6. Oracle FLEXCUBE Database Backup Recommendations

Backup Policy is a very important ingredient of any High Availability system. Oracle recommends RMAN utility for database backup.

RMAN is acronym for Recovery Manager, is Oracle utility which will backup, restore, and recover oracle data files. RMAN is an Oracle provided utility for efficiently performing Backup and Recovery. RMAN is available as a part of the standard Installation and no separate installation is required.

Recovery Manager is a client/server application that uses database server sessions to perform backup and recovery. It stores metadata about its operations in the control file of the target database and, optionally, in a recovery catalogue schema in an Oracle database.

You can invoke RMAN as a command-line executable from the operating system prompt or use some RMAN features through the Enterprise Manager GUI.

### 6.1 RMAN Vs Conventional Backup

During a conventional hot backup, the amount of Redo generated during the backup would be more due to the fact that the redo logs during the hot backup store the entire block images rather than the change vectors.

RMAN doesn't place the tablespace in a backup mode and hence the amount of Redo generated during the RMAN backup is considerably low.

RMAN can identify block corruption during backup operations and RMAN supports Block recovery.

RMAN automatically detects new data files and will backup them. Also, RMAN supports incremental backup method.

RMAN backs up only the blocks that have been used at least once. Unused blocks are never backed up. Unused block here refers to the blocks where in the block header is zeroed

RMAN enables us to test the backup without actually restoring the backup.

RMAN can verify physical and logical structures of the database without actually performing backup.

Usage of Shared Pool and Large Pool for RMAN

RMAN uses DBMS\_RCVMAN and DBMS\_BACKUP\_RESTORE packages for backup and recovery. These packages would be loaded in the shared pool for backup and restore operation. RMAN uses the PGA for backup and restore operation.

RMAN Requires LARGE\_POOL only if TAPE\_IO\_SLAVES and DBWR\_IO\_SLAVES are defined.

Sizing Large Pool -  $LARGE\_POOL = (Number\ of\ Channels) * (16\ MB + Tape\ Buffer)$



## **6.2 Benefits of Using RMAN**

RMAN is an intelligent tool that comes at no extra cost. It is available free with the Oracle Database.

RMAN introduced in Oracle 8 it has become simpler with newer versions and easier than user managed backups.

Provides proper security for Backups

You can be 100% sure your database has been backed up.

Controlfile and Spfile of the database can be configured to be automatically backed up by RMAN. It contains detail of the backups taken etc in its central repository Facility for testing validity of backups also commands like crosscheck to check the status of backup. Faster backups and restores compared to backups without RMAN.

RMAN is the only native backup tool which supports incremental backups.

Oracle 12c has got further optimized incremental backup which has resulted in improvement of performance during backup and recovery time.

Parallel operations (Multiple Channels for Backup and Restore) are supported. Better querying facility for knowing different details of backup.

No extra redo is generated when backup is performed, compared to conventional online backup. Maintains repository of backup metadata.

Remembers backup set location. Knows what need to backed up. Knows what is required for recovery. Knows what backup are redundant.

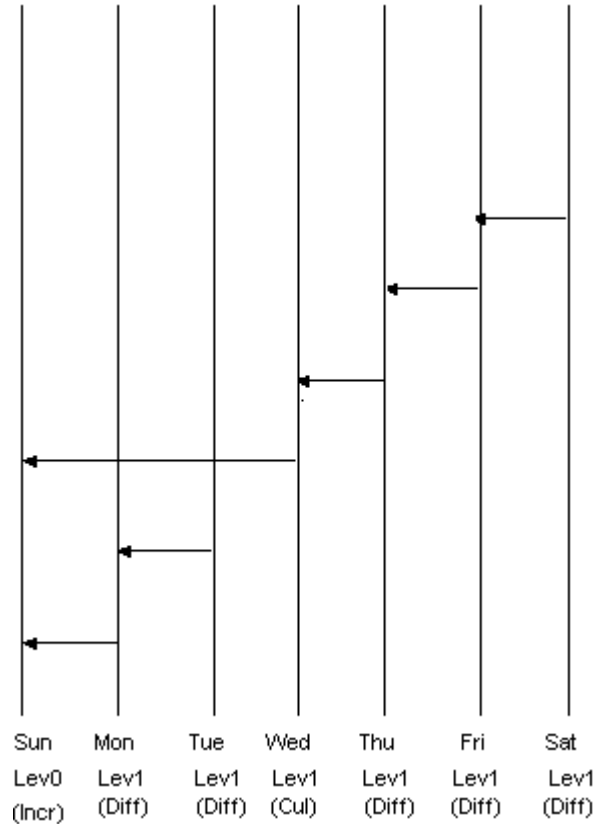
RMAN can back up the Database to Disk or directly to Tape. It is recommended that RMAN backup is performed to disk and then copied to tape.

## 6.3 Backup Strategy Recommendation

RMAN will not backup the below files so it is advised to take the copy of the below files on regular basis (weekly/any change/addition to the file).

Tnsnames.ora Listener.ora Password file Init.ora

The Best practice is to take create the pfile once the spfile is updated. Below is the recommended strategy.



---

## 7. Tablespace Layout and Moving Tables to Respective Tablespaces

Oracle FCIS Tables and Indexes are placed in Default Tablespace.

Oracle FCIS tables and indexes can also be placed in corresponding tablespaces according to their usage. i.e. heavily populated tables and corresponding indexes can be placed in tablespaces with higher extent size whereas the maintenance tables where the data population is less can be placed in a tablespace with smaller extent size. This avoids frequent space allocation in turn improve the performance.

For example table PERIODICLOADACCRUALHISTORY is heavily populated hence this table and its indexes can be placed in tablespace FCISDATAXL and FCISINDXXL respectively where extent size is high. The table funddemographicstbl where data population is less, data and its indexes are placed in tablespace FCISDATASML and FCISINDXSML respectively which is having smaller extent size.

Setting up of FCIS tables and indexes to corresponding tablespace is optional.

Oracle FLEXCUBE Tablespaces can be as follows:

Tablespace name	Tablespace type	Extent management	Segment space management
FCISDATASML	DATA	LOCAL	AUTO
FCISINDXSML	INDEX	LOCAL	AUTO
FCISDATAMED	DATA	LOCAL	AUTO
FCISINDEXMED	INDEX	LOCAL	AUTO
FCISDATALAR	DATA	LOCAL	AUTO
FCISINDEXLAR	INDEX	LOCAL	AUTO
FCISDATAXL	DATA	LOCAL	AUTO
FCISINDXXL	INDEX	LOCAL	AUTO
FCISDFLT	AD HOC	LOCAL	AUTO

Note: Tablespaces extent size depends on Oracle FLEXCUBE implementation (i.e. Small, Medium and Large). So these parameters are to be derived using base lined excel [FLEXCUBE-Disk-Layouts-initparams-12c.xlsm](#) based on implementation.

Sample script to move table and index:

```
Alter table PERIODICLOADACCRUALTBL move tablespace FCISDATALAR;
```

```
Alter index PERIODICLOADACCHISTXN_IDX2 rebuild tablespace FCISINDXLAR;
```

Similarly all tables and indexes should be moved to respective tablespaces.

## 8. Table and Index Partitioning

Table and index partitioning helps to reduce the contention and GC related delays in RAC environment. Table and index partitioning is mandatory if you have deployed Oracle FCIS in RAC database.

Table Partitioning:

Following are the list of tables to be partitioned:

Table_name	Column_name	Partitioning_type
CONSOLIDATEDTXNTBL	FUNDID	list
TXNSETTLEMENTTBL	txn number	hash (16partitions)
UHBALLEDGERTBL	fundid	List
AGEINGTBL	TRANSACTIONNUMBER	HASH (16partitions)
ALLOCATIONTBL	TRANSACTIONNUMBER	HASH (16partitions)
POLICYTXNTBL	policytxnnumber	Hash (16partitions)
TXNINTERMEDIARYTBL	TRANSACTIONNUMBER	Hash (16partitions)
TXNDETAILSTBL	TRANSACTIONNUMBER	hash (16partitions)
LOADHISTORYTBL	fundid	List
PERIODICLOADLOGTBL	fundid	List
POLICYTXNFUNDRATIOTBL	policytxnnumber	Hash (16partitions)
POLICYFUNDTXNLINKTBL	policytxnnumber	Hash (16partitions)
CLEARINGTBL	transactionnumber	Hash (16partitions)
DERIVEDLOADHISTORYTBL	fundid	List
TXNVATVALUESTBL	transactionnumber	Hash (16partitions)
UNITHOLDERTBL	Unitholderid	Hash (16partitions)
UHINTERMEDIARYTBL	Unitholderid	Hash (16partitions)
DIVIDENDUHSPLITLOGTBL	KEYSTRING	HASH

Index Partitioning:

Following are the list of tables whose indexes were partitioned:

Table_name	Index_name	Global / Partitioned Index
DIVIDENDECLAREDETAILSTB	DIVDECTBLPK	Global
	DIVDECTBL_IDX41	Global
	DIVIDENDECLAREDETAILSTBLIDX2	Hash
	DIVIDENDECLAREDETAILSTBLIDX3	Hash
UHDIVIDENDBALANCETBL	UHDIVBALIDX_1	Hash
	UHDIVBALIDX_2	Hash
	UHDIVBALIDX_9	Hash
DIVIDENDDetailSTBL	DIVDETSTBLPK	Hash
DIVIDENDUHSPLITLOGTBL	DIVIDENDSPLITPK	Hash
	DIVIDENDUHSPLITLOGIDX_1	Global
FORCEDDIVMODECHANGETBL	FORCEDDIVMODECHGTBLIDX1	Hash
FUNDDIVIDENDCOMPONENTV	DIVIDX01	Hash
FUNDDIVIDENDPAYMENTTBL	FNDDIVPAYTBLPK	Hash
	FNDDIVPAYTBL_IDX1	Hash
MOCKUHDIVIDENDBALANCETB	MOCKUHDIVIDENDBALANCEIDX_1	RANGE
UHDIVIDENDEQPAYMENTTBL	EQPAYMENTPK	Hash
UHDIVIDENDNPIPAYMENTTBL	UHDIVNPIPAYTBLPK	Hash
UHDIVIDENDPAYMENTTBL	UHDIVIDENDPAYMENT1PK	Hash
	UHDIVIDENDPAYMENTIDX_1	Hash
	UHDIVIDENDPAYMENTTBL_IDX	Global
	UHDIVIDENDPAYMENTTBL_IDX1	Global
WHTDIVIDENDCOMPONENTST	WHTDIVCOMPTBL_PK	Hash
PERLOADACCTEMPSTBL	PERLOADACCT_IDX1	List
	PERLOADACCT_IDX2	List

---

## 9. Sequence Caching

Sequence Caching is applicable only if Oracle FCIS is deployed in RAC database.

Heavy use of sequences in RAC database causes high DFS lock handle & row cache lock waits which affect the application scalability. In order to overcome this issue, the sequences are to be cached with noorder option.

All the FCIS sequences should be recreated cache 500 and noorder. Steps to alter existing sequences as follows:

1. Login to FLEXCUBE schema
2. SQL > Spool sequence.sql
3. SQL > select ' alter sequence ' || sequence\_name || ' cache 500 noorder;' from user\_sequences;
4. SQL > spool off;
5. SQL > @ sequence.sql

Verify that cache and order changed to all sequences.

6. Select order\_flag, cache\_size from user\_sequences;

In FCIS some of the sequences are recreated as part of end of day batches. Those sequences have to be taken care in spRecreateSequences procedure. This procedure should be modified to include caching and noorder as follows:

```
l_Create := 'CREATE SEQUENCE ' || p_Seq_Name || ' INCREMENT BY 1 START WITH 1  
MINVALUE 1 NOCYCLE CACHE 500 NOORDER';
```

---

## 10. Statistics Collection for FCIS Schema (Recommended Method)

Oracle provides a default scheduled job to collect statistics for the entire database and is default scheduled to run every night. Given that Oracle FLEXCUBE batch as well runs in the night it is critical that the statistics gathering is not run during the batch.

It is recommended to use the default database scheduled job that is shipped with Oracle

Database to collect statistics for Oracle FLEXCUBE Schema

**Note:** This document assumes that there is no other tool or a program is scheduled to collect statistics for the Database.

### 10.1 Customizing Default Statistics Collection Schedule

The Default Scheduler is to be customized for the following:

- Ensure that the default statistics gathering program is configured and Running.

```
SELECT STATUS
```

```
FROM DBA_AUTOTASK_CLIENT
```

```
WHERE CLIENT_NAME='auto optimizer stats collection';
```

Should return - ENABLED

- Ensure that the default statistics gathering program is configured to run only on weekends.

```
/* Start of Script – Script to be executed as SYS*/
```

```
BEGIN DBMS_AUTO_TASK_ADMIN.ENABLE(
```

```
CLIENT_NAME => 'auto optimizer stats collection',
```

```
OPERATION => NULL,
```

```
WINDOW_NAME => 'SATURDAY_WINDOW'); DBMS_AUTO_TASK_ADMIN.ENABLE(
```

```
CLIENT_NAME => 'auto optimizer stats collection', OPERATION => NULL,
```

```
WINDOW_NAME => 'SUNDAY_WINDOW'); END;
```

```
/* End of Script */
```

- Default schedule is daily. So disable the daily schedules for optimizer statistics.

```
/* Start of Script – Script to be executed as SYS*/
```

```
BEGIN DBMS_AUTO_TASK_ADMIN.DISABLE(
```



```

CLIENT_NAME => 'auto optimizer stats collection', OPERATION=> NULL,
WINDOW_NAME => 'MONDAY_WINDOW'); DBMS_AUTO_TASK_ADMIN.DISABLE(
CLIENT_NAME =>'auto optimizer stats collection', OPERATION => NULL,
WINDOW_NAME => 'TUESDAY_WINDOW');
DBMS_AUTO_TASK_ADMIN.DISABLE(
CLIENT_NAME =>'auto optimizer stats collection', OPERATION => NULL,
WINDOW_NAME => 'WEDNESDAY_WINDOW');
DBMS_AUTO_TASK_ADMIN.DISABLE(
CLIENT_NAME => 'auto optimizer stats collection', OPERATION=> NULL,
WINDOW_NAME => 'THURSDAY_WINDOW'); DBMS_AUTO_TASK_ADMIN.DISABLE(
CLIENT_NAME => 'auto optimizer stats collection', OPERATION=> NULL,
WINDOW_NAME => 'FRIDAY_WINDOW');
END;
/
/* End of Script */

```

Verify the setup using the following SQL

```

SELECT WINDOW_NAME,OPTIMIZER_STATS FROM DBA_AUTOTASK_WINDOW_CLIENTS;
Should return

```

```

MONDAY_WINDOW DISABLED
TUESDAY_WINDOW DISABLED
WEDNESDAY_WINDOW DISABLED
THURSDAY_WINDOW DISABLED
FRIDAY_WINDOW DISABLED
SATURDAY_WINDOW ENABLED
SUNDAY_WINDOW ENABLED

```

## 10.2 Customizing Statistics Gathering for FCIS

The default statistics gathering is designed to be generic. It is recommended to customize the default statistics gathering to suit FLEXCUBE online and batch.

Following are the areas that would need customization for FLEXCUBE:

- Statistics Histograms
- Sample Size of Statistics

### 10.2.1 Statistics Histograms

**Note the following:**

- The default statistics gathering routine decides to collect histograms on specific tables based on certain criteria that are not documented.
- Statistics Histograms are not recommended for FLEXCUBE tables. Configure the default statistics gathered without Histograms.

```
/* Start of Script – Script to be executed as SYS*/
```

```
BEGIN
```

```
DBMS_STATS.SET_PARAM ('METHOD_OPT','FOR ALL COLUMNS SIZE 1'); END;
```

```
/
```

```
/*End of Script */ Verify the setup using
```

```
SELECT DBMS_STATS.GET_PARAM ('METHOD_OPT') FROM DUAL;
```

Should return

```
FOR ALL COLUMNS SIZE 1
```

### 10.2.2 Sample Size of Statistics

The default statistics gathering routine decides on the percentage of data sampling

(AUTO\_SAMPLE\_SIZE).

The idea of sampling is to reduce the time taken for collecting statistics. Sampling could be effective for very large historical tables but not for medium and small tables and hence sampling of data for all FCIS tables is not recommended

Configure the default statistics gathered with 100% data coverage.

```
/* Start of Script – Script to be executed as SYS*/ BEGIN  
DBMS_STATS.SET_PARAM('ESTIMATE_PERCENT',100); END;
```

```
/
```

/\* End of Script \*/

Verify the setup using

```
SELECT DBMS_STATS.GET_PARAM('ESTIMATE_PERCENT') FROM DUAL;
```

Should return 100

---

## 11. FCIS Periodic Table Maintenance

Find below the list of tables and the corresponding action that needs to be planned in a periodical basis so that no performance degradation is observed over a period of time. These activities need to be planned ahead since this would require downtime. Also ensure proper backups are taken prior to any table maintenance activity.

Details of the various Actions are as below:

Truncate Table: Take the backup of the current table data and truncate the table

Recreate Table: These tables and it corresponding indexes needs to be rebuild.

Table Name	Action
CSTB_MSG_LOG	Truncate Table
SMTB_IMAGE_UPLOAD	Truncate Table
STTB_RECORD_LOG	Recreate table
STTB_FIELD_LOG	Recreate table
PERIODICLOADACCRUALHISTORYTBL	Recreate table
CONSOLIDATEDTXNTBL	Recreate table
PERIODICLOADACCRUALTBL	Recreate table
PERIODICLOADHISTORYTBL	Recreate table
AGEINGTBL	Recreate table
POLICYTXNTBL	Recreate table
UHDIVIDENDBALANCETBL	Recreate table

For Maintenance Activity related to Truncate and Recreate table, the impact to be analyzed at site level before implementing the action. Also for any purging related solutions required, please refer the Section 2.1 Purge Frequency Maintenance (UserManuals/PDF/Admin/Admin.pdf).

---

## 12. FCIS Functionality Related Performance Changes

The following parameters are discussed on functionality related performance changes:

Parameter	Recommended Value	How to find
Real debug parameter	N	select param_val from cstb_param where param_name='REAL_DEBUG'
PROCESSORCOUNT	Based on the processor count	select * from paramstbl where PARAMCODE = 'PROCESSORCOUNT'; Value decides the parallel degree for EOD jobs
JOBCOUNT	Based on the processor count	select * from paramstbl where PARAMCODE = 'JOBCOUNT'; Value decides the parallel degree for EOD jobs
PARALLELLEVEL	2	select * from paramstbl where PARAMCODE = 'PARALLELLEVEL'. Value decides the parallel degree for Allocation batch

---

## 13. Appendix

### 13.1 Oracle 12c DB Benefits of consolidating application workloads

1. Simplified Management – Reduce the number of distinct environments to manage. Manage many as one [CDB / PDB].
2. Streamlined Provisioning and Patching
3. Standardization
  - Reduces complexity
  - Improves manageability

Please refer <https://docs.oracle.com/database/121/NEWFT/toc.htm> for more details.

### 13.2 Script to Check Histograms on FCIS Schema

Following script would have to be executed in Oracle FLEXCUBE schema:

```
select distinct
table_name from
(
select table_name from user_tab_columns where histogram!='NONE'
)
```

Should return No Records

### 13.3 Script to Remove Histograms on FCIS Schema

Following script would have to be executed in Oracle FLEXCUBE schema if there are any rows:

```
declare

cursor cur_tables is

select distinct
table_name from

(

select table_name from user_tab_columns where histogram!='NONE'

);

begin

for rec_tables in cur_tables loop

dbms_stats.gather_table_stats(ownname=>USER,tabname=>rec_tables

.t
able_name,METHOD_OPT=>'FOR ALL COLUMNS SIZE

1',CASCADE=>TRUE,DEGREE=>2,ESTIMATE_PERCENT=>NULL);

end loop;

end;
```



Database Practices Guide  
[May] [2018]  
Version 14.0.0.0.0

Oracle Financial Services Software Limited  
Oracle Park  
Off Western Express Highway  
Goregaon (East)  
Mumbai, Maharashtra 400 063  
India

Worldwide Inquiries:  
Phone: +91 22 6718 3000  
Fax: +91 22 6718 3001  
[www.oracle.com/financialservices/](http://www.oracle.com/financialservices/)

Copyright © 2007, 2018, Oracle and/or its affiliates. All rights reserved.

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

**U.S. GOVERNMENT END USERS:** Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are “commercial computer software” pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. 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 failsafe, 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.

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.

This software or hardware and documentation may provide access to or information on 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. 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.