

ORACLE®

ORACLE®

# Oracle Database 12c Release 1 CoreTech Seminar

## Upgrade / DB Admin / Utility

日本オラクル株式会社

田島紀幸

2013/08/23

ORACLE®  
DATABASE 12<sup>c</sup>



Plug into the **Cloud**.

以下の事項は、弊社の一般的な製品の方向性に関する概要を説明するものです。また、情報提供を唯一の目的とするものであり、いかなる契約にも組み込むことはできません。以下の事項は、マテリアルやコード、機能を提供することをコミットメント(確約)するものではないため、購買決定を行う際の判断材料になさらないで下さい。オラクル製品に関して記載されている機能の開発、リリースおよび時期については、弊社の裁量により決定されます。

Oracle と Java は、Oracle Corporation 及びその子会社、関連会社の米国及びその他の国における登録商標です。文中の社名、商品名等は各社の商標または登録商標である場合があります。

# Agenda

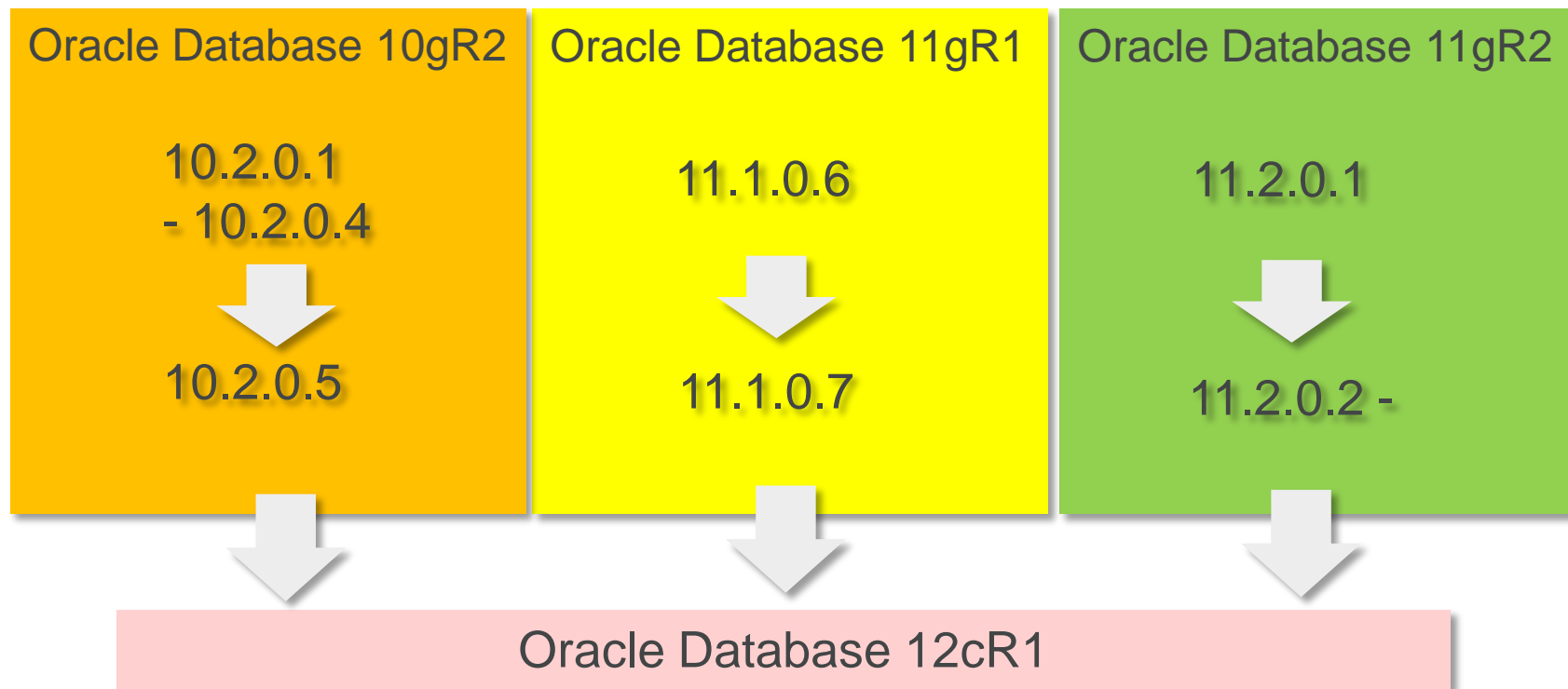
- アップグレード
- 追加された初期化パラメータ、バックグラウンド・プロセス
- ユーティリティ

# Agenda

- アップグレード
  - 基本情報
  - 非互換情報
  - アップグレード
    - アップグレード前情報ツール
    - Database Upgrade Assistant (DBUA) の変更点
- 追加された初期化パラメータ、バックグラウンド・プロセス
- ユーティリティ

# アップグレード・パス

直接アップグレード可能なバージョンは、10.2.0.5/11.1.0.7/11.2.0.2-



# クライアントとサーバーの互換性情報

NOTE:207303.1 Client / Server / Interoperability Support Matrix For Different Oracle Versions

クライアント・バージョン	サーバー・バージョン			
	12.1.0	11.2.0	11.1.0	10.2.0
12.1.0	○	○	△	△*
11.2.0	○	○	△	△*
11.1.0	△	△	△	△*
10.2.0	△*	△*	△*	△

△ : Extended Support(ES) 契約を有する場合のみ不具合修正が可能

\*: 11.1 以上 から 10.2 ヘデータベース・リンクを利用する PL/SQL を使用するため場合、  
10.2.0.2 以上を利用する必要があります。

# 対応プラットフォーム情報

現在までにリリースされてるもの

- Linux x86-64
  - Oracle Linux 6
  - Oracle Linux 5 Update 6
  - Red Hat Enterprise Linux 6
  - Red Hat Enterprise Linux 5 Update 6
  - SUSE Linux Enterprise Server 11 SP2
- Oracle Solaris on SPARC (64-bit)
  - Oracle Solaris 11 SRU 7.5
  - Oracle Solaris 10 Update 10
- Oracle Solaris on x86-64 (64-bit)
  - Oracle Solaris 11 SRU 7.5
  - Oracle Solaris 10 Update 10
- Windows x64 (64-bit)
  - Windows Server 2008 x64
  - Windows Server 2008 R2 x64
  - Windows Server 2012 x64 ※
  - Windows 7 x64
  - Windows 8 x64

詳細な情報は My Oracle Support の動作保障タブより確認



# アップグレード前に適用が必要なパッチ

- Oracle Grid Infrastructure 11g Release 2 (11.2.0.2)
  - 11.2.0.2.3(Patch:12539000 )以降の PSU を適用
- Oracle RAC 11g Release 1 (11.1.0.7)
  - 11.1.0.7.7 (Patch:11724953 )の CRS PSU を適用
- Oracle RAC 10g Release 2 (10.2.0.5)
  - Patch:8350262 と Patch:14617909 を適用

※アップグレード前に CVU を実行し、適用が必要なパッチを確認

# 非互換情報(一部)

- サポートが終了した機能
  - Oracle Enterprise Manager Database Control
  - RAW デバイス
  - Oracle Change Data Capture (CDC)
- 非推奨の機能
  - Oracle Streams
  - Oracle Database Advanced Replication
  - Server Control Utility の一文字オプション
  - Oracle Restart
- ツールの導入や統合
  - catupgrd.sql の実行方法変更
  - deinstall ツールのインストール・メディア統合
- デフォルト動作の変更
  - DB\_SECUREFILE パラメータ
  - PARALLEL\_MIN\_SERVER パラメータ
  - PRE\_PAGE\_SGA パラメータ
  - XMLDB のインストール
  - RAC 環境での Direct NFS

参考: Oracle Database Upgrade Guide 12c Release 1 (12.1)  
8 Deprecated and Desupported Features for Oracle Database 12c

# サポートが終了した機能

## Oracle Enterprise Manager Database Control

- Oracle Enterprise Manager Database Control のサポートが終了
  - Oracle Database 12c では利用不可
    - アップグレード時に DB Control で利用していたリポジトリは削除
  - EM Express (Oracle Enterprise Manager Database Express) が導入
    - Oracle Database 12c のインストールに含まれ、別途インストールは不要
  - Oracle Enterprise Manager Cloud Control も Oracle Database 12c に対応
    - 対応する Plug-in リリース済
    - 個別にインストールする必要あり

# サポートが終了した機能

## RAW デバイス、Oracle Change Data Capture (CDC)

- RAW デバイスのサポートが終了
  - RAW デバイス上に直接データファイル (OCR や Voting disk 含) を配置している環境を、Oracle Database 12c 以降にアップグレードする場合、Oracle ASM 等への移行が必要
- Oracle Change Data Capture (CDC) のサポートが終了
  - Oracle Database 12c には CDC は含まれていません。
  - Oracle GoldenGate で代用可能なため、Oracle GoldenGate の利用を検討ください
    - ※ Oracle GoldenGate は、Oracle Database とは別製品です

# 非推奨の機能

## Oracle Streams、Oracle Database Advanced Replication

- Oracle Streams の非推奨
- Oracle Database Advanced Replication の非推奨
  - Oracle Database 12c にこれらの機能は含まれており、現状機能の利用可能およびサポートも継続
  - Oracle GoldenGate でも同様の処理が実施可能なため、Oracle GoldenGate の利用を検討ください
  - ※ Oracle GoldenGate は、Oracle Database とは別製品です

# 非推奨の機能

## Server Control Utility (SRVCTL) の一文字オプション、Oracle Restart

- Server Control Utility (SRVCTL) の一文字オプションが非推奨
  - GNU スタイルの長いオプションをサポート

```
srvctl start instance -db orcl -instance orcl1
```

- これまでのリリースで利用可能な一文字オプションは非推奨

```
srvctl start instance -d orcl -i orcl1
```

- Oracle Restart の非推奨
  - Oracle Database 12c でも継続して構成、利用可能
  - 必要に応じて RAC One Node 構成などへの移行を検討ください

# ツールの導入や統合

## catupgrd.sql の実行方法変更、deinstall ツールのインストール・メディア統合

- catupgrd.sql の実行方法変更

- アップグレード・ユーティリティ (catctl.pl) を利用して実行

```
$ cd $ORACLE_HOME/rdbms/admin  
$ $ORACLE_HOME/perl/bin/perl catctl.pl catupgrd.sql
```

- deinstall ツールのインストール・メディア統合

- インストール・メディアが統合され別メディアを利用する必要なし
- 従来通り \$ORACLE\_HOME/deinstall 配下の deinstall の実行可能
- runInstaller を利用した deinstall の実行も可能

```
$ ./runInstaller -deinstall -home <ORACLE_HOMEのパス>
```

# デフォルト動作の変更

## DB\_SECUREFILE パラメータ

- Oracle Database 12c より、COMPATIBLE に依存してデフォルトが変化
  - LOB ファイルを SecureFiles として処理するか制御するパラメータ
  - PREFERRED
    - COMPATIBLE = 12.0.0.0 以上
    - STORAGE 句に BASICFILE を指定しない場合、SecureFiles として作成
  - PERMITTED
    - COMPATIBLE = 11.2.0.1 or 11.2.0.2 or 11.2.0.3
    - これまでと同様に STORAGE 句がない場合、BasicFiles として作成
    - STORAGE 句に SECUREFILE を指定すれば、SecureFiles の作成可能



# デフォルト動作の変更

## PARALLEL\_MIN\_SERVER パラメータ

- PARALLEL\_MIN\_SERVERS パラメータのデフォルト値変更
    - インスタンスの起動時に起動するパラレル・プロセス数を指定
    - Oracle Database 12c より
      - $\text{CPU\_COUNT} \times \text{PARALLEL\_THREADS\_PER\_CPU} \times 2$ 
        - ※ PARALLEL\_THREADS\_PER\_CPU は、プラットフォーム依存 (通常 2)
        - ※ CPU 数に応じて、プロセス数が増加
    - Oracle Database 11g まで 0
- ※ パラレル・プロセスが必要以上に起動するような場合は、明示的にパラメータを調整

# デフォルト動作の変更

## PRE\_PAGE\_SGA パラメータ、XMLDBのインストール

- PRE\_PAGE\_SGA パラメータのデフォルト値変更
  - インスタンスの起動時に SGA 全体をメモリーに読み込むかどうかを制御
  - Oracle Database 12c より TRUE (これまでは FALSE)
  - インスタンス起動に時間がかかる場合や、プロセスの作成・破棄が繰り返されるようなアプリケーションでは、パフォーマンスに影響が出る場合もある
- XMLDB のインストール
  - Oracle Database 12c より XMLDB が必須コンポーネント
  - アップグレード時に自動でインストール

# デフォルト動作の変更

## RAC環境でのdNFSのデフォルト有効化

- Oracle Database 12c より、RAC 環境では dNFS の設定がデフォルトで有効化
  - Oracle Database 11g までは、\$ORACLE\_HOME/rdbms/lib/ins\_rdbms.mk の make が必要

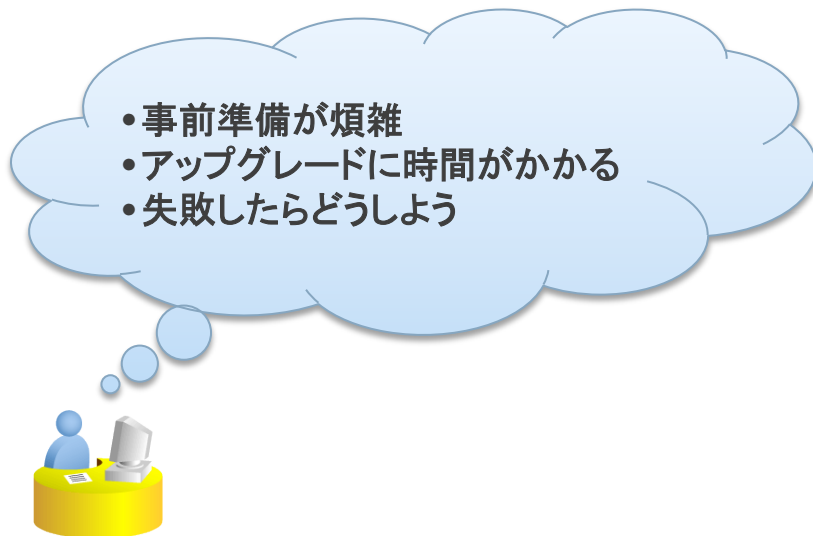
```
$make -f ins_rdbms.mk dnfs_on
```

- シングルインスタンス環境や Oracle Restart 環境では、make の実行が必要
- RAC 環境では、必要な設定は orafstab の作成のみ
- 無効化する場合は、ORACLE\_HOME と GRID\_HOME で以下のコマンドを実行

```
$make -f ins_rdbms.mk dnfs_off
```

# Oracle Database 12c へのアップグレード

## アップグレードに対するネガティブな認識



### アップグレード前情報ツール

- アップグレード前に対処が必要なレポートの作成
- 修正スクリプトの提供

### アップグレード・ユーティリティ

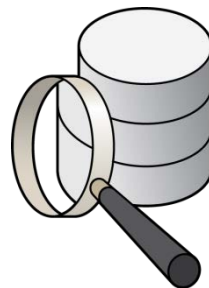
- アップグレード時の処理をパラレルで実行

### リカバリ・オプション

- Oracle Database のテクノロジーを利用したバックアップ、リカバリの実施

# アップグレード前情報ツール

## preupgrd.sql スクリプト



- 対象データベースの情報を取得し、修正が必要な項目や推奨事項をレポートを生成 **~ 12c**
  - 11.2 では、utlu112i.sql として提供
- 修正可能な問題を解決するためのスクリプトを生成 **12c New**
- \$ORACLE\_HOME/rdbms/admin 配下の preupgrd.sql と utluppkg.sql を利用
- レポートおよびスクリプトの出力先と名前
  - 出力先
    - \$ORACLE\_BASE/cfgtoollogs/db\_unique\_name/preupgrade/配下
  - レポート: preupgrade.log
  - スクリプト: preupgrade\_fixups.sql 、 postupgrade\_fixups.sql

# アップグレード前情報ツール

## preupgrd.sql スクリプト

### ■ 実行方法

1. 12.1 環境の \$ORACLE\_HOME/rdbms/admin 配下に配置されている preupgrd.sql、utluppkg.sql を、アップグレード対象のデータベースよりアクセス可能な場所に配置
2. 対象データベースの DBA権限を持ったユーザーで、preupgrd.sql を実行
3. 生成されたレポートおよび fixup スクリプトの内容を確認
4. 必要に応じて、fixup スクリプト をアップグレード対象のデータベースで実行
5. 手動で実行する必要のある問題に対応

必要な回数だけ、アップグレード前情報ツールを実行  
※再実行時、既存のレポート、スクリプトを上書き



# アップグレード前情報ツール

## スクリプト実行例(1/2)



```
$ sqlplus / as sysdba
SQL> @preupgrd.sql
Loading Pre-Upgrade Package...
Executing Pre-Upgrade Checks...
Pre-Upgrade Checks Complete.
*****

Results of the checks are located at:
/u01/app/oracle/cfgtoollogs/orcl/preupgrade/preupgrade.log

Pre-Upgrade Fixup Script (run in source database environment):
/u01/app/oracle/cfgtoollogs/orcl/preupgrade/preupgrade_fixups.sql

Post-Upgrade Fixup Script (run shortly after upgrade):
/u01/app/oracle/cfgtoollogs/orcl/preupgrade/postupgrade_fixups.sql

*****

Fixup scripts must be reviewed prior to being executed.

*****
```

生成されるログ  
およびスクリプトの  
出力先情報

# アップグレード前情報ツール

## スクリプト実行例(2/2)



- RECYCLE BIN に情報が残った状態でアップグレード前情報ツールを実行

```
*****
====> USER ACTION REQUIRED <====
*****

The following are *** ERROR LEVEL CONDITIONS *** that must be addressed
      prior to attempting your upgrade.
      Failure to do so will result in a failed upgrade.

1) Check Tag:      PURGE_RECYCLEBIN
   Check Summary:  Check that recycle bin is empty prior to upgrade
   Fixup Summary:
   "The recycle bin will be purged."

      You MUST resolve the above error prior to upgrade

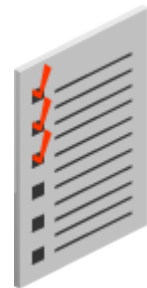
*****
```

アップグレード前に対  
処が必要な項目として、  
検知、対処方法を提示



# アップグレード前情報ツール

## 生成されるスクリプト



- 生成されるスクリプトの例: `preupgrade_fixups.sql` 抜粋

```
...
BEGIN
-- ***** Fixup Details *****
-- Name:          PURGE_RECYCLEBIN
-- Description:   Check that recycle bin is empty prior to upgrade
-- Severity:      Error
-- Action:        Fixup routine
-- Fix Summary:
--       The recycle bin will be purged.

dbms_preup.run_fixup_and_report('PURGE_RECYCLEBIN');
END;
/
...
```

- 具体的には、パッケージを利用して fixup を実施

# アップグレード前情報ツール

## fixup スクリプト実行例(抜粋)



```
SQL> conn / as sysdba
SQL> @preupgrade_fixups.sql
...
Beginning Pre-Upgrade Fixups...
..(略)..
*****
Check Tag:          PURGE_RECYCLEBIN
Check Summary: Check that recycle bin is empty prior to upgrade
Fix Summary:   The recycle bin will be purged.
*****
Fixup Succeeded
*****

PL/SQL procedure successfully completed.

..(略)..
```

- fixup スクリプトの実行により修正が成功したことを示す

# Database Upgrade Assistant (DBUA)

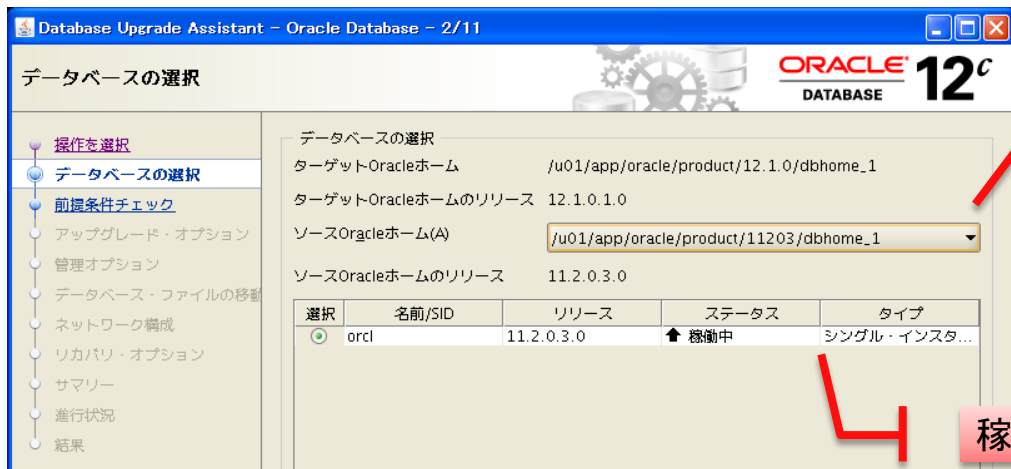
## 紹介内容一覧

- データベースの選択
- アップグレード・オプション
  - アップグレード並列度
  - タイムゾーン・データのアップグレード
  - アップグレード前の統計の収集
  - アップグレード中にユーザーの表領域を  
読取り専用に変更
  - ファイルの位置(診断先、監査ファイル  
保存先)
- ネットワーク構成
  - リスナーの選択、作成
- リカバリオプション
  - RMAN バックアップ
  - 保証付きリストア・ポイント
- 進行状況
  - アラート・ログ
  - アクティビティ・ログ
- アップグレード結果

# Database Upgrade Assistant (DBUA)

## データベースの選択

- 選択可能なデータベースとその情報が確認可能



アップグレード対象の  
Oracle ホームを選択

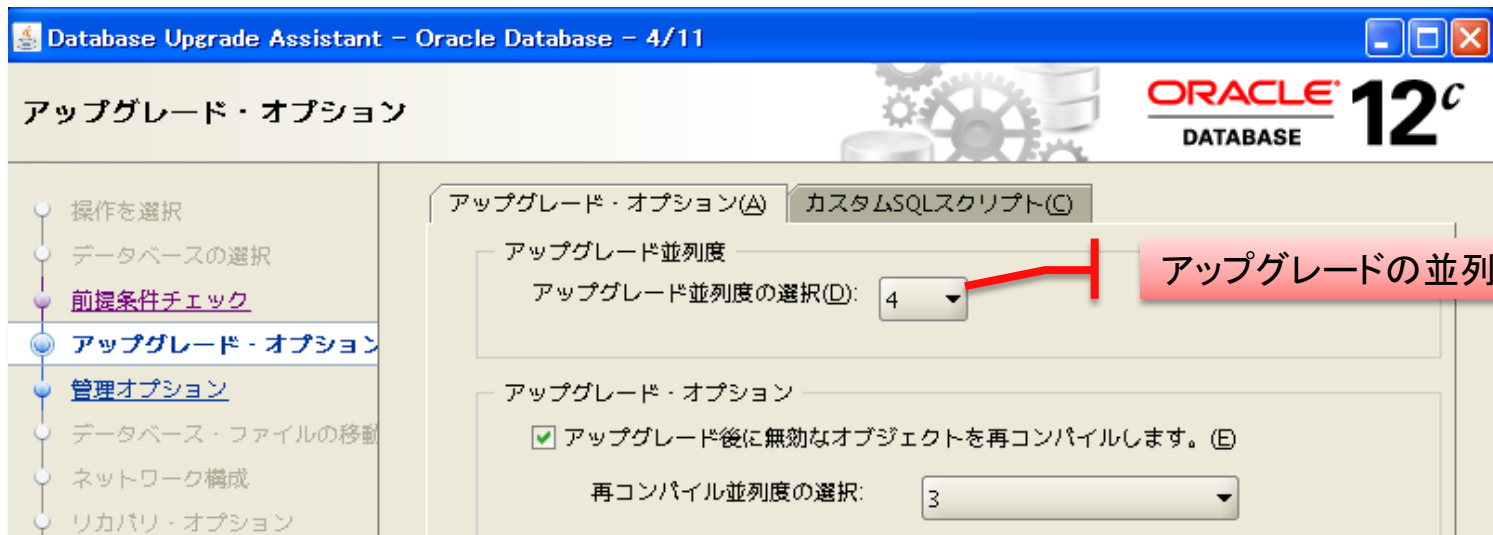
稼働状況やインスタンスのタイプの表示

- インスタンスタイプの表示
- 稼働状況の確認

# Database Upgrade Assistant (DBUA)

## アップグレード・オプション

- アップグレード・スクリプトを並列実行



- デフォルトは、CPU数。CPU数が4未満の場合は 2

# アップグレードの並列度補足

- アップグレード・ユーティリティ (catctl.pl) のオプションを利用し、手動でもパラレル実行可能
- 実行例

```
cd $ORACLE_HOME/rdbms/admin  
$ORACLE_HOME/perl/bin/perl catctl.pl -n 8 catupgrd.sql
```

-n X: パラレル度の設定  
例は8多重

- パラレル度に応じて、catupgrdX.log が生成
  - DBUA実行時には、下記のディレクトリに出力  
\$ORACLE\_BASE/cfgtoollogs/dbua/<SID>/upgrade<#>

# Database Upgrade Assistant (DBUA)

## アップグレード・オプション

アップグレード・オプション

アップグレード後に無効なオブジェクトを再コンパイルします。(E)

再コンパイル並列度の選択:

タイムゾーン・データのアップグレード(G)

アップグレード前の統計の収集(I)

アップグレード中にユーザーの表領域を読取り専用に変更(J)

ファイルの位置

診断先(K)  参照(L)

監査ファイル保存先(O)  参照(P)

タイムゾーン・データ対応

ディクショナリ・オブジェクトの統計情報を収集  
DBMS\_STATS.GATHER\_DICTIONARY\_STATS

これらオプションにより、データベースのアップグレード時間を短縮

# Database Upgrade Assistant (DBUA)

## ネットワーク構成

- リスナーの選択

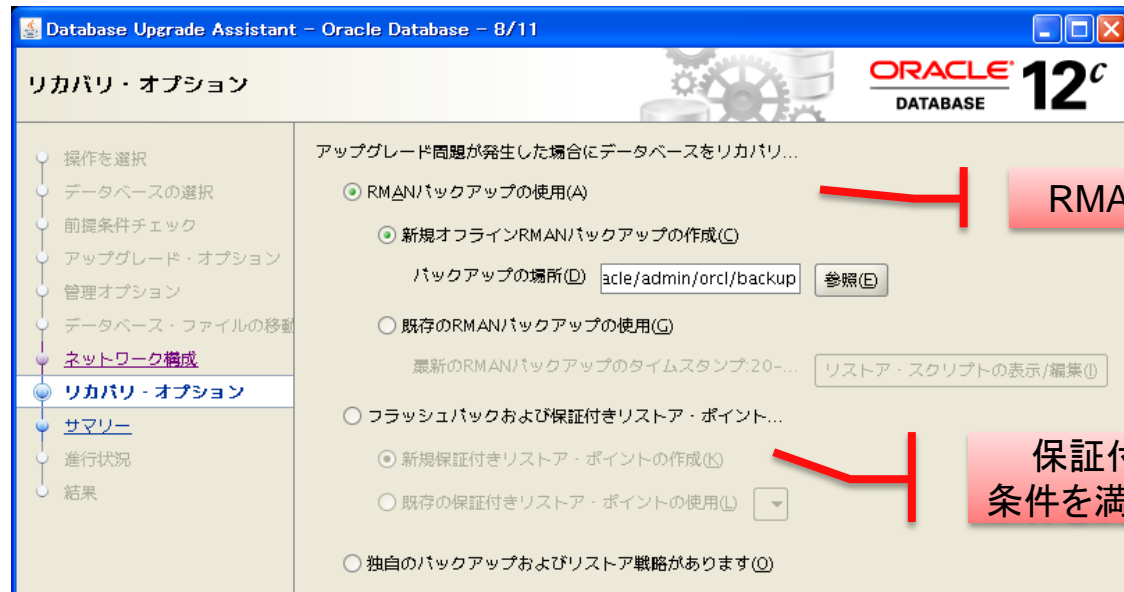
選択	名前	ポート	Oracleホーム	ステー...	移行
<input checked="" type="checkbox"/>	LISTENER	1521	/u01/app/oracle/product/12.1.0/dbhome_1	稼働中	いいえ
<input type="checkbox"/>			/u01/app/oracle/product/12.1.0/dbhome_1		いいえ

- リスナーの作成、登録、アップグレードが可能



# Database Upgrade Assistant (DBUA)

## リカバリ・オプション



RMANでのバックアップ取得

保証付きリストア・ポイントの利用  
条件を満たしていない場合は選択不可

- 事前に取得した RMAN バックアップや保証付きリストアポイントを選択することも可能

# Database Upgrade Assistant (DBUA)

## 進行状況

- アップグレード実行中のアラートログ、アクティビティ・ログをDBUA 経由で確認可能

クリックするとアラート・ログ、  
アクティビティ・ログを別ウィンドウで表示

- アクティビティ・ログは、以下のディレクトリのログ

\$ORACLE\_BASE/cfgtoollogs/dbua/<SID>/upgradeX/Oracle\_Server.log

進行状況

Oracle Databaseのアップグレード処理中...

0%

Oracle Databaseリリース11.2.0.3.0 (/u01/app/oracle/product/11203/dbhome\_1)からOracle Databaseリリース12.1.0.1.0 (/u01/app/oracle/product/12.1.0/dbhome\_1)にデータベースorclをアップグレードしています。

	ステップ	時刻	ステータス
🕒	➡ アップグレード前の処理		処理中
	➡ データベース・アップグレードの処理		
	➡ アップグレード後の処理		

アクティビティ・ログ(A)    アラート・ログ(B)    停止(C)

# Database Upgrade Assistant (DBUA)

## アップグレード結果

- ソースとターゲットの情報を出力
- アップグレード結果に問題があり、データベースを切り戻す場合、データベースのリストアボタンを実行

# Agenda

- アップグレード
- 追加された初期化パラメータ、バックグラウンド・プロセス
  - PGA\_AGGREGATE\_LIMIT パラメータ
  - DNFS\_BATCH\_SIZE パラメータ
  - PROCESSOR\_GROUP\_NAME パラメータ
  - THREADED\_EXECUTION パラメータ
  - バックグラウンド・プロセス(一部)
- ユーティリティ

# 追加された初期化パラメータ

## 新機能に関連して追加された初期化パラメータ

初期化パラメータ	概要
ENABLE_PLUGGABLE_DATABASE	コンテナ・データベースの作成を制御
NONCDB_COMPATIBLE	プラグブル・データベースからの操作を Non-CDB と同様の挙動にするかを制御
PDB_FILE_NAME_CONVERT	プラグブル・データベース作成時にファイル名のマッピングパターンを指定
HEAT_MAP	ヒート・マップおよび自動データ最適化機能を制御
MAX_STRING_SIZE	VARCHAR2、NVARCHAR2 および RAW データ型の最大サイズを制御
TEMP_UNDO_ENABLED	一時 UNDO 機能の使用を制御
OPTIMIZER_ADAPTIVE_FEATURES	適応的なオプティマイザの機能を制御
OPTIMIZER_ADAPTIVE_REPORTING_ONLY	適応計画、自動再最適化の制御

# 追加された初期化パラメータ

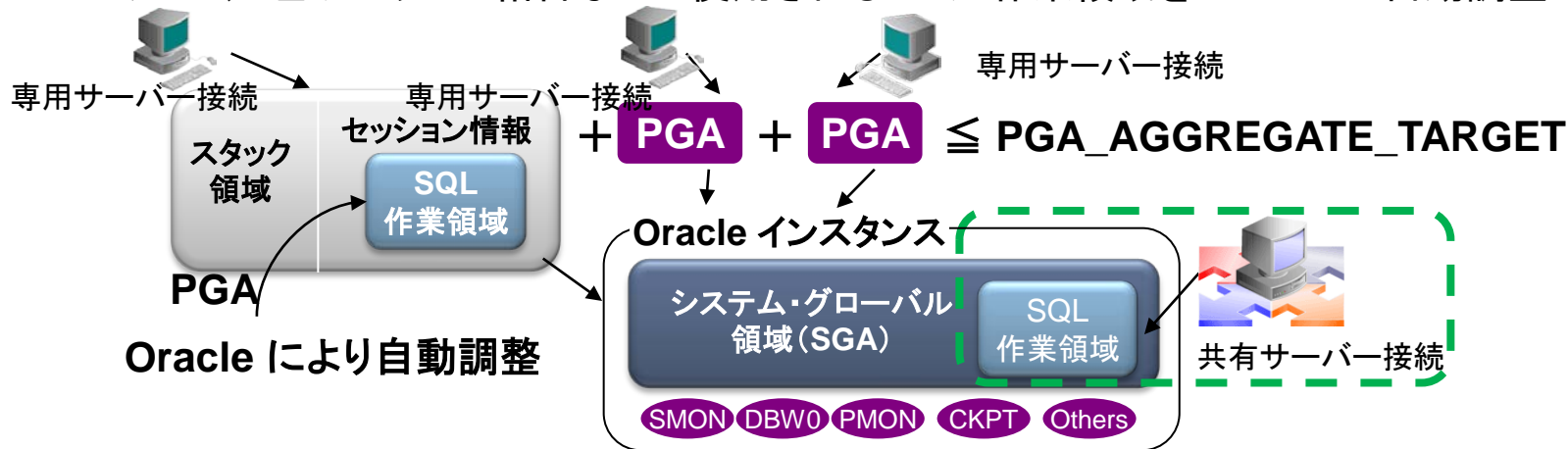
## 新機能に関連して追加された初期化パラメータ

初期化パラメータ	概要
PGA_AGGREGATE_LIMIT	インスタンス全体の PGA の上限値を指定
DNFS_BATCH_SIZE	Direct NFS 使用時に Oracle プロセスのキュー可能な非同期 I/O 数を指定
PROCESSOR_GROUP_NAME	インスタンスで利用するプロセッサグループ名を指定
THREADED_EXECUTION	マルチプロセス・マルチスレッド・モデルの利用を制御
CONNECTION_BROKERS	接続ブローカ・タイプやその数などを指定
USE_DEDICATED_BROKER	専用サーバーの起動方法を制御
UNIFIED_AUDIT_SGA_QUEUE_SIZE	統合監査用にとられる SGA キューのサイズを指定
SPATIAL_VECTOR_ACCELERATION	spatial vector acceleration の利用を制御

# PGA\_AGGREGATE\_LIMIT パラメータ

## 従来の PGA のメモリー管理

- Oracle 9i Database で自動 PGA メモリー (PGA 自動調整機能) が導入された
- インスタンスでアクティブな作業領域に使用できる PGA メモリーの総量を、PGA\_AGGREGATE\_TARGET 初期化パラメータから自動的に導出
  - ソート処理やハッシュ結合などに使用される SQL 作業領域を Oracle が自動調整



# PGA\_AGGREGATE\_LIMIT パラメータ

## 自動 PGA のメモリーの動作と制限

- PGA\_AGGREGATE\_TARGET 初期化パラメータで指定する値は、目標値であり、必ずしも指定した値以内のメモリーサイズに留めることはできない
  - 指定した値よりもサイズが大きくなるケース
    - 大量のプロセスが一斉にソート処理を始めたケース
    - インスタンスへ接続するプロセス数が増大していった場合
  - SGA の SGA\_TARGET 初期化パラメータのように、指定した値のサイズを超過しないような仕組みではない
- OS の仮想メモリーを空き領域が不足するまで PGA のサイズが拡大した場合や、プロセスが使用可能なメモリーの制限に達した場合、ORA-4030 エラーが発生する
  - OS 全体のメモリー を枯渇させて、システム全体を危険な状態にしてしまう可能性がある



# PGA\_AGGREGATE\_LIMIT パラメータ

## PGA の上限サイズ制限の導入

- インスタンス全体で獲得可能な PGA の上限値を設定
  - 上限を超えた場合、PGA を消費しているセッションで ORA-4036 が発生
  - SYS ユーザー、バックグラウンド・プロセスはエラーの対象外
  - CKPT プロセスが 3秒毎に上限を超えていないかチェック
- デフォルト値: 以下で算出された値のうちの最大値
  - 2048 MB、PGA\_AGGREGATE\_TARGET × 2、PROCESSES × 3MB
  - デフォルト値の場合でもアラートログに出力

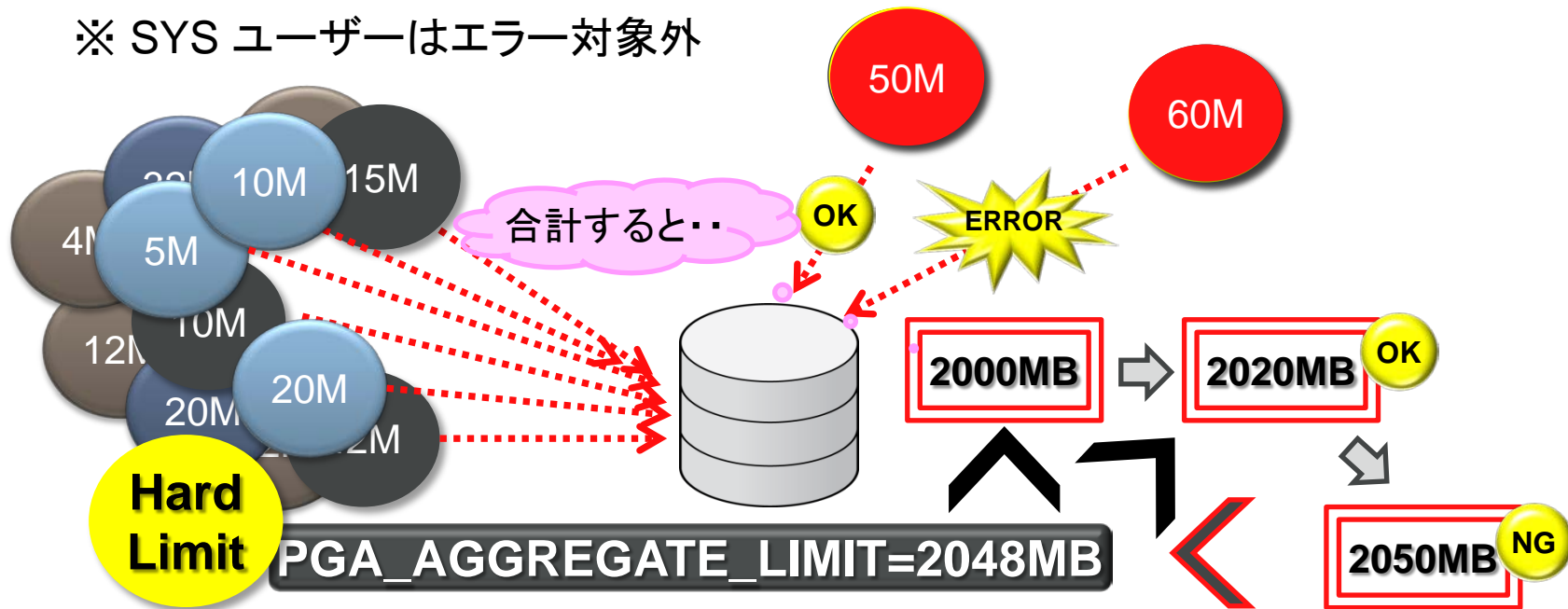
出力例

```
Using default pga_aggregate_limit of 2048 MB
```

# PGA\_AGGREGATE\_LIMIT パラメータ

## PGA\_AGGREGATE\_LIMIT の動作イメージ

- インスタンスの PGA の合計値が超えた場合に**エラー (ORA-4036)** を返す  
※ SYS ユーザーはエラー対象外



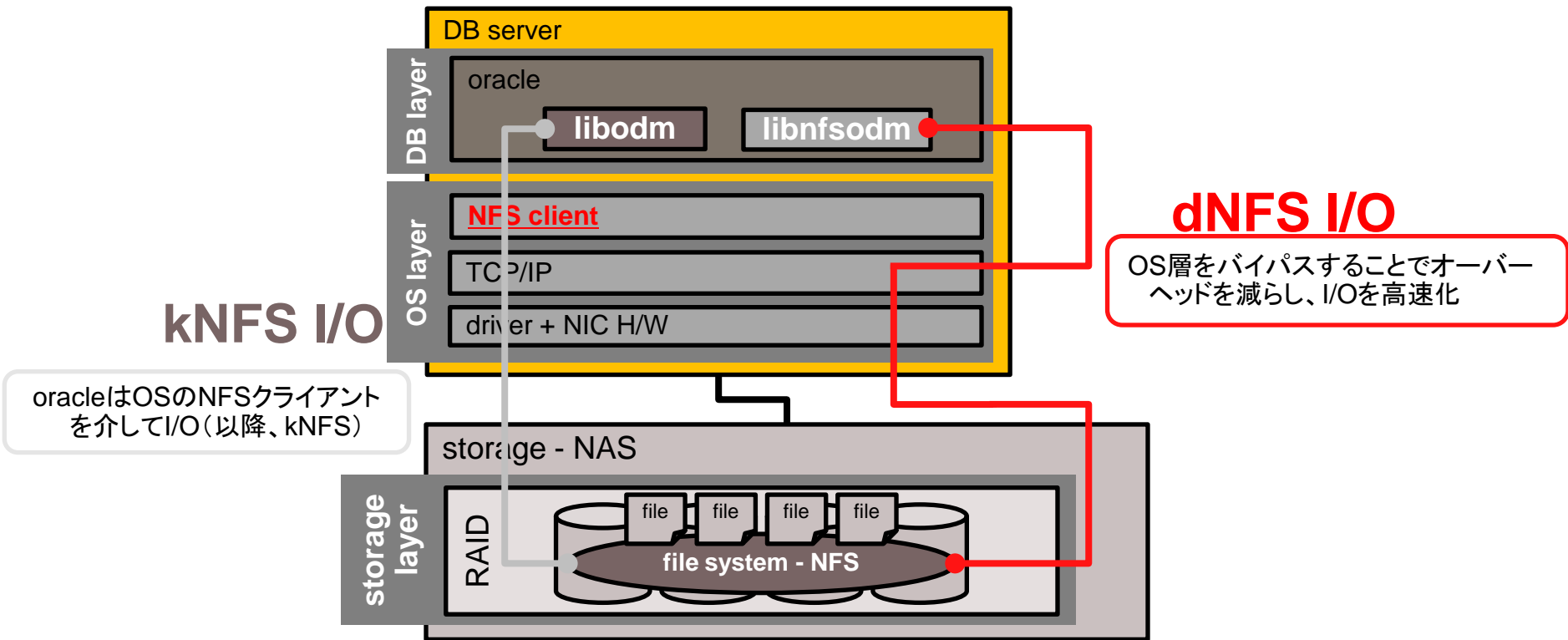
# DNFS\_BATCH\_SIZE パラメータ

## Direct NFS (dNFS) の特徴

- dNFSとは
  - Oracle Database内部に実装されたNFSクライアント機能
  - Oracle Database 11g Release 1から使用可能
  
- dNFSの特徴
  1. OSカーネルのNFSクライアント (kNFS) より高いディスクI/O性能
  2. 簡単な手順で機能を有効化
    - アプリケーションの書き換えは必要ない
    - ストレージの構成や運用に影響はない
    - 複数イーサネット・ポートを使用したネットワーク帯域のスケーラビリティの設定が簡単

# DNFS\_BATCH\_SIZE パラメータ

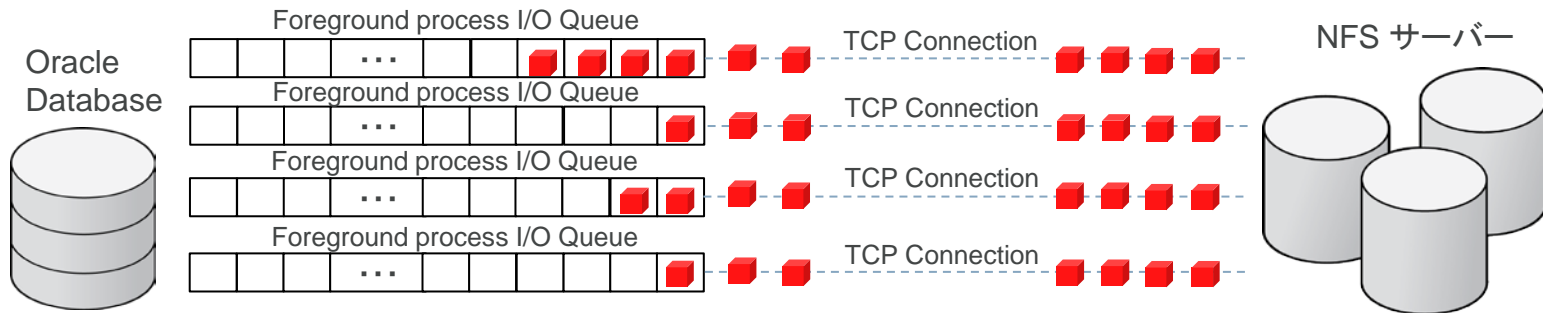
## Direct NFS (dNFS) の概要



# DNFS\_BATCH\_SIZE パラメータ

## dNFSのアーキテクチャ

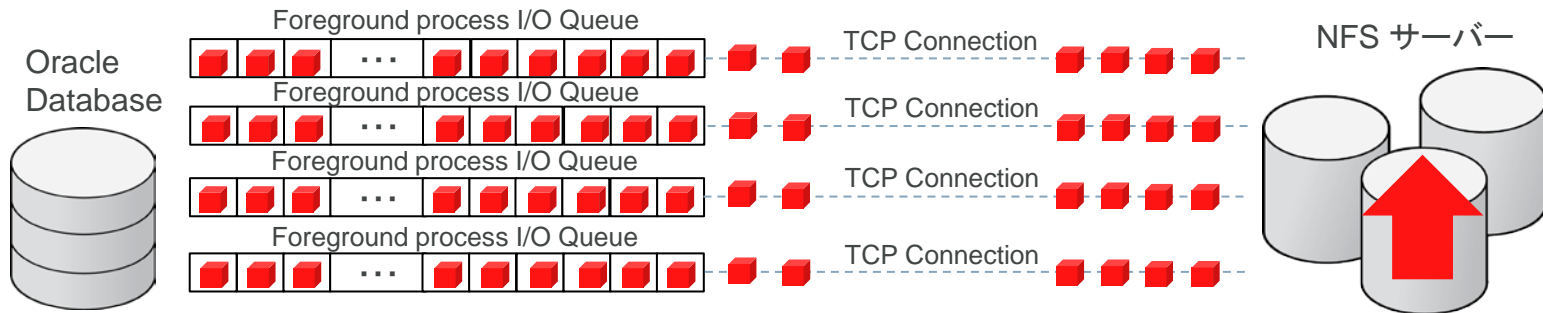
- dNFSでは、RPCデーモンをバイパスし、各dNFSクライアント(Oracle プロセス)がTCPコネクションをNFSサーバーに対して確立する
  - dNFSでは、1000までの並列処理が可能
  - RPCデーモンをバイパスすることで、直接NFSサーバーとのやりとりが可能



# DNFS\_BATCH\_SIZE パラメータ

## dNFS での課題

- foregroundプロセスが直接TCPコネクションを確立し、短期間に大量の非同期I/Oをすることで、NFSサーバーに対しての負荷が、kNFSのアーキテクチャと比較すると増えてしまう
  - コネクション数が増え、NFSサーバーに対するI/Oが増えれば、NFSサーバーの負荷も比例して増える



# DNFS\_BATCH\_SIZE パラメータ

## dNFSクライアントの非同期I/O制御

- DNFS\_BATCH\_SIZE パラメータの導入
  - dNFSクライアント(Oracle Database)の発行するI/Oリクエストの数を制限し、NFSサーバーのパフォーマンスに合わせた設定が可能
  - 設定可能な値 : 0 ~ 4096 (デフォルト: 4096)
  - 動的変更は不可
  - 設定指針
    - 非同期 I/O リクエストを大量に処理できずパフォーマンスが低下する環境では、128 に設定した上で、NFSサーバーのパフォーマンスにより調整

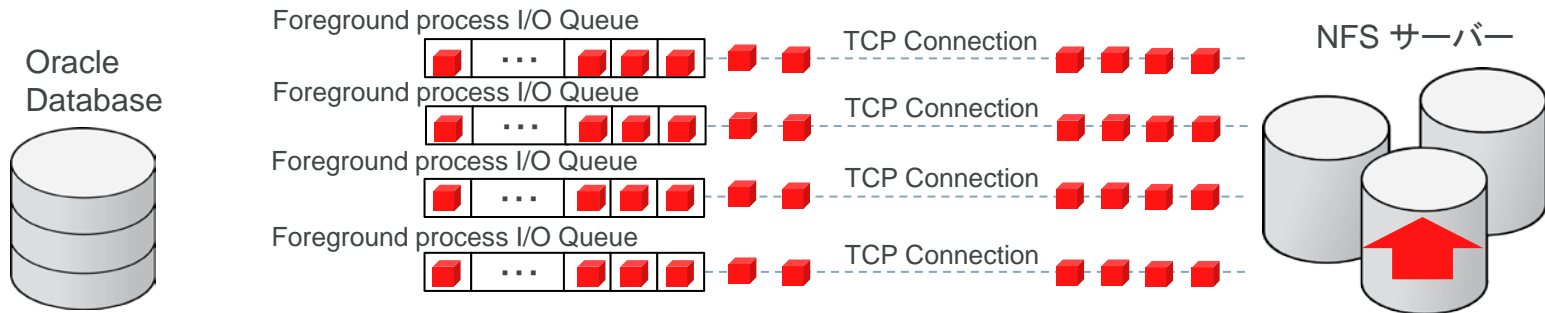
参考: Oracle Database Performance Tuning Guide 12c Release 1 (12.1)

18 Managing Operating System Resources

# DNFS\_BATCH\_SIZE パラメータ

## dNFS側でのI/Oリクエスト数の制限

- foreground プロセスが持つ I/O キューの数を制限し、foreground プロセスからの非同期 I/O の同時大量リクエストの発生を低減し、NFS サーバーのオーバーロードを防ぐ

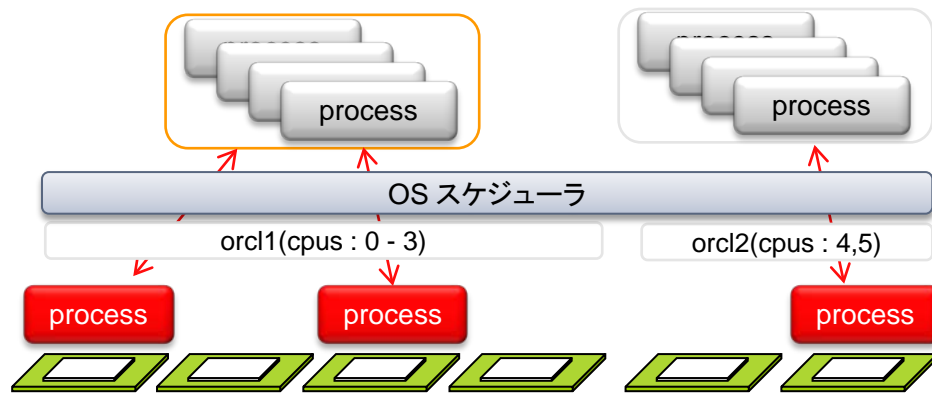




# PROCESSOR\_GROUP\_NAME パラメータ

## 概要

- インスタンスが稼働するプロセッサ・グループ名を指定することが可能
  - インスタンスのプロセスが指定されたプロセッサ・グループ内の CPU 上で稼働
  - 対応プラットフォームおよびバージョン
    - Linux: cgroup (2.6.32カーネル以上) に対応
    - Solaris: リソース・プール (Solaris 11 SRU4) に対応



OSレベルで使用可能なCPUを論理的に分割する

# PROCESSOR\_GROUP\_NAME パラメータ

## 設定方法と確認方法

### ■ 設定方法

1. \$ORACLE\_HOME/rdbms/install/setup\_processor\_group.sh を使用してプロセッサ・グループを作成
2. 初期化パラメータ PROCESSOR\_GROUP\_NAME に 1. で指定したグループ名を指定し、インスタンスを再起動

### ■ 確認方法

- alter.log に以下のようなメッセージが出力される

```
Sat Aug 10 15:27:13 2013
```

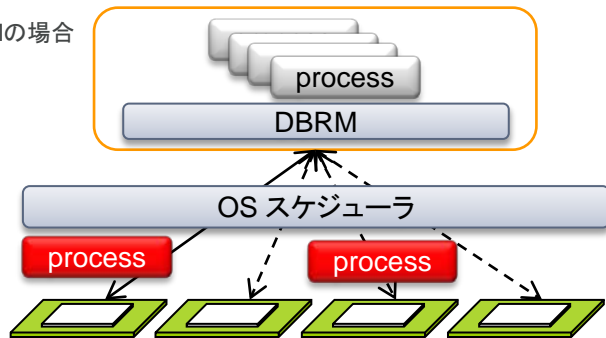
```
Instance has been started in processor group ORCL (NUMA Nodes: 0 CPUs: 0)
```

# PROCESSOR\_GROUP\_NAME パラメータ

## データベース・リソース・マネージャとの違い

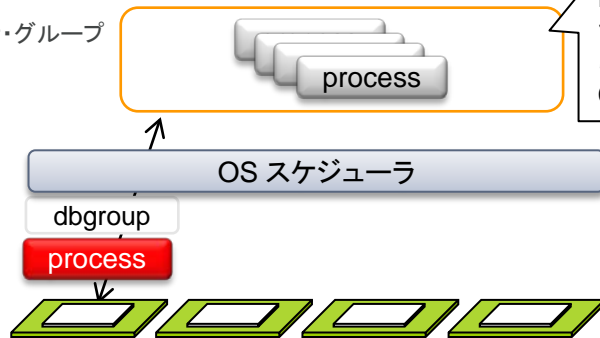
- データベース・リソース・マネージャ(DBRM)は、Oracle Database内でのCPU使用率を調整
  - どのコアで稼働するかはOS側の制御にゆだねられる
- プロセッサ・グループはOS(カーネル)でのリソースを制御
  - カーネルでグループに対する CPU 割り当てを制御

DBRMの場合



DBRMでは、Oracle Database内で、(CPU\_COUNT/OSのCPU数)で計算される値までしかCPUを使用しない  
実際にどのCPUが使用されるのまでは定義できない

プロセッサ・グループの場合



cgroupやresource pool では、OS(カーネル)のレイヤーで、CPUリソースをグループ毎に分割する  
実際にどのCPUを使用するかというところまで定義可能

PROCESSOR\_GROUP\_NAME  
で指定されたグループに割り当て  
られているCPU数に  
CPU\_COUNTの値が変更される

# THREADED\_EXECUTION パラメータ

## プロセス・アーキテクチャ

- マルチプロセス・モデル (Linux / Unix)
  - Oracle プロセスと OS プロセスが 1対1 に対応
    - Oracle プロセス数が増えると OS プロセス数増
  - Oracle プロセスの独立性大
- マルチスレッド・モデル (Windows)
  - Oracle プロセスが 1つの OS プロセスに対応
    - Oracle プロセス数が増えると OS スレッド数増
  - Oracle プロセスの独立性低



# THREADED\_EXECUTION パラメータ

## マルチプロセス・マルチスレッド・モデル

- 複数の OS プロセスに対して、複数の Oracle プロセスがスレッドとして対応するモデル
- 特徴
  - OS プロセス数の増減、Oracle プロセスの OS プロセスへの割当ては、Oracle が管理
  - OS プロセス数が減ることで、リソースを抑制
  - 必須 Oracle プロセス (LGWR、SMON など) は、1つの OS プロセスに集約
  - アプリケーション側で大きな設定変更を必要としない



ORACLE

# THREADED\_EXECUTION パラメータ

## 概要

- THREADED\_EXECUTION パラメータ
  - 設定可能な値
    - FALSE : 従来型モデル(デフォルト)
    - TRUE : 新しいプロセスモデルを利用
  - Linux、UNIX 環境で利用可能
  - TRUE にした場合
    - ora\_uNNN\_<SID> のスレッドとして起動する
    - v\$process の SPID、STID より、OS 側のプロセスID、スレッドID を確認可能



# THREADED\_EXECUTION パラメータ

## 制限事項

- 新しいモデルを利用時には、OS 認証 (sqlplus / as sysdba) によるログインは不可 (ORA-1031 が発生)
  - パスワード認証でのログインが必要
- リスナー経由のサーバープロセスを OS プロセスのスレッドとする場合、事前に listener.ora に下記のパラメータの設定が必要
  - デフォルトでは、スレッド化されない

```
dedicated_through_broker_<リスナー名>=ON
```

- PMON、DBWR、PSP0、VKTM はスレッド化の対象外

# THREADED\_EXECUTION パラメータ

## パラメータ設定時の見え方

```
SQL> select count(*) from v$process;
```

```
  COUNT(*)  
-----  
         52
```

Oracle プロセスは 52 個に対して、  
対応するOSプロセスは 6個

```
SQL> ! ps -elf |grep orcl |grep -v grep
```

```
0 S oracle  15297    1 0 80  0 - 340832 semtim 12:26 ?      00:00:00 ora_pmon_orcl  
0 S oracle  15299    1 0 80  0 - 340832 semtim 12:26 ?      00:00:00 ora_psp0_orcl  
0 S oracle  15301    1 2 58  - - 340832 hrtim  12:26 ?      00:00:01 ora_vktm_orcl  
0 S oracle  15305    1 2 80  0 - 624963 semtim 12:26 ?      00:00:01 ora_u004_orcl  
0 S oracle  15311    1 32 80  0 - 1189388 semtim 12:26 ?     00:00:18 ora_u005_orcl  
0 S oracle  15317    1 0 80  0 - 342543 semtim 12:26 ?      00:00:00 ora_dbw0_orcl
```

スレッド化されたプロセスのトレースファイル名は、<SID>\_<PNAME>\_<OSPID>\_<OSTID>.trc

```
alert_orcl.log          orcl_dbw0_15874.trc      orcl_ora_15868_15887.trc  
orcl_aqpc_15868_15897.trc orcl_lgwr_15862_15875.trc orcl_ora_15868_15892.trc  
orcl_ckpt_15862_15876.trc orcl_mmon_15868_15882.trc orcl_tt00_15868_15894.trc  
orcl_dbrm_15862_15871.trc orcl_ora_15843.trc      orcl_vktm_15858.trc
```



# 追加されたバックグラウンド・プロセス(一部)

- TTnn(nn:数字) : REDO 転送スレーブ・プロセス
  - Data Guard 環境で REDO を非同期転送する
- TMON:トランスポート・モニター
  - TTnn の起動および監視の実施
- SCCn、SCRB、SCRn、SCVn
  - ASMインスタンスで、スクラビング機能動作時に起動するプロセス
  - スクラビングの各フェーズ毎にプロセスあり

# 追加されたバックグラウンド・プロセス(一部)

- LGnn(nn:数字) : ログ・ライターのワーカープロセス
  - LGWR の REDO ログ書き込みを補助
- BWnn(nn:数字) : データベース・ライター・プロセス
  - DBWn(データベース・ライター) が 37個以上起動する場合に利用
  - ※ DB\_WRITER\_PROCESSES で指定可能な値が100まで増加
- LREG : リスナー登録プロセス
  - データベース・インスタンスやディスパッチャ・プロセスの情報をリスナーに登録する(これまでは、PMONにて実施)



# ***DEMONSTRATION***

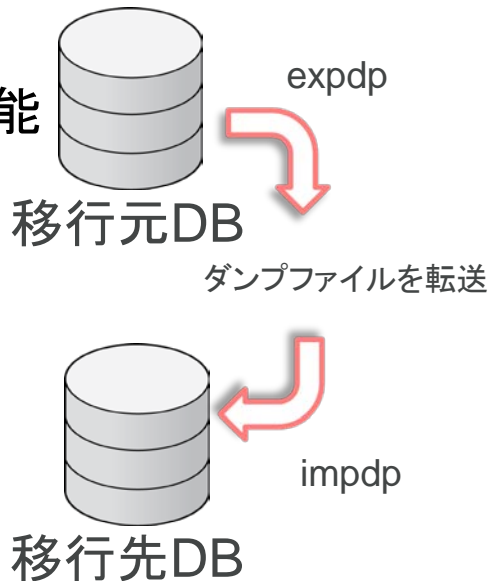
# Agenda

- アップグレード
- 追加された初期化パラメータ、バックグラウンド・プロセス
- ユーティリティ
  - Oracle Data Pump フル・トランスポートابل
  - Oracle Data Pump 新しいパラメータ・オプション
  - Oracle SQL\*Loader Express
  - Oracle SQL\*Loader 及び外部表での dNFS の使用

# Oracle Data Pump

## 概要

- データおよびメタデータを高速にアンロードまたはロードする機能
  - 主にデータベース間のデータ移動に利用
- ユーティリティまたは PL/SQL パッケージから利用可能
  - expdp および impdp
  - DBMS\_DATAPUMP、DBMS\_METADATA
- 豊富な付加機能
  - エクスポート・インポートの処理の平行化
  - ダンプファイルの圧縮や暗号化
  - 外部表の利用



# Oracle Data Pump の新機能 (フル・トランスポートابل)

## 概要

- トランスポートابل表領域 **~ 12c**
  - 表領域のメタデータ(管理情報)のみを expdp/impdp
  - データはメタデータ取得時のデータファイルとして転送
  - プラットフォームをまたぐ転送が可能
- フル・トランスポートابل **12c New**
  - 従来までのトランスポートابل表領域の特徴を継承しつつ、表領域レベルからデータベースレベルに拡張
  - 管理者表領域(SYSTEM 表領域など)に格納されたデータは、ダンプファイルを利用して移行

# Oracle Data Pump の新機能 (フル・トランスポートابل)

## フル・トランスポートابل・エクスポート/インポートの利用用途

- マルチテナント・アーキテクチャでの利用
  - NON-CDB を別の NON-CDB へ移行
  - NON-CDB を PDB として CDB へ移行
  - PDB を別の PDB へ移行
  - PDB を NON-CDB へ移行
- 異なるプラットフォームへのデータベースの移行
- 新しいリリースへのアップグレード
  - Oracle Database 11g (11.2.0.3 - )から Oracle Database 12c への移行
  - ※ Oracle Database 11g 環境では11.2.0.3 以降でエクスポートに対応
  - ※ 11.2 環境では expdp 実行時にversion=12 の指定が必要

# Oracle Data Pump の新機能 (フル・トランスポートابل)

## フル・トランスポートابل・エクスポート

- フル・トランスポートابل・エクスポートを実行するためのパラメータ
  - TRANSPORTABLE = ALWAYS
  - FULL = Y
  - VERSION = 12 (11.2 環境の場合)
- コマンド例 (11.2.0.3 環境の場合)

```
$ expdp system full=y transportable=always version=12 ¥  
dumpfile=exp.dmp logfile=exp.log
```

※コマンド発行前にユーザー管理表領域をすべて読取り専用モードに変更

※暗号化された表領域もしくは列を含む場合、ENCRYPTION\_PASSWORD  
または ENCRYPTION\_PWD\_PROMPT パラメータを設定



# Oracle Data Pump の新機能 (フル・トランスポータブル)

## フル・トランスポータブル・エクスポート

- 実行例 (11.2.0.3 環境では version=12 を指定)

```
$ expdp system full=y transportable=always version=12 dumpfile=exp.dmp
..(略)..
マスター表"SYSTEM"."SYS_EXPORT_FULL_01"は正常にロード/アンロードされました
*****
SYSTEM.SYS_EXPORT_FULL_01に設定されたダンプ・ファイルは次のとおりです：
  /u01/app/oracle/admin/fromdb/dpdump/exp.dmp ←
*****
トランスポータブル表領域EXAMPLEにはデータファイルが必要です：
  /u01/app/oracle/oradata/fromdb/example01.dbf ←
トランスポータブル表領域USERSにはデータファイルが必要です：
  /u01/app/oracle/oradata/fromdb/users01.dbf ←
ジョブ"SYSTEM"."SYS_EXPORT_FULL_01"が 13:39:20で正常に完了しました
```

- impdp 実行に必要なファイルを確認

# Oracle Data Pump の新機能 (フル・トランスポートブル)

## ダンプ・ファイルとデータファイルの転送

- エクスポート・ダンプ・ファイルの転送
  - 移行先データベースからアクセス可能なディレクトリに配置
  - ディレクトリ・オブジェクトがなければ、ディレクトリ・オブジェクトを作成
- ユーザ定義表領域のすべてのデータファイルを転送
  - 移行先データベースからアクセス可能なディレクトリに配置
  - 移行元と移行先の間で、プラットフォームのエンディアンが異なる場合、DBMS\_FILE\_TRANSFER パッケージもしくは、RMAN の CONVERT コマンドにより移行先プラットフォームのエンディアンへの変換が必要
  - データファイル転送後、必要に応じて、移行元の表領域を  
読取り/書込みモードに変更

# Oracle Data Pump の新機能 (フル・トランスポートابل)

## フル・トランスポートابل・インポート

- フル・トランスポートابل・インポートを実行するためのパラメータ
  - FULL=Y
  - TRANSPORT\_DATAFILES = '<データファイルのパス>', ...
- ※ データファイルのパスには、expdp 実行時に確認したファイルのパス
- コマンド例

```
$ impdp system full=y transport_datafiles= ¥  
'/u01/app/oracle/oradata/todb/example01.dbf', ¥  
'/u01/app/oracle/oradata/todb/users01.dbf' ¥  
dumpfile=exp.dmp logfile=imp.log
```

※暗号化された表領域もしくは列を含む場合、ENCRYPTION\_PASSWORD  
または ENCRYPTION\_PWD\_PROMPT パラメータを設定

# Oracle Data Pump の新機能 (フル・トランスポートابل)

## その他の制限事項や考慮点

- トランスポートに関する一般的な制限事項はフル・トランスポートابلも同様
- PDB で impdp/expdp を実行する場合の考慮点
  - PDB に接続して実行が必要 `$ expdp pdbadmin @pdb1 ....`
  - PDB 上でディレクトリ・オブジェクトの作成が必要 (デフォルトの利用不可)
- 暗号化された表領域は、エンディアンが異なる環境へトランスポート不可
- 管理表領域 (SYSTEM表領域等) とユーザ一定義表領域の両方で定義されているようなオブジェクトはトランスポート不可

参考: Oracle Database Administrator's Guide 12c Release 1 (12.1)  
15 Transporting Data  
Limitations on Full Transportable Export/import

# Oracle Data Pump の新機能

## その他の新規パラメータ、オプション

- VIEWS\_AS\_TABLES パラメータ (expdp)
- TRANSFORM パラメータ (impdp)
  - DISABLE\_ARCHIVE\_LOGGING オプション
  - TABLE\_COMPRESSION\_CLAUSE オプション
  - LOB\_STORAGE オプション
- ENCRYPTION\_PWD\_PROMPT パラメータ (expdp / impdp)
- LOGTIME パラメータ (expdp / impdp)
- COMPRESSION\_ALGORITHM パラメータ (expdp)

# Oracle Data Pump の新機能

## VIEWS\_AS\_TABLES パラメータ **12c New**

- expdp 時に指定したビューを表としてエクスポート可能
- 設定可能な値
  - ビューの存在するスキーマ名 (デフォルト : expdp 実行ユーザー)
  - 表としてエクスポートするビュー名 (必須)
  - エクスポートするビューのメタデータのソースとなる表名  
(デフォルト: ビューと同じ列およびデータ型を持つテンプレート表を作成)

```
$ expdp scott/tiger ... views_as_tables=scott.emp_v.emp
```

- 対象となるビュー: LOB以外のスカラー列のみを含むリレーショナル・ビュー
- KU\$VATから始まるテンプレート表が一時的に作成される

# Oracle Data Pump の新機能

## DISABLE\_ARCHIVE\_LOGGING オプション **12c New**

- impdp 時の表または索引のロギング属性を変更可能
- 設定可能な値
  - Y: インポート中のロギング属性を無効化
  - N: 無効化しない(デフォルト)

```
$ impdp scott/tiger ... transform=disable_archive_logging:Y
```

```
$ impdp scott/tiger ... transform=disable_archive_logging:Y:TABLE
```

※ Y にした場合でも、import 完了時にオブジェクトのロギング属性は戻る

※ FORCE LOGGINGモードの場合、設定オプションは無効

※ トランスポータブルのインポート時には指定できない

# Oracle Data Pump の新機能

## TABLE\_COMPRESSION\_CLAUSE オプション **12c New**

- impdp 時に表圧縮形式を変更可能
- 設定可能な値
  - 表圧縮句に準拠 (NOCOMPRESS など)
    - 従来の指定方法 (compress for OLTP など) にも対応
  - NONE : 表圧縮句なし (表領域のデフォルトの圧縮形式)

```
$impdp scott/tiger ...  
transform=table_compression_clause:¥"compress for OLTP¥"
```

※ 複数語の圧縮句を指定する場合、エスケープ文字とともに一重引用符  
または二重引用符が必要

※ オプション未指定の場合、移行元の表圧縮形式



# Oracle Data Pump の新機能

## LOB\_STORAGE オプション **12c New**

- impdp 時に LOBセグメントの記憶域を変更可能
- 設定可能な値
  - SECUREFILE: SECUREFILE LOB として作成
  - BASICFILE: BASICFILE LOB として作成
  - DEFAULT: デフォルトの記憶域 (DB\_SECUREFILE パラメータに依存)
  - NO\_CHANGE (デフォルト): export 実行時と同じ記憶域

```
$ impdp scott/tiger ... transform=lob_storage:securefile
```

※ トランスポートブルのインポート時には指定できない

# Oracle Data Pump の新機能

## ENCRYPTION\_PWD\_PROMPT パラメータ

- ENCRYPTION\_PASSWORD パラメータ **~ 12c**
  - expdp / impdp 時に暗号化のためのパスワードを引数として指定

```
$ expdp scott/tiger ... encryption_password = <passwd>
```

- ENCRYPTION\_PWD\_PROMPT パラメータ **12c New**
  - expdp / impdp 時に暗号化パスワードを入力要求を Yes、No で制御

```
$ expdp scott/tiger ... encryption_pwd_prompt = yes
```

- Data Pump 実行時にパスワードを求められる

暗号化パスワードが画面に表示されず、セキュリティ面に配慮

# Oracle Data Pump の新機能

## ENCRYPTION\_PWD\_PROMPT パラメータ

### ■ 実行例

```
$ expdp scott/tiger tables=emp encryption_password=password
```

画面への表示あり

```
$ expdp scott/tiger tables=emp encryption_pwd_prompt=yes
```

```
Export: Release 12.1.0.1.0 - Production on 水 7月 21 02:53:23 2013
```

```
Copyright (c) 1982, 2013, Oracle and/or its affiliates. All rights reserved.
```

```
接続先: Oracle Database 12c Enterprise Edition Release 12.1.0.1.0 - 64bit Production  
With the Partitioning, OLAP, Advanced Analytics and Real Application Testing options
```

暗号化パスワード:

画面への表示なし

```
"SCOTT"."SYS_EXPORT_TABLE_01"を起動しています: scott/***** tables=emp  
encryption_pwd_prompt=yes  
BLOCKSメソッドを使用して見積り中です...  
...
```

# Oracle Data Pump の新機能

## LOGTIME パラメータ **12c New**

- export / import 実行時のメッセージにタイムスタンプを付与
  - NONE : タイムスタンプを付けない(デフォルト)
  - STATUS : ステータス・メッセージにのみタイムスタンプ付与
  - LOGFILE : ログ・ファイルにのみタイムスタンプ付与
  - ALL : ステータス・メッセージおよびログ・ファイルにタイムスタンプ付与
- 出力例

```
$ expdp scott/tiger ... logtime=status  
...  
13-07-12 14:50:12.858: BLOCKSメソッドを使用して見積り中です...  
13-07-12 14:50:14.129: オブジェクト型TABLE_EXPORT/TABLE/TABLE_DATAの処理中です  
...
```

# Oracle Data Pump の新機能

COMPRESSION\_ALGORITHM パラメータ **12c New**

- expdp で生成されるダンプ・ファイルの圧縮アルゴリズムを指定
- 設定可能な値
  - BASIC : 圧縮率と速度の適度な組合せ (デフォルト)
  - LOW : 圧縮率低、リソース消費低
  - MEDIUM : BASIC と同様に適度な組合せ (BASIC とアルゴリズムは異なる)
  - HIGH : 圧縮率高、リソース消費大

```
$ expdp scott/tiger ... compression_algorithm=low
```

※ 圧縮レベルを選択する上では、実際にテストを実施することを推奨

# SQL\*Loader Express

## 概要

### SQL\*Loader

~ 12c

#### □ 様々なデータを高速にロード

- 高速なデータのロードを実現
- 様々な形式の表・データに対応
- パラメータによる多種多様なカスタマイズ

#### Negative

- 制御ファイル作成
- 手順が複雑



### SQL\*Loader Express

12c New

#### □ シンプルなデータを簡単にロード

- 制御ファイルの作成が不要
- パラメータを指定したカスタマイズも可能
- SQL\*Loader と同等の高速なロード

# SQL\*Loader Express

## 使用方法

- コマンド

- ユーザ名・表名の指定のみでのデータのロード

```
$ sqlldr scott/tiger table=EMP
```

- 実行時に必要なファイル

- ロードするデータのみ

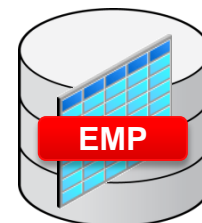
- 各列のデータは「,」で区切り、各行は「改行」で変更
- ファイル名は「<表名>.dat」で作成

```
7369,SMITH,CLERK,7902,1980-12-17 00:00:00,800,,20  
7499,ALLEN,SALESMAN,7698,1981-02-20 00:00:00,1600,300,30  
7521,WARD,SALESMAN,7698,1981-02-22 00:00:00,1250,500,30  
.....
```

制御ファイルは不要

- 列データは「,」で区切る
- 行データは「改行」で区切る

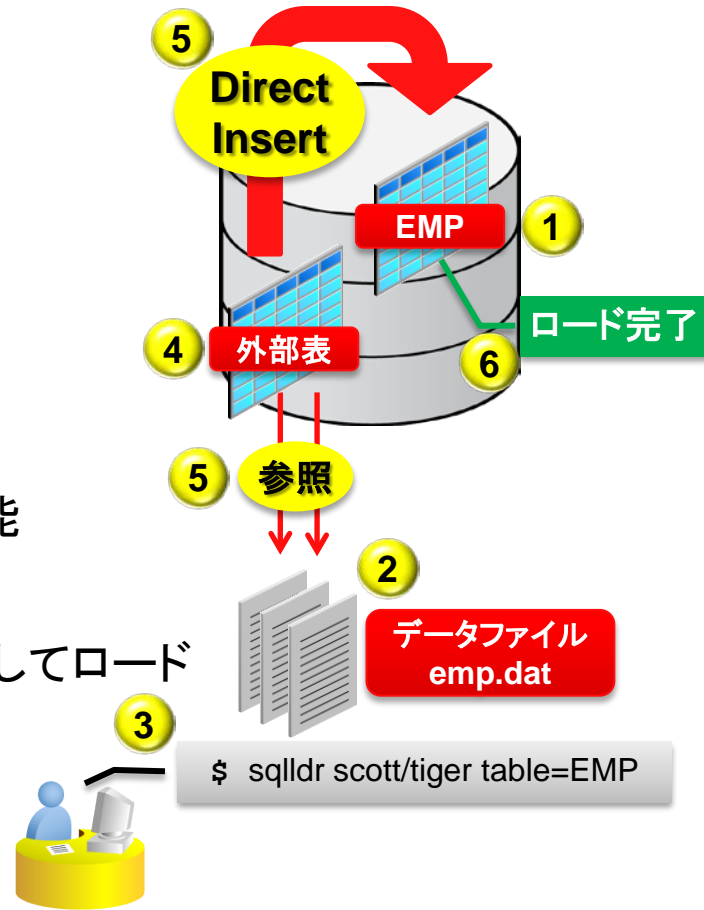
Load



# SQL\*Loader Express

## デフォルト動作

- 外部表を経由したロード
  - ロード中に外部表の作成・削除を自動実行
  - 外部表を使用して、ロードするデータを参照
  - ダイレクト・インサートによりデータをロード
  - ロードを実行せず、SQL文だけの生成も可能
  - EXTERNAL\_TABLE パラメータで制御
    - EXECUTE (デフォルト) : 外部表を使用してロード
    - NOT\_USERD : 従来型モードでロード
    - GENERATE\_ONLY : SQL文のみ生成





# SQL\*Loader Express

## デフォルト動作

- データファイルの指定
  - カレントディレクトリの「<表名>.dat」を読み込む
  - DATA パラメータで任意のファイルの読み込みも可能

- 単一ファイルの読み込み

```
$ sqlldr scott/tiger table=EMP data=emp.dat
```

- 複数ファイルの読み込み

- 複数ファイルのデータを一括ロード

```
$ sqlldr scott/tiger table=EMP data='emp*.dat'
```

```
$ sqlldr scott/tiger table=EMP data='emp?.dat'
```

```
$ sqlldr scott/tiger table=EMP data='emp1.dat','emp2.dat'
```

```
$ sqlldr scott/tiger table=EMP data=emp1.dat data=emp2.dat
```

任意の値・文字列を含むファイル  
(ex) emp.dat, emp1.dat, emp10.dat

任意の値(1文字)を含むファイル  
(ex) emp1.dat, emp2.dat, emp3.dat

# SQL\*Loader Express

## 出力ファイル

- 実行時に以下のファイルが出力される
  - <表名>.log
    - SQL\*Loader 実行時のログファイル
    - SQL\*Loader から実行されている SQL 文も参照可能
  - <表名>\_<実行プロセスのPID>.log\_xt
    - Database 上での処理実行時のログファイル
    - エラー発生時にはそのエラー情報を出力
  - <表名>\_<PID>.bad
    - ロードできなかったデータ
    - ロード時にエラーが発生した場合のみ出力される
    - DATA パラメータ指定時は「<最初のDATAファイル名>\_<PID>.bad」となる



```
$ sqlldr scott/tiger table=EMP
```

- emp.log
- emp\_21300.log\_xt
- emp\_21300.bad

```
$ sqlldr scott/tiger table=EMP  
DATA=emp1.dat, emp2.dat
```

- emp.log
- emp\_21301.log\_xt
- emp\_21302.log\_xt
- emp\_21301.bad
- emp\_21302.bad

# SQL\*Loader Express

## パラメータ

- DEGREE\_OF\_PARALLELISM
  - パラレル度を設定、デフォルト AUTO でパラレル化
- DIRECT
  - ダイレクト・ロードの使用可否を設定、デフォルト外部表ロードのため FALSE
- BAD
  - bad ファイルの出力ディレクトリ・名前を指定
- SILENT
  - 画面上への出力メッセージを制御
- DATA
  - インプットファイルの指定

# SQL\*Loader及び外部表でのdNFSの使用

- Oracle SQL\*Loader 及び外部表の入力ファイルに対して、dNFS を使用したアクセスが可能
  - 1GB以上の入力ファイルに対しては、自動で dNFS を使用したアクセスを実施
  - 1GB以下の入力ファイルに対しては、kNFS を使用したアクセスを実施
- また、DNFSによる読み取りバッファ数に対して制限をかけることも可能
  - DNFS\_READBUFFERSパラメータによる指定

# SQL\*Loader及び外部表でのdNFSの使用

## SQL\*Loaderでの使用方法

1. dNFSを使用して入力ファイルにアクセスする場合

```
$ sqlldr username/password control=control_file DNFS_ENABLE=TRUE
```

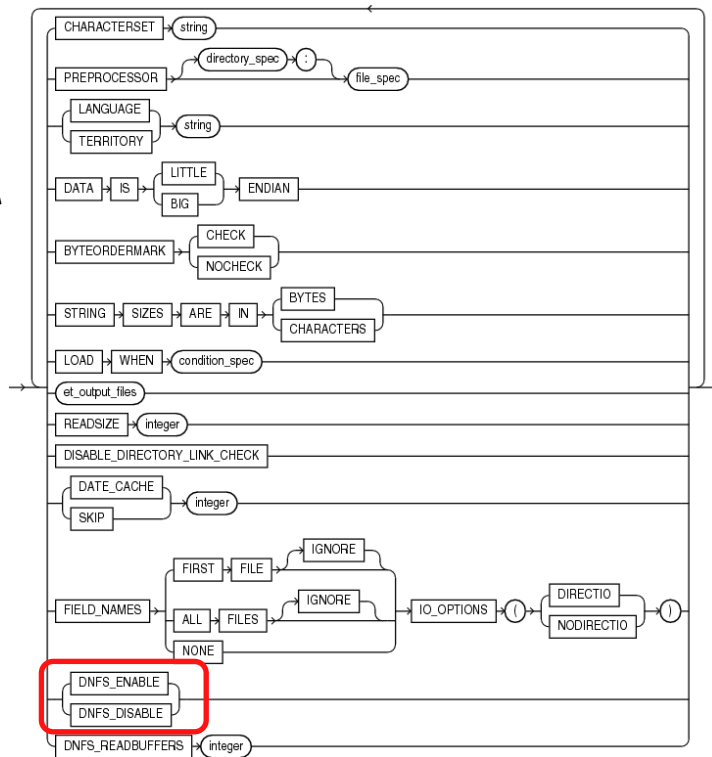
2. dNFSを使用せずに入力ファイルにアクセスする場合

```
$ sqlldr username/password control=control_file DNFS_ENABLE=FALSE
```

# SQL\*Loader及び外部表でのdNFSの使用

## 外部表での使用方法

- 外部表でDNFSが利用されるのは、ORACLE\_LOADERを使用した場合
  - ORACLE\_DATAPUMPでは使用できない
- access パラメータのDNFS\_ENABLE/DNFS\_DISABLEで指定する





# ***DEMONSTRATION***





**Hardware and Software**

**ORACLE®**

**Engineered to Work Together**

ORACLE®