

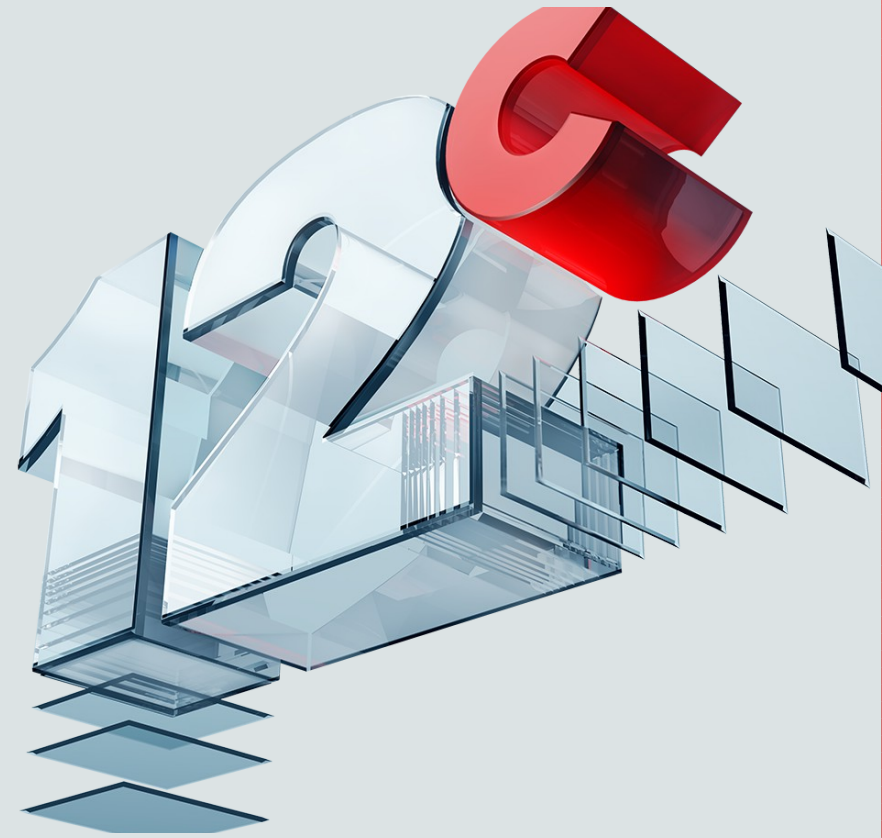
ORACLE®

ORACLE®

# Oracle Database 12c Release 2 New Features (OVERVIEW)

Rhoda Sarmiento-Pereira

Oracle Support



## Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

# Globalization Support

- AL32UTF8 is the default database character set
- Unicode 7.0 support
- Multitenant (CDB) support of multiple PDB (Pluggable database) with different character set.

# Globalization Support Enhancements – Column Collation and Case Insensitive Database

- Allows DBAs to declare collation (or sort ordering) in the column level
- The case insensitive collation can be specified in column level or object level or even schema level. And we can thus declare all the character columns in a database as case-insensitive by default.

## •How to use the feature?

- Prerequisite: **MAX\_STRING\_SIZE=EXTENDED** and **COMPATIBLE>=12.2**
- Collation can be specified in the session level using **DEFAULT\_COLLATION** parameter or specify **DEFAULT COLLATION** in the schema level. It can be specified in the object level using **DEFAULT COLLATION** clause or even use **COLLATE** clause for column level.

Resources:

Oracle® Database

Globalization Support Guide

12c Release 2 (12.2)

E49910-10

May 2017

Column level collation – case insensitive and accent insensitive  
=> data-bound collation  
VARCHAR, CHAR, LONG,  
NVARCHAR2, NCHAR, CLOB,  
NCLOB

Column Collation and Case Insensitive Database - 12.2 Feature (Doc ID 2198234.1)

# New SQL built-in Function to handle conversion errors

## **FUNCTION: VALIDATE\_CONVERSION**

– Determines whether a given input value can be converted to the requested data type.

- **How to use the feature?**

- `SELECT VALIDATE_CONVERSION(1000 AS BINARY_DOUBLE) FROM DUAL;`

Expected Result:

It returns 1 if a given expression can be converted to the specified data type, else it returns 0.

## **Documentation:**

[http://docs.oracle.com/database/122/SQLRF/VALIDATE\\_CONVERSION.htm](http://docs.oracle.com/database/122/SQLRF/VALIDATE_CONVERSION.htm)

# SQLPLUS

- Command “history”
- Tune query performance
  - SET PREFETCH
  - SET LOBPREFETCH (amount in bytes, 0-2gb => convert in bytes)
  - SET STATEMENTCACHE (need to be less than the value of open\_cursor parameter, value 0-32767)

# SCHEDULER: JOB Incompatibilities

- Incompatibility specifies incompatible jobs or programs where only one of the group can be running at a time.

## How to use the feature?

- Specify a job-level or program-level incompatibility by using the **CREATE\_INCOMPATIBILITY** procedure in the DBMS\_SCHEDULER package.
- Useful for prioritizing the jobs that too when the schedule clashes

## Documentation:

Oracle® Database

Administrator's Guide

12c Release 2 (12.2)

E49631-15

Section 28.2.12 Incompatibilities

```
BEGIN
dbms_scheduler.create_incompatibility(
  incompatibility_name => 'incomp',
  object_name => 'job1,job2,job3',
  enabled => true );
END;
/

constraint_level => JOB_LEVEL or
=> PROGRAM_LEVEL
```



## SCHEDULER: RESOURCE QUEUES

- We can control how many of a specified resource are available to a job.
- Jobs cannot execute until the required resources is available.
- If there are a resource limitation
  - You can define a resource and set its properties
  - Job definition – where you can specify which resources are required to run a job.
- **How to use the feature?**
  - create a resource by using the CREATE\_RESOURCE procedure in the DBMS\_SCHEDULER package.

```
BEGIN DBMS_SCHEDULER.CREATE_RESOURCE( resource_name => 'my_resource',  
                                     units => 3,  
                                     state => 'ENFORCE_CONSTRAINTS',  
                                     comments => 'Resource1' ) END;  
  
/
```

## SCHEDULER: IN-MEMORY JOBS

- In memory jobs can be created when many jobs needs to be created and run during a short period of time

### Types:

- Repeating in-memory \*
- One-time in-memory \*\*

### •How to use the feature?

- Use an attribute job\_style in dbms\_scheduler as IN\_MEMORY\_RUNTIME , IN\_MEMORY\_FULL
- IN\_MEMORY\_RUNTIME – For running multiple times and has repeat interval \*
- IN\_MEMORY\_FULL – For running once and discarded \*\*

```
BEGIN DBMS_SCHEDULER.CREATE_JOB (   job_name => 'my_immediate_job',
                                   program_name => 'fast_op',
                                   job_style => 'IN_MEMORY_FULL',
                                   enabled => true); END
```

**Test Case:** New Feature 12.2: How To Create In Memory Jobs (Doc ID 2170874.1)

# Identify Interconnected Databases and Trace SCN Flow Across Distributed Databases

- With this new feature, we would be able to identify the dblink connection(whether inward or outward) which resulted in the external SCN spike. A new view DBA\_EXTERNAL\_SCN\_ACTIVITY is created which would facilitate this.

- **How to use the feature?**

- Connect to the database as a user who can query the DBA\_EXTERNAL\_SCN\_ACTIVITY, DBA\_DB\_LINK\_SOURCES, and DBA\_DB\_LINK and then run the select on these views after observing the SCN jump.

Resource:

**Finding out External SCN jump using DBA\_EXTERNAL\_SCN\_ACTIVITY in 12.2 (Doc ID 2171090.1)**

# Cross-Platform Migration Support for Encrypted Tablespaces

- Allow cross-platform operations with encrypted tablespaces.
- **How to use the feature?**

- Example:

```
RMAN> SET PASSPHRASE ON IDENTIFIED BY encr_temp;  
RMAN> BACKUP  
  TO PLATFORM 'Solaris[tm] OE (64-bit)'  
  FORMAT '/tmp/xplat_backups/trans_ts.bck'  
  DATAPUMP FORMAT '/tmp/xplat_backups/trans_ts_dmp.bck'  
  TABLESPACE projects, tasks;
```

Documentation:

Backup and Recovery User's Guide 12c Release 2 (12.2)

Chapter 28 Transporting Data Across Platforms

## Multi-tenant

- PDBs per container increased from 252 to 4,096
- Memory resource prioritization in addition to CPU and I/O

## I/O Rate Limits for PDB

- This feature limits the rate of physical I/O issued by a pluggable database (PDB)
- can only be applied to a PDB, not to the multitenant container database (DCB) or non-CDB
- **How to use the feature?**
  - You can set the MAX\_IOPS initialization parameter and the MAX\_MBPS initialization parameter to limit the disk I/O generated by a PDB.
  - Use one or both of the following initialization parameters to limit the I/O generated by a particular PDB: The MAX\_IOPS initialization parameter limits the number of I/O operations for Background I/O, such as redo log writes or dirty buffer cache writes, are not limited. The limits are specified with the new PDB parameters, MAX\_IOPS and MAX\_MBPS.

```
ALTER SYSTEM SET MAX_IOPS = 1000 SCOPE = BOTH;
```

### Useful Notes:

- [DBMS\\_PROCESS \(Doc ID 2165358.1\)](#)
- [I/O Rate Limits for PDBs 12.2 New feature . \(Doc ID 2164827.1\)](#)

# Undo Mode for a CDB

- When local undo is enabled, each container has its own Undo tablespace for every instance in which it is open. Otherwise, there is one Undo tablespace for the entire CDB.

- **How to use the feature?**

- Local Undo mode can be enabled by executing the below statement in Upgrade mode:

```
ALTER DATABASE LOCAL UNDO ON;
```

RESOURCE:

**Undo Modes in 12.2 Multitenant Databases - Local and Shared Modes (Doc ID 2169828.1)**

# Oracle Enterprise Manager Database Express (EM Express) Resource Manager Support

- The EM Express now has the ability to create, delete, edit, activate, and deactivate resource manager plans for both multitenant container databases (CDBs) and non-CDB databases.

## •How to use the feature?

- From the Resource Management page in EM Express for a non-CDB, CDB, or PDB, the Resource Manager features can be accessed.

RESOURCE:

**EM Express: Single Port Access for Multitenant - 12.2 New Feature (Doc ID 2170770.1)**



# Restart Redefinition From Failure

- Online redefinition can restart from most points of failure, eliminating the need to redo the entire operation.

- **How to use the feature?**

- Two test scenarios:

- 1) redefining a table to a tablespace on which the table owner has no predefined quota.
- 2) redefining a table to a tablespace that has no enough space.

- Check DBA\_REDEFINITION\_STATUS
  - Y – correct the error and restart
  - N – abort the redefinition operation

```
SELECT BASE_TABLE_NAME,  
       INTERIM_OBJECT_NAME,  
       OPERATION,  
       STATUS,  
       RESTARTABLE,  
       ACTION  
FROM DBA_REDEFINITION_STATUS;
```

# Rollback Redefinition

- After online table redefinition, someone can roll back the table to its definition before online table redefinition while preserving all data manipulation language (DML) changes made to the table.

- **How to use the feature?**

- Full test case can be found in (Doc ID 2164079.1).

- To use the ROLLBACK procedure, online table redefinition rollback must be enabled during online table redefinition. If you decide to retain the changes made by online table redefinition, then you can run the ABORT\_ROLLBACK procedure.

- `ENABLE_ROLLBACK = TRUE` → `DBMS_REDEFINITION.START_TABLE_REDEF => ROLLBACK` proc

- `FINISH_REDEF_TABLE => ROLLBACK` (view)

- `SYNC_INTERIM_TABLE` proc – synchronizes the interim table periodically to apply DML changes to the redefined table to the interim table.

Master Note: Overview of Online Redefinition of Tables (DBMS\_REDEFINITION) (Doc ID 1357825.1)

- Online Redefinition Progress Monitoring view `V$ONLINE_REDEF` (Doc ID 2185967.1)

# In-memory Enhancement

- Faster in-memory joins
  - Joins occur on dictionary rather than data
  - Join Group specifies columns used to join tables
    - Columns share compression dictionary
- Dataguard is now supported
  - Enables real-time analytics on standby with no impact on production database
  - Makes productive use of standby database resources
- In-Memory Fast-start
- IM column format persisted to storage
  - In-Memory column store contents checkpointed to secure file lob on populate
  - When DB restarts, population is faster as population process reads the column format directly from storage
  - Faster restore (2-5x) of column store since no need to reformat data

# Oracle Dataguard Database Compare

- compares data blocks stored in Oracle Data Guard Primary database and its physical databases.
- Use for disk errors (dbverify cannot do)

DBMS\_DBCOMP.DBCOMP

- v\$session\_longops data dictionary view status (activity).

## Performance

- New Index High compression enables further compression of indexes
  - Default in 12.2
  - Compatible=12.2
- Significant space savings for index heavy applications (i.e. OLTP)
- Index compression can also result in I/O improvements as a higher proportion of index can be cached
- => All\_indexes => compression column (advanced high, advanced low, disabled, or enabled).
- Advanced index compression is not supported for bitmap indexes or index-organized tables.

# Security

- 12.2 introduces online encryption of existing tablespaces with TDE tablespace live conversion
  - Also supports online re-key and decrypt.
  - Zero downtime
  - Full encryption of internal database structures i.e. SYSTEM, SYSAUX, and UNDO tablespaces

A complete documentation is found in Oracle Database Security Guide, Chapter 3 Configuring Transparent Data Encryption

# Pluggable Databases Ease-of-Adoption – PDB Character set

- When the character set of the root is AL32UTF8, PDBs that are plugged into the CDB or cloned can have a different character set than the root. PDBs that are created from the seed inherit the AL32UTF8 from it.
- When the character set of the root is not AL32UTF8, all of the PDBs in the CDB use the character set of the root.

## BENEFIT:

- In 12cR1 release, the character set of the PDB had to be binary-compatible with that of the CDB, which is a barrier to consolidation. This restriction is removed with this new feature.

## •How to use the feature?

- During Plug –in.
- For PDBs plugged in directly to CDB, plug in succeeds if:
  - If the database character set of the PDB candidate is the same as that of the CDB
  - If the database character set of the CDB is AL32UTF8. [In this case, plug-in succeeds regardless of the database character set of the PDB.]

## Resources:

- Feature Documentation Summary
- Documentation



*ORA*

*Rhoda.Sarmiento@oracle.com*



# **Hardware and Software Engineered to Work Together**

# **Hardware and Software Engineered to Work Together**

ORACLE®