

ORACLE®

ORACLE®

達人に聞く！データベースアップグレード成功の極意

Oracle Database 12c への アップグレード

日本オラクル株式会社
テクノロジー製品事業統括本部
製造ソリューション部
大橋 洸輔



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

- はじめに
- マルチテナント・アーキテクチャ概要
- Oracle Database 12c へのアップグレード事例
- Appendix: 12c 新機能

Agenda

- はじめに
- マルチテナント・アーキテクチャ概要
- Oracle Database 12c へのアップグレード事例
- Appendix: 12c 新機能

アップグレード時に確認すること

質問	回答
新規ハードウェアへの移行を考えていますか？	
オペレーティング・システムを変更する予定はありますか？	
キャラクタ・セットを変更しますか？	
データベース統合（スキーマ / インスタンス）を計画していますか？	
いくつのデータベースをアップグレードしますか？	
アップグレード・移行するデータベースの容量はどれくらいですか？	
移行元と移行先の Oracle の正確なバージョンは？	
許容できるダウンタイムはどの程度ですか？	
切り戻しの要件は何ですか？	
テスト環境とテストツールは？	
パフォーマンスの基準値はありますか？	
新機能を有効化しますか？	

一般的な準備手順（チェックリスト）

移行元の環境で実施

クリーンナップ

- リサイクル・ビンをパージ
- SYS / SYSTEM のINVALID オブジェクトをチェック
- SYS / SYSTEM の重複オブジェクトをチェック
- 古いパラメータを削除

コンポーネント

- INVALID コンポーネントをチェック
- 必要なコンポーネントをチェック
- 廃止されたコンポーネントを削除

パフォーマンス

- パフォーマンス統計を保存
- ネットワーク・パフォーマンスをチェック

オプション

- 整合性チェックを実施

一般的な準備手順（チェックリスト）

移行先の環境で実施

チェック

- ソフトウェア・サーティフィケーション情報をチェック
- インストール要件をチェック

ダウンロード

- ソフトウェアとパッチ・セット (PSR)
- CPU、PSU、バンドルパッチ (BP)
- 個別パッチ

設定

- 推奨パラメータ

参考 : Upgrade Companion

Oracle Database のアップグレードにおけるベスト・プラクティス

- [MOS Note:1462240.1](#) Oracle Database 12cR1 Upgrade Companion



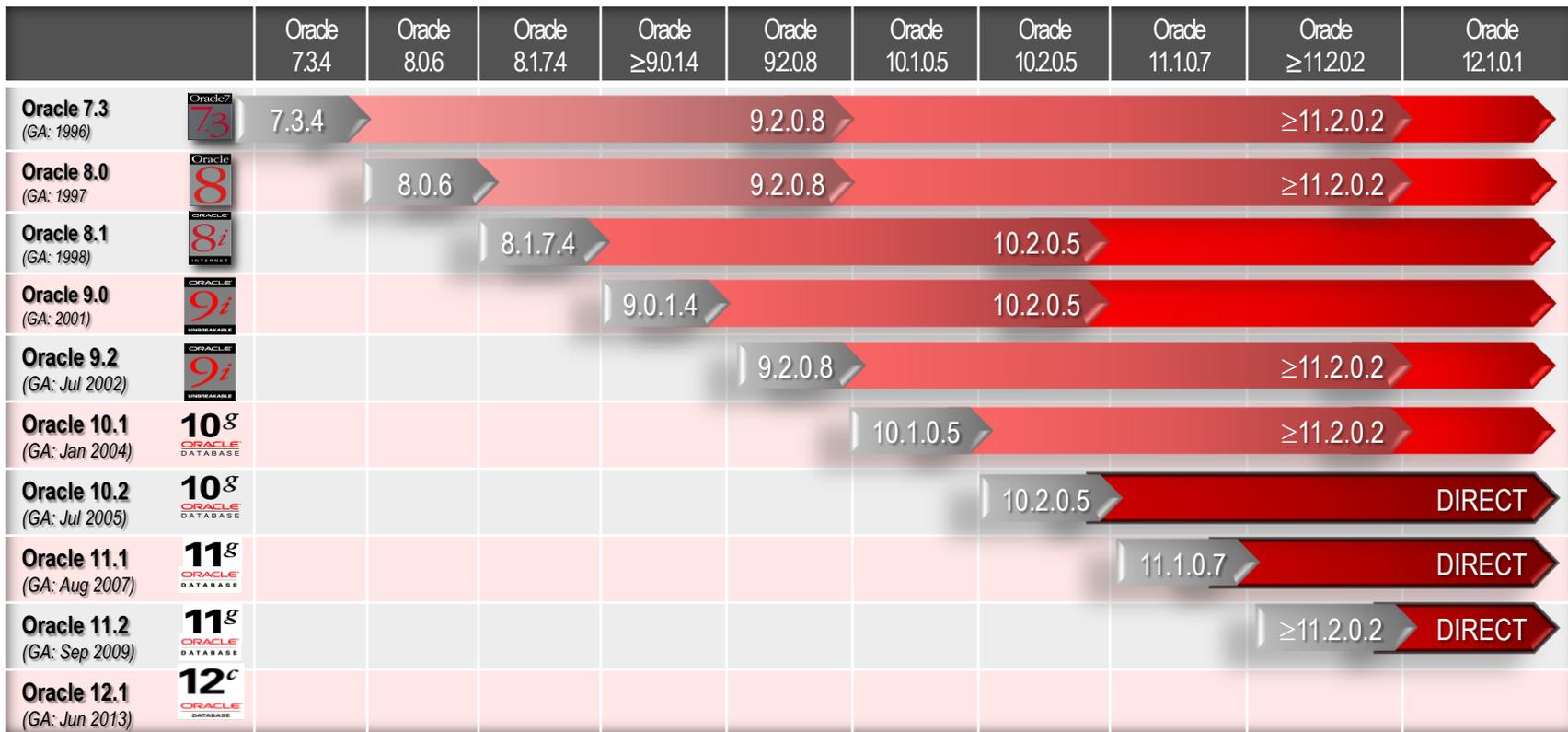
データベース・パッチセットのロードマップ

- [MOS Note:742060.1](#)

Release Schedule of Current Database Releases

リリース	パッチ終了日	例外など
11.2.0.4	31-Jan-2018	HP-UX Itanium は Jan 2020 にパッチ終了予定 Feb 1, 2018 以降は Sev 1 に関する修正のみ (no PSU or CPU will be produced.)
11.2.0.3	27-Aug-2015	
11.2.0.2	31-Oct-2013	28-Feb-2013 から延長
11.2.0.1	13-Sep-2011	Exadata のパッチ終了日は 30-Apr-2012
11.1.0.7	31-Aug-2015	HP-UX Itanium は Dec 2015 にパッチ終了予定 Sep 1, 2015 以降は Sev 1 に関する修正のみ (no PSU or CPU will be produced.)
11.1.0.6	18-Sep-2009	

Upgrade to Oracle Database 12c



Please note: This graph will apply to database upgrades only!

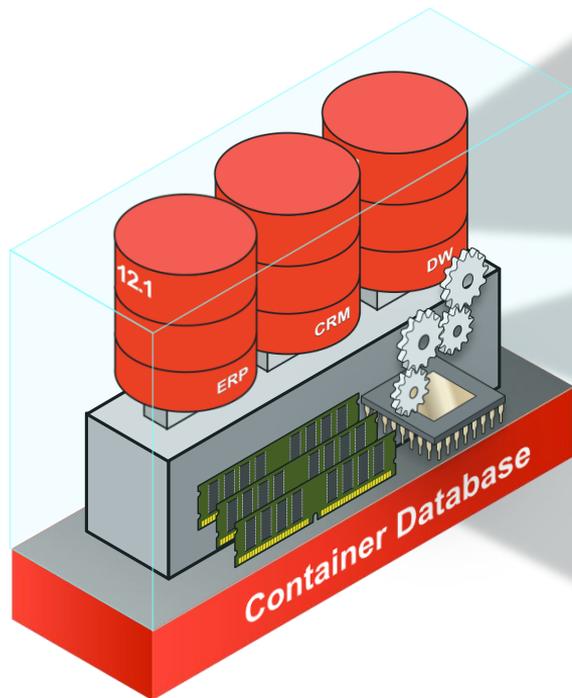


Agenda

- はじめに
- マルチテナント・アーキテクチャ概要
- Oracle Database 12c へのアップグレード事例
- Appendix: 12c 新機能

CONSOLIDATING DATABASES ON CLOUD

Oracle Database 12c のマルチテナント・アーキテクチャ



アプリケーションごとにプラグブル・データベースが使用可能

- アプリケーションへの変更は不要
- クローンを利用した迅速なプロビジョニング
- プラガブル・データベース単位でポータビリティの提供

メモリとバックグラウンド・プロセスの共有

- より多くのアプリケーションを集約

データベースと同様の操作性

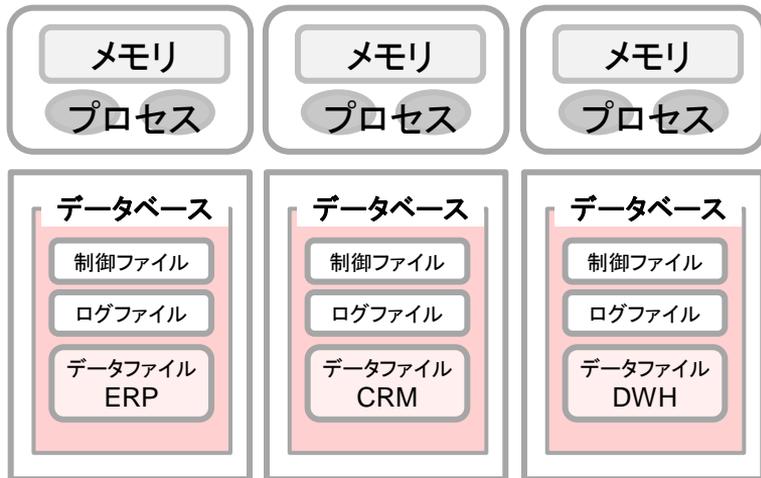
- 単一のデータベースとしての操作性
- プラガブル・データベースごとの操作性

➔ CapEx および OpEx の削減を実現

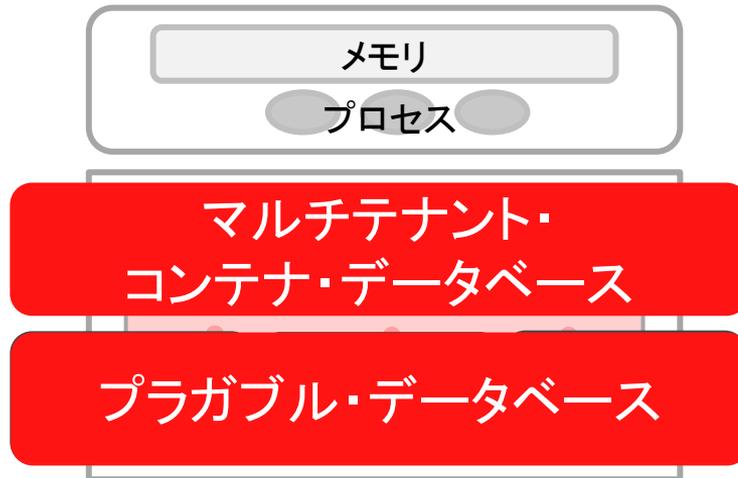
MULTITENANT ARCHITECTURE

アプリケーションをプラグブル・データベースとして構築

Oracle Database 11g



Oracle Database 12c



INCREASE SERVER UTILIZATION

サーバー・リソースの効率的な利用

いままでのサーバ統合

【低密度】

載せられるデータベースは多くない



マルチテナント・アーキテクチャ

【高密度】

より多くのデータベースを稼働できる



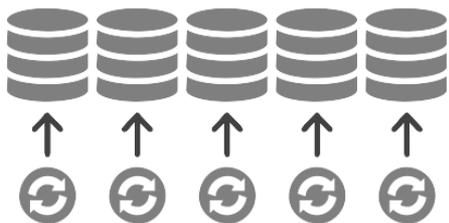
MANAGE MANY DATABASES AS ONE

シンプルなパッチ適用やアップグレード

いままでの運用管理

【データベースが多数】

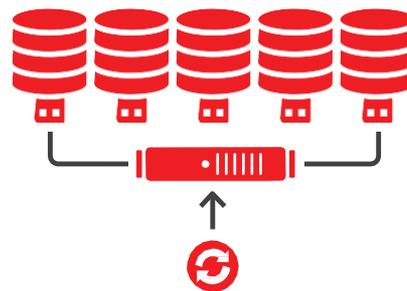
パッチ適用やアップグレードは、それぞれのデータベースで行う



マルチテナント・アーキテクチャ

【データベースは単一】

パッチ適用やアップグレードの対象となるデータベースは1つ



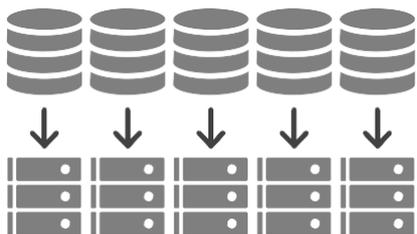
BACKUP MANY DATABASES AS ONE

一元化されたバックアップ

いままでのバックアップ

【システムの数だけ】

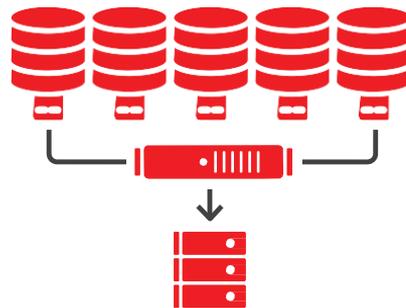
バックアップはシステムごとに取得する



マルチテナント・アーキテクチャ

【1度だけ】

マルチテナント・コンテナ・データベース全体で1つのバックアップを取得する



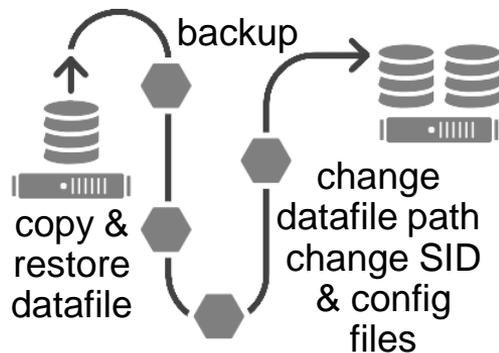
PROVISION DATABASE QUICKLY

短時間で実施する環境構築

いままでのクローニング

【いくつものステップ】

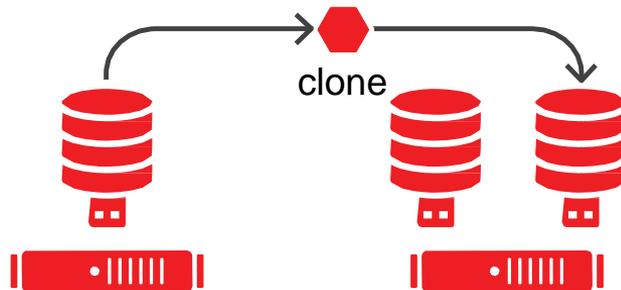
手順が多く時間がかかる



マルチテナント・アーキテクチャ

【ワン・ステップ】

たった1回の手順で完了



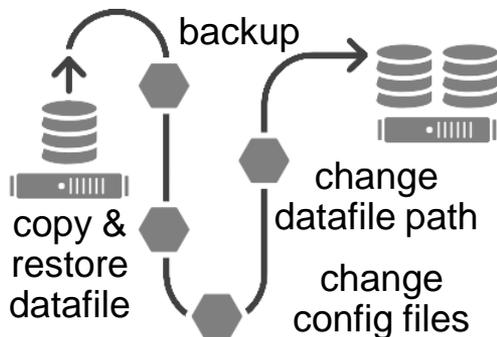
MOVE DATABASE RAPIDLY

シンプルなデータベースの移動

いままでのクローニング

【いくつものステップ】

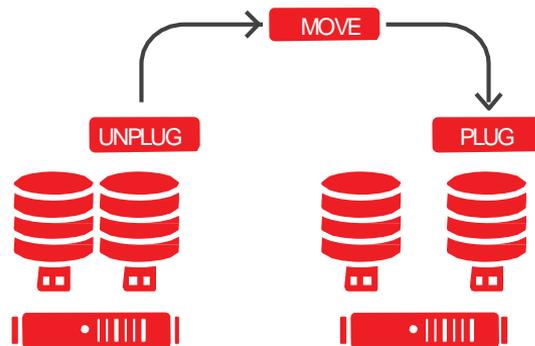
手順や制約が多く複雑



マルチテナント・アーキテクチャ

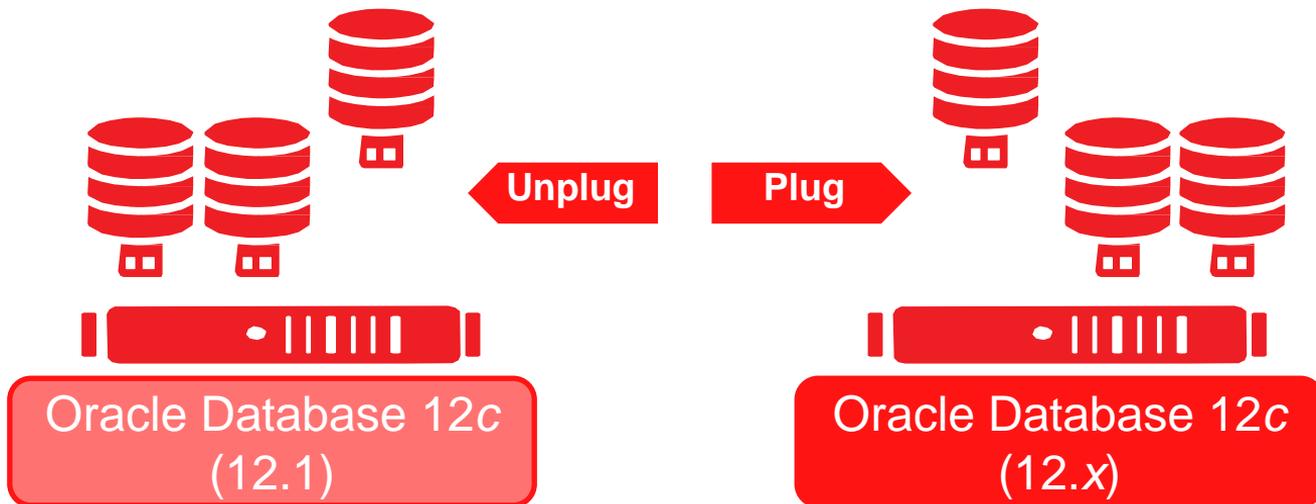
【ワン・ステップ】

"unplug" して "plug" するだけ



SIMPLE UPGRADE IN A SIMPLE PLATFORM

バージョン混在環境でのアップグレード



Oracle Exadata Database Machine

Agenda

- はじめに
- マルチテナント・アーキテクチャ概要
- Oracle Database 12c へのアップグレード事例
- Appendix: 12c 新機能

12c へのアップグレード

12c へのアップグレード

- 新サーバーへの移行を伴うデータベースのアップグレード

実例 & 説明

1. ハードウェアの移行
2. Oracle 10.2.0.5 Linux 32bit からOracle 12.1 Linux 64bit へのアップグレード
3. 最大許容ダウンタイムは24時間
4. データベースのサイズは1TB



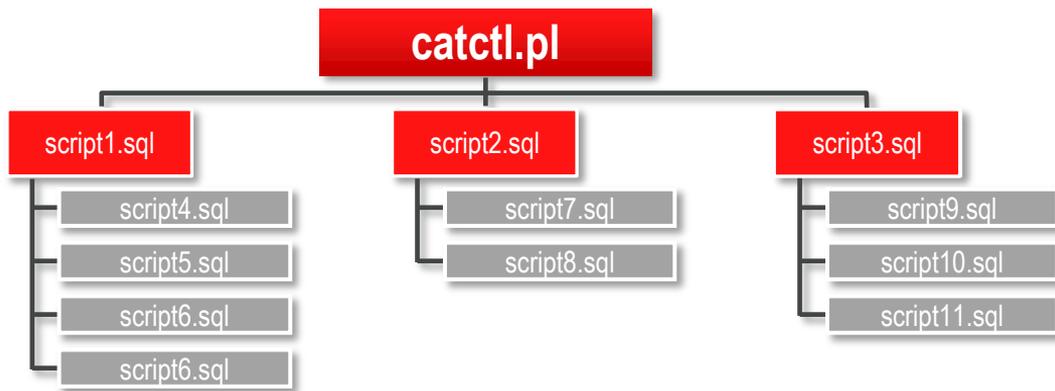
12c へのアップグレード

- 新サーバーへの移行を伴うデータベースのアップグレード



Oracle 12c へのアップグレード最新情報

- SQL を自動化する新しい修正スクリプト
 - アップグレード前後に問題を修正する
- アップグレードの平行化
 - アップグレード時間を最大40% 削減
 - `catupgrd.sql` ではなく `catctl.pl` によるアップグレード



インストールと12c
の適用

アップグレード前
のチェック

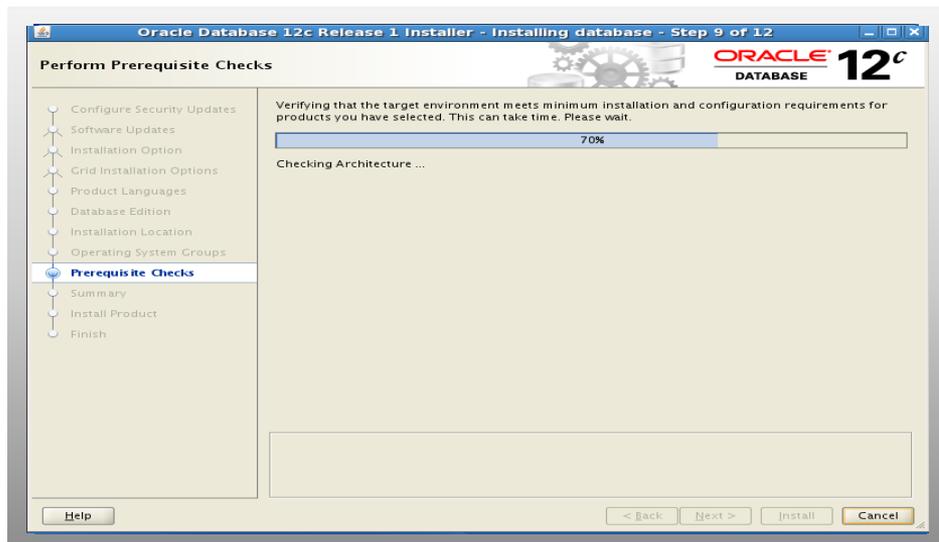
環境の準備

データベース
全体のリストア

データベースの
アップグレード

アップグレード後
の作業

■ Oracle Database 12c のインストール



Oracle 10.2.0.5
RHEL4 – 32bit

Oracle 12.1.0.1
OEL6 – 64bit

ORACLE

インストールと12c
の適用

アップグレード前
のチェック

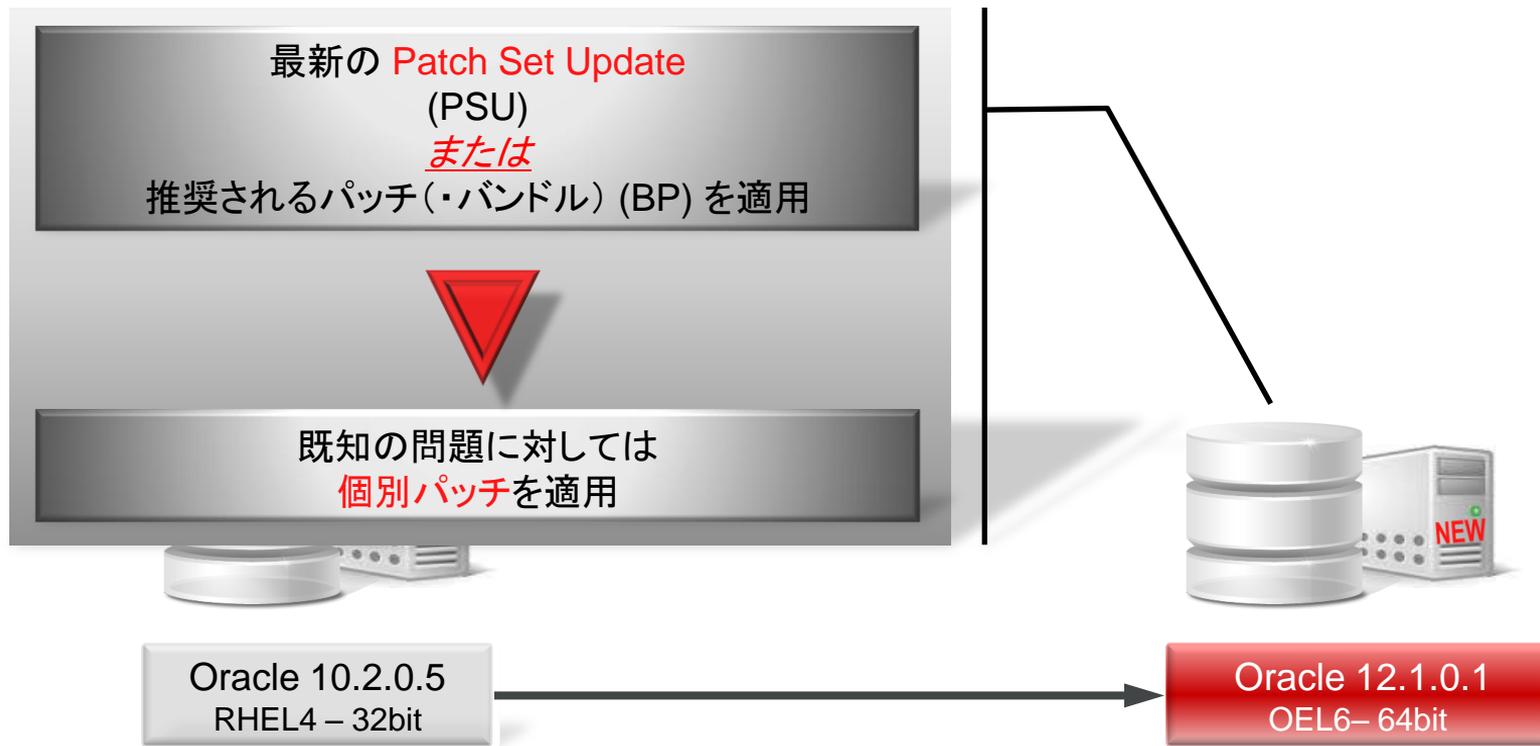
環境の準備

データベース
全体のリストア

データベースの
アップグレード

アップグレード後
の作業

■ 新しいOracle Database 12c ホームの適用



ORACLE

インストールと12c
の適用

アップグレード前
のチェック

環境の準備

データベース
全体のリストア

データベースの
アップグレード

アップグレード後
の作業

■ 新しいアップグレード前情報ツール

- utlu112i.sql から `preupgrd.sql` と `utluppkg.sql` へ置換

- 新しいOracle 12c のホームから、既存のOracle ホームである

`$ORACLE_HOME/rdbms/admin` ^ `*/rdbms/admin/preupgrd.sql` と
`*/rdbms/admin/utluppkg.sql` をコピー

- ソース・データベースで `preupgrd.sql` を実行

- コピーをせずに、Oracle 12c のホームからソース側のSQL*Plus を使って直接
`preupgrd.sql` を実行してもよい



ORACLE

インストールと12c
の適用

アップグレード前
のチェック

環境の準備

データベース
全体のリストア

データベースの
アップグレード

アップグレード後
の作業

- preupgrd.sql によって、
ログが1つと
スクリプトが2つ生成される



Oracle 10.2.0.5
RHEL4 – 32bit

```
SQL> @?/rdbms/admin/preupgrd.sql
```

結果は次のディレクトリに生成される:

```
$ORACLE_BASE/cfgtoollogs/V102/preupgrade
```

preupgrade.log

ソース・データベースに関する検出、推奨、警告、修正、エラー
といった内容について詳細な情報を含む

preupgrade_fixups.sql

ソース・データベースで実行するスクリプト。このスクリプトで
問題を直接修正する、または DBA によって修正を行う

postupgrade_fixups.sql

アップグレード後すぐに実行し、未解決の問題を修正するスクリプト。この
スクリプトには、手動での操作の説明も含む(例: タイムゾーンの変更)

Oracle 12.1.0.1
OEL6 – 64bit

ORACLE

インストールと12c
の適用

アップグレード前
のチェック

環境の準備

データベース
全体のリストア

データベースの
アップグレード

アップグレード後
の作業

- アップグレード前情報ツール `preupgrd.sql` を実行



Oracle 10.2.0.5
RHEL4 – 32bit

```
oracle@localhost.localdomain: /u01/app/oracle/product/12.1.0/rdbms/admin
File Edit View Terminal Tabs Help
Connected to:
oracle@localhost.localdomain: /u01/app/oracle/cfgtoollogs/V102/preupgrade
File Edit View Terminal Tabs Help
[V102] oracle@localhost:/u01/app/oracle/cfgtoollogs/V102/preupgrade
$ ls -lrt
total 16
-rw-r--r-- 1 oracle oinstall 6192 Feb 22 14:44 preupgrade.log
-rw-r--r-- 1 oracle oinstall 2829 Feb 22 14:44 preupgrade_fixups.sql
-rw-r--r-- 1 oracle oinstall 2358 Feb 22 14:44 postupgrade_fixups.sql
[V102] oracle@localhost:/u01/app/oracle/cfgtoollogs/V102/preupgrade
$
/u01/app/oracle/cfgtoollogs/V102/preupgrade/preupgrade_fixups.sql
Post-Upgrade Fixup Script (run shortly after upgrade):
/u01/app/oracle/cfgtoollogs/V102/preupgrade/postupgrade_fixups.sql
*****
Fixup scripts must be reviewed prior to being executed.
*****
```

Oracle 12.1.0.1
OEL6 – 64bit

インストールと12c
の適用

アップグレード前
のチェック

環境の準備

データベース
全体のリストア

データベースの
アップグレード

アップグレード後
の作業

■ log の出力

```
oracle@localhost.localdomain: /u01/app/oracle/cfgtoollogs/V102/preupgrade
File Edit View Terminal Tabs Help
*****
[Post-Upgrade Recommendations]
*****

***** Fixed Object Statistics *****

Please create stats on fixed objects two weeks
after the upgrade using the command:
EXECUTE DBMS_STATS.GATHER_FIXED_OBJECTS_STATS;

^^^ MANUAL ACTION SUGGESTED ^^^

***** Summary *****

1 ERROR exist that must be addressed prior to performing your upgrade.
3 WARNINGS that Oracle suggests are addressed to improve database performance.
1 INFORMATIONAL message that should be reviewed prior to your upgrade.

After your database is upgraded and open in normal mode you must run
rdbms/admin/catuppst.sql which executes several required tasks and completes
```



Oracle 10.2.0.5
RHEL4 – 32bit



Oracle 12.1.0.1
OEL6 – 64bit

インストールと12c
の適用

アップグレード前
のチェック

環境の準備

データベース
全体のリストア

データベースの
アップグレード

アップグレード後
の作業

preupgrade_fixups.sql の実行

```
oracle@localhost.localdomain: /u01/app/oracle/cfgtoollogs/V102/preupgrade
File Edit View Terminal Tabs Help

*****
***** Fixup Summary *****

1 fixup routine was successful.
1 fixup routine returned INFORMATIONAL text that should be reviewed.
1 ERROR LEVEL check returned INFORMATION that must be acted on prior to upgrade

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

1) Check Tag: COMPATIBLE_PARAMETER failed.
Check Summary: Verify compatible parameter value is valid
Fixup Summary:
""compatible" parameter must be increased manually prior to upgrade."
^^^ MANUAL ACTION REQUIRED ^^^

*****
You MUST resolve the above error prior to upgrade
*****
```

Oracle 10.2.0.5
RHEL4 – 32bit

Oracle 12.1.0.1
OEL6 – 64bit

ORACLE

インストールと12c
の適用

アップグレード前
のチェック

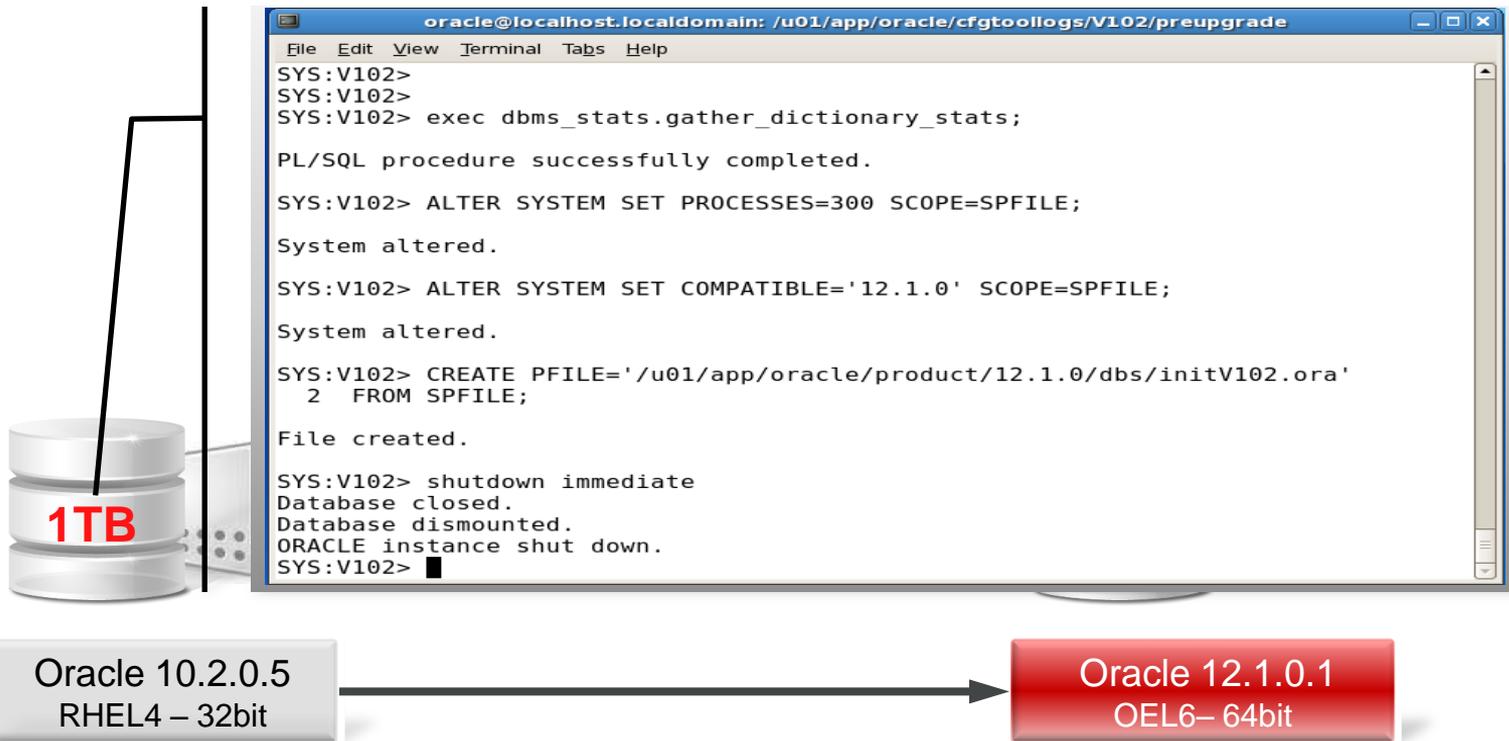
環境の準備

データベース
全体のリストア

データベースの
アップグレード

アップグレード後
の作業

必要とされる手動操作



ORACLE

インストールと12c
の適用

アップグレード前
のチェック

環境の準備

データベース
全体のリストア

データベースの
アップグレード

アップグレード後
の作業

追加の手動操作を推奨する場合がある

– Enterprise Manager Database Control リポジトリの削除

```
$> emctl stop dbconsole
```

```
SQL> @$/rdbms/admin/emremove.sql
```



▪ EM DB Control は12c で EM Express に置き換えられている

– リポジトリは不要

– EM のリポジトリが削除されていない場合、アップグレード後の `catuppst.sql` 実行中に自動的に削除される



Oracle 10.2.0.5
RHEL4 – 32bit



Oracle 12.1.0.1
OEL6 – 64bit

ORACLE

インストールと12c
の適用

アップグレード前
のチェック

環境の準備

データベース
全体のリストア

データベースの
アップグレード

アップグレード後
の作業

追加の手動操作を推奨する場合がある

- もしアップグレード前のデータベースに、OLS (Label Security) および/またはDV (Database Vault) が使用されていれば追加で手順を実行

- SQL> @?/rdbms/admin/olspreupgrade.sql



- SYSTEM からSYSへAUD\$ 表を移動する準備をする

- 停止時間を最小化するために監査レコードを移動する

- レコードを暫定的に一時表へ移動

- DV についても同様の操作を適宜行う(ドキュメントを参照)



Oracle 10.2.0.5
RHEL4 – 32bit



Oracle 12.1.0.1
OEL6 – 64bit

ORACLE

インストールと12c
の適用

アップグレード前
のチェック

環境の準備

データベース
全体のリストア

データベースの
アップグレード

アップグレード後
の作業

■ 新しいinit.ora を編集

```
oracle@localhost.localdomain: /u01/app/oracle/product/12.1.0/db5  
File Edit View Terminal Tabs Help  
*.compatible='12.1.0'  
*.control_files='/oradata/V102/control01.ctl', '/oradata/V102/control02.ctl', '/oradata/V102/control03.ctl'  
*.db_block_size=8192  
*.db_domain=''  
*.db_name='V102'  
*.job_queue_processes=10  
*.open_cursors=300  
*.pga_aggregate_target=47185920  
*.processes=300  
*.remote_login_passwordfile='EXCLUSIVE'  
*.sessions=92  
*.session_cached_cursors=200  
*.sga_target=419430400  
*.undo_management='AUTO'  
*.undo_tablespace='UNDOTBS1'
```



Oracle 10.2.0.5
RHEL4 – 32bit



Oracle 12.1.0.1
OEL6 – 64bit



■ パスワード・ファイルの新規作成

```
$> cd $ORACLE_HOME/dbs  
$> orapwd file=orapwV102  
password=<string> entries=15  
format=12 force=yes
```



Oracle 10.2.0.5
RHEL4 – 32bit



Oracle 12.1.0.1
OEL6 – 64bit

インストールと12c
の適用

アップグレード前
のチェック

環境の準備

データベース
全体のリストア

データベースの
アップグレード

アップグレード後
の作業

■ 環境

12.1 の環境へ設定

ORACLE_BASE, ORACLE_HOME, ORACLE_SID, TNS_ADMIN, TMP およ
び TMPDIR を設定する

Windows のみ

サービスを新規作成:



```
oradim -NEW -SID ORASID -SYSPWD passwd -STARTMODE a -  
PFILE initfile
```



Oracle 10.2.0.5
RHEL4 – 32bit



Oracle 12.1.0.1
OEL6 – 64bit



ORACLE

インストールと12c
の適用

アップグレード前
のチェック

環境の準備

データベース
全体のリストア

データベースの
アップグレード

アップグレード後
の作業

リスナーの構成



接続クライアントが 11g 以下の場合は設定が必要
\$ORACLE_HOME/network/admin/**sqlnet.ora** に
SQLNET.ALLOWED_LOGON_VERSION_SERVER=10

Oracle 10.2.0.5
RHEL4 – 32bit

Oracle 12.1.0.1
OEL6 – 64bit



ORACLE

インストールと12c
の適用

アップグレード前
のチェック

環境の準備

データベース
全体のリストア

データベースの
アップグレード

アップグレード後
の作業

■ RMAN オンライン・バックアップでリストア・リカバリ

```
RMAN> run { SET ARCHIVELOG DESTINATION TO '/tmp/tmprest';  
            RESTORE DATABASE;  
            RECOVER DATABASE; }
```

- TEMP 表領域は新規作成する
- または新しいサーバーへすべての関連ファイルをコピー
 - データファイル、REDO ログ、制御ファイル



ORACLE

インストールと12c
の適用

アップグレード前
のチェック

環境の準備

データベース
全体のリストア

データベースの
アップグレード

アップグレード後
の作業

- 同一OSプラットフォーム、かつ**32bit → 64bit**の変更
 - ワードサイズの変更は**アップグレード作業の一部として自動的に実行**

移行元

```
startup upgrade
@?/rdbms/admin/utlirp.sql
shutdown immediate
```

移行先

```
startup upgrade
@?/rdbms/admin/utlirp.sql
```



Oracle 10.2.0.5
RHEL4 – 32bit



Oracle 12.1.0.1
OEL6 – 64bit

ORACLE

インストールと12c
の適用

アップグレード前
のチェック

環境の準備

データベース
全体のリストア

データベースの
アップグレード

アップグレード後
の作業

■ データベースをSTARTUP UPGRADE モードで起動

```
oracle@localhost.localdomain: /u01/app/oracle/product/12.1.0/dbs
File Edit View Terminal Tabs Help
[CDB12] oracle@localhost:/u01/app/oracle/product/12.1.0/dbs
$ sqlplus / as sysdba

SQL*Plus: Release 12.1.0.1.0 Production on Mon Feb 25 19:57:09 2013

Copyright (c) 1982, 2013, Oracle. All rights reserved.

Connected to an idle instance.

SYS:V102> startup upgrade pfile=initV102.ora
ORACLE instance started.

Total System Global Area 417546240 bytes
Fixed Size 2288432 bytes
Variable Size 142607568 bytes
Database Buffers 264241152 bytes
Redo Buffers 8409088 bytes
Database mounted.
Database opened.
SYS:V102> exit
Disconnected from Oracle Database 12c Enterprise Edition Release 12.1.0.1.0 - 64
bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application Testing opt
ions
```

Oracle 10.2.0.5
RHEL4 – 32bit

Oracle 12.1.0.1
OEL6 – 64bit

ORACLE

インストールと12c
の適用

アップグレード前
のチェック

環境の準備

データベース
全体のリストア

データベースの
アップグレード

アップグレード後
の作業

- STARTUP UPGRADE モード中に何が起きるのか？
 - ORA-942 といった20以上のエラー・メッセージを抑制する

STARTUP UPGRADE モード中における alert.log からの抜粋

```
ALTER SYSTEM enable restricted session;  
ALTER SYSTEM SET _system_trig_enabled=FALSE SCOPE=MEMORY;  
Autotune of undo retention is turned off.  
ALTER SYSTEM SET _undo_autotune=FALSE SCOPE=MEMORY;  
ALTER SYSTEM SET undo_retention=900 SCOPE=MEMORY;  
ALTER SYSTEM SET aq_tm_processes=0 SCOPE=MEMORY;  
ALTER SYSTEM SET enable_ddl_logging=FALSE SCOPE=MEMORY;  
Resource Manager disabled during database migration: plan '' not set  
ALTER SYSTEM SET resource_manager_plan= SCOPE=MEMORY;  
ALTER SYSTEM SET recyclebin='OFF' DEFERRED SCOPE=MEMORY;  
Resource Manager disabled during database migration  
replication_dependency_tracking turned off (no async multimaster replication  
found)  
AQ Processes can not start in restrict mode  
Completed: ALTER DATABASE OPEN MIGRATE
```



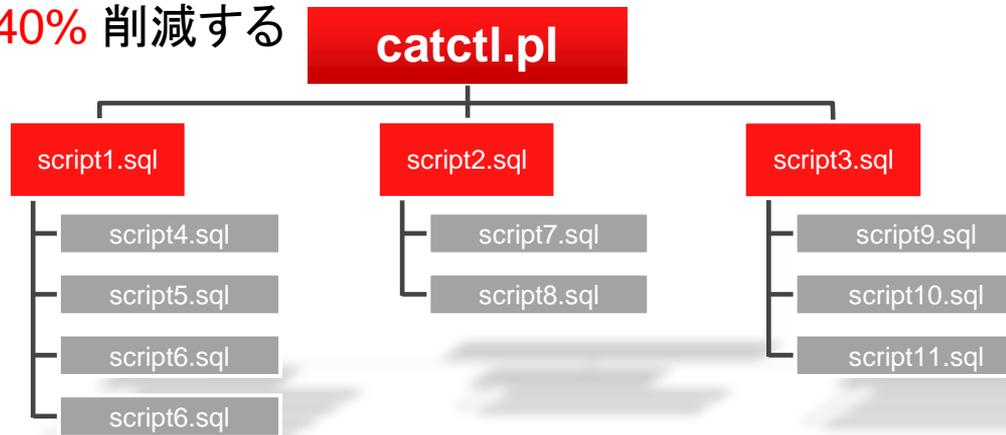
Oracle 10.2.0.5
RHEL4 – 32bit

Oracle 12.1.0.1
OEL6 – 64bit

ORACLE

■ 新しい**パラレル**・アップグレード・スクリプト

- Oracle Database のデフォルト機能
- 実行可能な状況においては、SERVER コンポーネントのアップグレード・スクリプトは**パラレル**で実行される
- その他の関連性のないコンポーネントに関するしても**パラレル**で実行する
- アップグレードの総時間を**20%** から**40%** 削減する
- **パラレル・アップグレード**は Perl スクリプトにより実行



インストールと12c
の適用

アップグレード前
のチェック

環境の準備

データベース
全体のリストア

データベースの
アップグレード

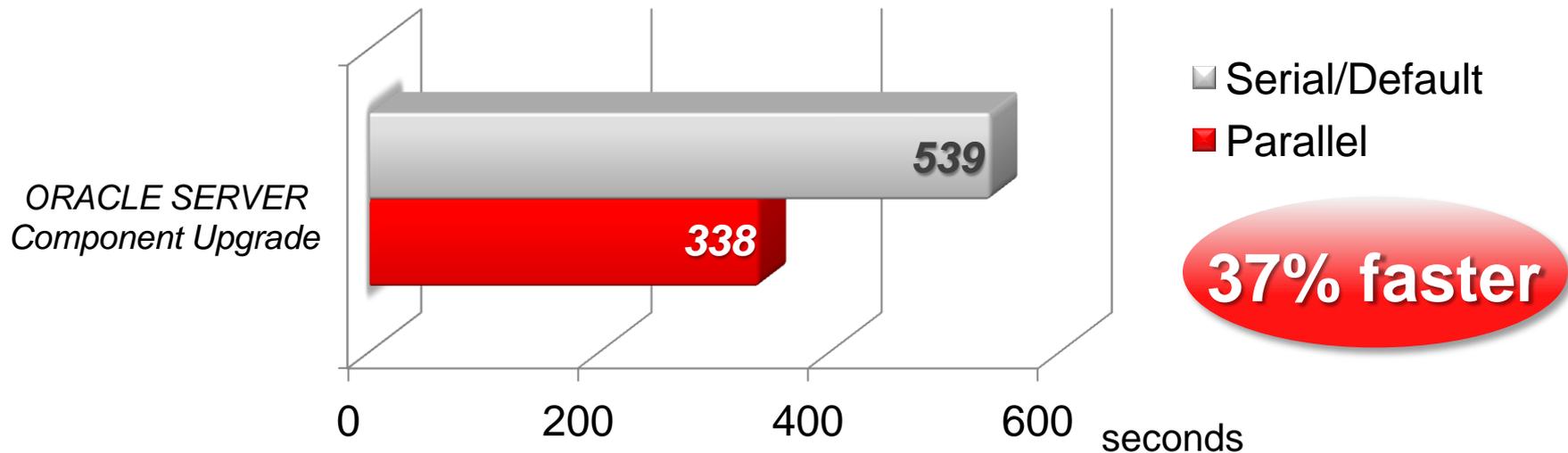
アップグレード後
の作業

パラレル・アップグレード - 顧客事例

■ Payback GmbH, Germany



Oracle 11.1.0.7 から Oracle 11.2.0.3 の Exadata X2-2
へアップグレード



ORACLE

インストールと12c
の適用

アップグレード前
のチェック

環境の準備

データベース
全体のリストア

データベースの
アップグレード

アップグレード後
の作業

■ Oracle 12c へのアップグレード時間に関わる主な要因

- インストール済みのコンポーネントとデータベース・オプションの数
- 大量の新しいディクショナリによるデータベース・オブジェクトの数、およびSYSTEM 表の再構築
- CPU の性能

Component	HH:MM:SS
Oracle Server	00:16:17
JSERVER JAVA Virtual Machine	00:05:19
Oracle Workspace Manager	00:01:01
Oracle Enterprise Manager	00:10:13
Oracle XDK	00:00:48
Oracle Text	00:00:58
Oracle XML Database	00:04:09
Oracle Database Java Packages	00:00:33
Oracle Multimedia	00:07:43
Oracle Expression Filter	00:00:18
Oracle Rule Manager	00:00:12
Gathering Statistics	00:04:53
Total Upgrade Time: 00:52:31	

Component	HH:MM:SS
Oracle Server	00:16:17
JSERVER JAVA Virtual Machine	00:05:19
Oracle XDK	00:00:48
Oracle Text	00:00:58
Oracle XML Database	00:04:09
Oracle Database Java Packages	00:00:33
Gathering Statistics	00:02:43
Total Upgrade Time: 00:30:47	

■ アップグレードをスピードアップさせる

- アップグレードの**前夜**にディクショナリの統計を取得する

- SQL>

```
exec DBMS_STATS.GATHER_DICTIONARY_STATS;
```

- XDB 使用時は、特定の XDB 向けに統計を取得する

- SQL>

```
exec DBMS_STATS.GATHER_TABLE_STATS(ownname=>'XDB',  
tabname=>'XDB$RESOURCE', estimate_percent=>NULL);
```

- 監査が有効である(であった)場合、AUD\$ 表の監査レコードをチェック

- [Note: 1329590.1](#) で監査レコードの事前作業を行うスクリプトを提供
 - 監査が有効である(であった)場合

- SQL>

```
truncate table SYS.AUD$ | SYSTEM.AUD$;
```

- Oracle APEX 使用時は、別途アップグレードすることを検討する

- 参照 [Note:1088970.1](#): Master Note for Oracle APEX Upgrades

インストールと12c
の適用

アップグレード前
のチェック

環境の準備

データベース
全体のリストア

データベースの
アップグレード

アップグレード後
の作業

catctl.pl でアップグレードを平行で開始する

```
oracle@localhost.localdomain: /u01/app/oracle/product/12.1.0/rdbms/admin
File Edit View Terminal Tabs Help
$ cd $ORACLE_HOME/rdbms/admin
[CDB12] oracle@localhost:/u01/app/oracle/product/12.1.0/rdbms/admin
$ $ORACLE_HOME/perl/bin/perl catctl.pl -n 3 catupgrd.sql

Analyzing file catupgrd.sql
14 scripts found in file catupgrd.sql
Next path: catalog.sql
32 scripts found in file catalog.sql
Next path: catproc.sql
37 scripts found in file catproc.sql
Next path: catptabs.sql
61 scripts found in file catptabs.sql
Next path: catpdbms.sql
205 scripts found in file catpdbms.sql
Next path: catpdeps.sql
77 scripts found in file catpdeps.sql
Next path: catpprvt.sql
260 scripts found in file catpprvt.sql
Next path: catpexec.sql
26 scripts found in file catpexec.sql
Next path: cmpupgrd.sql
16 scripts found in file cmpupgrd.sql

[Phase 0] type is 1 with 1 Files
```

- catctl.pl はPERL スクリプト
- アップグレード・スクリプトを可能であれば平行で実行する
 - catuppst.sql と同様に catbundle.sql も実行する
- -n y で平行度を定義する
- 平行度の最大値は 8

Oracle 10.2.0.5
RHEL4 – 32bit

Oracle 12.1.0.1
OEL6 – 64bit



インストールと12c
の適用

アップグレード前
のチェック

環境の準備

データベース
全体のリストア

データベースの
アップグレード

アップグレード後
の作業

catctl.pl 実行中の進捗確認

```
oracle@localhost.localdomain: /u01/app/oracle/product/12.1.0/rdbms/admin
File Edit View Terminal Tabs Help
Serial Phase #:36 Files: 4 Time: 103s
Restart Phase #:37 Files: 1 Time: 2s
Parallel Phase #:38 Files: 13 Time: 150s
Restart Phase #:39 Files: 1 Time: 0s
Parallel Phase #:40 Files: 10 Time: 37s
Restart Phase #:41 Files: 1 Time: 0s
Serial Phase #:42 Files: 1 Time: 13s
Restart Phase #:43 Files: 1 Time: 0s
Serial Phase #:44 Files: 1 Time: 79s
Serial Phase #:45 Files: 1 Time: 6s
Restart Phase #:46 Files: 1 Time: 0s
Serial Phase #:47 Files: 2 Time: 715s
Restart Phase #:48 Files: 1 Time: 2s
Serial Phase #:49 Files: 2 Time: 541s
Restart Phase #:50 Files: 1 Time: 1s
Serial Phase #:51 Files: 2 Time: 22s
Restart Phase #:52 Files: 1 Time: 0s
Serial Phase #:53 Files: 1 Time: 4s
Serial Phase #:54 Files: 1 Time: 390s
Serial Phase #:55 Files: 1 Time: 297s
Serial Phase #:56 Files: 1 Time: 28s
Grand Total Time: 3415s
```



Oracle 10.2.0.5
RHEL4 – 32bit

Oracle 12.1.0.1
OEL6 – 64bit



インストールと12c
の適用

アップグレード前
のチェック

環境の準備

データベース
全体のリストア

データベースの
アップグレード

アップグレード後
の作業

catctl.pl のログ・ファイルのデフォルト配置場所

- 現在の作業ディレクトリ
 - ヒント: `-l parameter` を使用して特定する
- パラレル実行のスレッドごとに生成される

```
6876330 Feb 25 21:04 catupgrd2.log
6626849 Feb 25 21:04 catupgrd1.log
16983150 Feb 25 21:04 catupard0.log
SYS:V102> SYS:V102> 2
CATCTL_TIMESTAMP
-----
-----
Started: catshutdown.sql on CPU: 0 at 13-02-25 09:03:37
SYS:V102> Database closed.
Database dismounted.
ORACLE instance shut down.
SYS:V102> SYS:V102>
SYS:V102> Disconnected from Oracle Database 12c Enterprise
0.1.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Ap
ions
□===== PROCESS ENDED =====
```

Oracle 10.2.0.5
RHEL4 – 32bit

Oracle 12.1.0.1
OEL6 – 64bit

ORACLE

インストールと12c
の適用

アップグレード前
のチェック

環境の準備

データベース
全体のリストア

データベースの
アップグレード

アップグレード後
の作業

■ NORMAL モードでデータベースを再起動

– 起動に使用しているinit.ora からspfile を作成する

```
oracle@localhost.localdomain: /u01/app/oracle/product/12.1.0/dbs
File Edit View Terminal Tabs Help
Copyright (c) 1982, 2013, Oracle. All rights reserved.
Connected to an idle instance.
SYS:V102> startup pfile=initV102.ora
ORACLE instance started.

Total System Global Area  417546240 bytes
Fixed Size                  2288432 bytes
Variable Size               142607568 bytes
Database Buffers            264241152 bytes
Redo Buffers                 8409088 bytes
Database mounted.
Database opened.
SYS:V102> create spfile from pfile;
File created.
```



Oracle 10.2.0.5
RHEL4 – 32bit



Oracle 12.1.0.1
OEL6 – 64bit

インストールと12c
の適用

アップグレード前
のチェック

環境の準備

データベース
全体のリストア

データベースの
アップグレード

アップグレード後
の作業

utlprp.sql による再コンパイル

- SQL> @utlprp n を直接実行してCPU 使用率の設定変更が可能

▪ 最も効果的な値はCPU コア数 x 2

- 再コンパイルの時間を最小化するために固定表の統計情報を取得する

▪ SQL> exec DBMS_STATS.GATHER_FIXED_OBJECTS_STATS;

```
oracle@localhost.localdomain: ~
File Edit View Terminal Tabs Help
SYS:V102> exec dbms_stats.gather_fixed_objects_stats;
PL/SQL procedure successfully completed.
Elapsed: 00:00:31.76
SYS:V102> @?/rdbs/admin/utlprp.sql
TIMESTAMP
-----
COMP_TIMESTAMP UTLRP_BGN 2013-02-28 19:06:57
```



Oracle 10.2.0.5
RHEL4 – 32bit

Oracle 12.1.0.1
OEL6 – 64bit

インストールと12c
の適用

アップグレード前
のチェック

環境の準備

データベース
全体のリストア

データベースの
アップグレード

アップグレード後
の作業

■ utlrp.sql 実行中の確認

1. コンパイルが必要なオブジェクト

```
SELECT COUNT(*) FROM obj$  
WHERE status IN (4, 5, 6);
```

2. すでにコンパイルされたオブジェクト

```
SELECT COUNT(*) FROM UTL_RECOMP_COMPILED;
```

3. UTL_RECOMPによって作成されたパラレル・ジョブ

```
SELECT job_name FROM dba_scheduler_jobs  
WHERE job_name like 'UTL_RECOMP_SLAVE_%';
```

4. パラレルで実行中のジョブ

```
SELECT job_name FROM dba_scheduler_running_jobs  
WHERE job_name like 'UTL_RECOMP_SLAVE_%';
```



Oracle 10.2.0.5
RHEL4 – 32bit



Oracle 12.1.0.1
OEL6 – 64bit

ORACLE

インストールと12c
の適用

アップグレード前
のチェック

環境の準備

データベース
全体のリストア

データベースの
アップグレード

アップグレード後
の作業

■ postupgrade_fixups.sql を実行

```
oracle@localhost.localdomain: /u01/app/oracle/cfgtoollogs/V102/preupgrade
File Edit View Terminal Tabs Help
SYS:V102> @/u01/app/oracle/cfgtoollogs/V102/preupgrade/postupgrade_fixups.sql
Post Upgrade Fixup Script Generated on 2013-02-22 14:44:43 Version: 12.1.0.1.8u
Build: 006
Beginning Post-Upgrade Fixups...
PL/SQL procedure successfully completed.

*****
Check Tag:      OLD_TIME_ZONES_EXIST
Check Summary:  Check for use of older timezone data file
Fix Summary:    Update the timezone using the DBMS_DST package after upgrade is complete.
*****

Fixup Returned Information:
INFORMATION: --> Older Timezone in use

Database is using a time zone file older than version 18.
After the upgrade, it is recommended that DBMS_DST package
be used to upgrade the 12.1.0.1.0 database time zone version
to the latest version which comes with the new release.
Please refer to My Oracle Support note number 1509653.1 for details.
*****
```

Oracle 10.2.0.5
RHEL4 – 32bit

Oracle 12.1.0.1
OEL6 – 64bit



インストールと12c
の適用

アップグレード前
のチェック

環境の準備

データベース
全体のリストア

データベースの
アップグレード

アップグレード後
の作業

- アップグレードのステータス確認を実行: utlu121s.sql

```
oracle@localhost.localdomain: ~  
File Edit View Terminal Tabs Help  
SYS:V102- @~/rdbms/admin/utlu121s.sql  
Oracle Database 12.1 Post-Upgrade Status Tool          02-28-2013 14:38:12  
Component Name          Current Status      Version Number      Elapsed Time  
                          HH:MM:SS  
Oracle Server            VALID              12.1.0.1.0          00:23:23  
JServer JAVA Virtual Machine  
                          VALID              12.1.0.1.0          00:07:46  
Oracle Workspace Manager  
                          VALID              12.1.0.1.0          00:01:32  
Oracle XDK                VALID              12.1.0.1.0          00:04:07  
Oracle XML Database      VALID              12.1.0.1.0          00:06:50  
Oracle Database Java Packages  
                          VALID              12.1.0.1.0          00:00:36  
Final Actions  
                          00:05:53  
Total Upgrade Time: 00:50:44
```



Oracle 10.2.0.5
RHEL4 – 32bit



Oracle 12.1.0.1
OEL6 – 64bit

インストールと12c
の適用

アップグレード前
のチェック

環境の準備

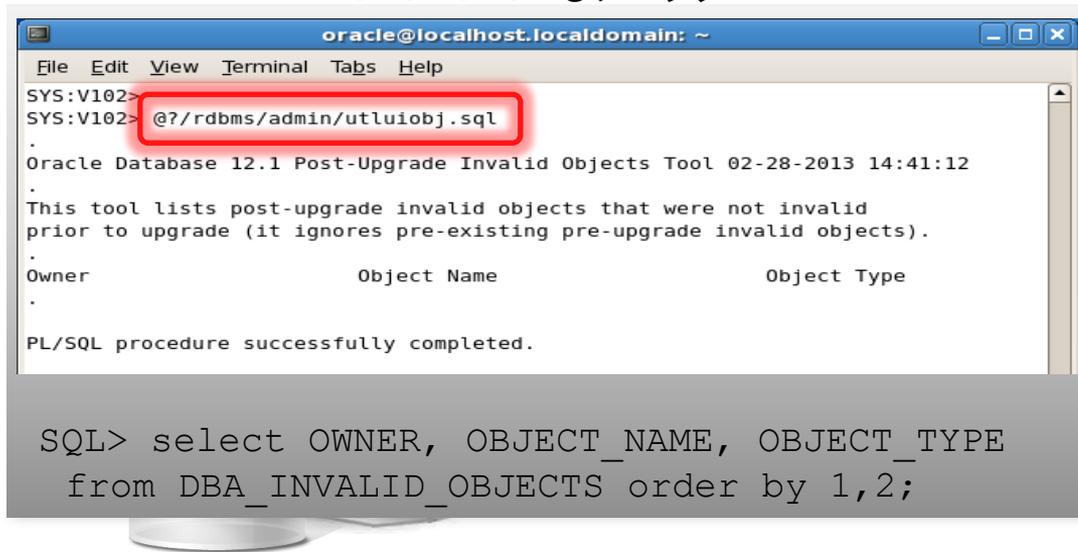
データベース
全体のリストア

データベースの
アップグレード

アップグレード後
の作業

- invalid オブジェクトのチェックを実行: utluiobj.sql

- DBA INVALID OBJECTS もチェック



```
oracle@localhost.localdomain: ~
File Edit View Terminal Tabs Help
SYS:V102>
SYS:V102> @?/rdbsms/admin/utluiobj.sql
.
Oracle Database 12.1 Post-Upgrade Invalid Objects Tool 02-28-2013 14:41:12
.
This tool lists post-upgrade invalid objects that were not invalid
prior to upgrade (it ignores pre-existing pre-upgrade invalid objects).
.
Owner                Object Name          Object Type
.
PL/SQL procedure successfully completed.

SQL> select OWNER, OBJECT_NAME, OBJECT_TYPE
       from DBA_INVALID_OBJECTS order by 1,2;
```

Oracle 10.2.0.5
RHEL4 – 32bit

Oracle 12.1.0.1
OEL6 – 64bit

ORACLE

■ タイムゾーン調節

- タイムゾーン変更はアップグレード後にすべき
- データ型 "TIMESTAMP WITH TIME ZONE" が使われている場合は必須
- **ルール:** $TZ_{Destination} \geq TZ_{Source}$
- タイムゾーン定義: `$ORACLE_HOME/oracore/zoneinfo`

Oracle Database Release	Default Time Zone Vers.
10.2.0.3 – 10.2.0.5	DST V4
11.1.0.6 – 11.1.0.7	DST V4
11.2.0.1	DST V11
11.2.0.2 – 11.2.0.3	DST V14
12.1.0.1	DST V18
最新パッチは MOS Note:412160.1 を確認	DST V20

インストールと12c
の適用

アップグレード前
のチェック

環境の準備

データベース
全体のリストア

データベースの
アップグレード

アップグレード後
の作業

■ タイムゾーン調節

[MOS Note: 1509653.1](#) のスクリプトを利用

(or [MOS Note: 977512.1](#): Updating the RDBMS DST version in 11.2)

```
oracle@localhost.localdomain: ~  
File Edit View Terminal Tabs Help  
SYS:V102> @/home/oracle/DST/DST_adjust.sql  
Connected.  
Database closed.  
Database dismounted.  
ORACLE instance shut down.  
ORACLE instance started.  
  
Total System Global Area 417546240 bytes  
Fixed Size 2288432 bytes  
Variable Size 142607568 bytes  
Database Buffers 264241152 bytes  
Redo Buffers 8409088 bytes  
Database mounted.  
Database opened.  
  
PROPERTY_NAME  
-----  
VALUE  
-----  
DST_PRIMARY_TT_VERSION
```

データベースの
リスタート

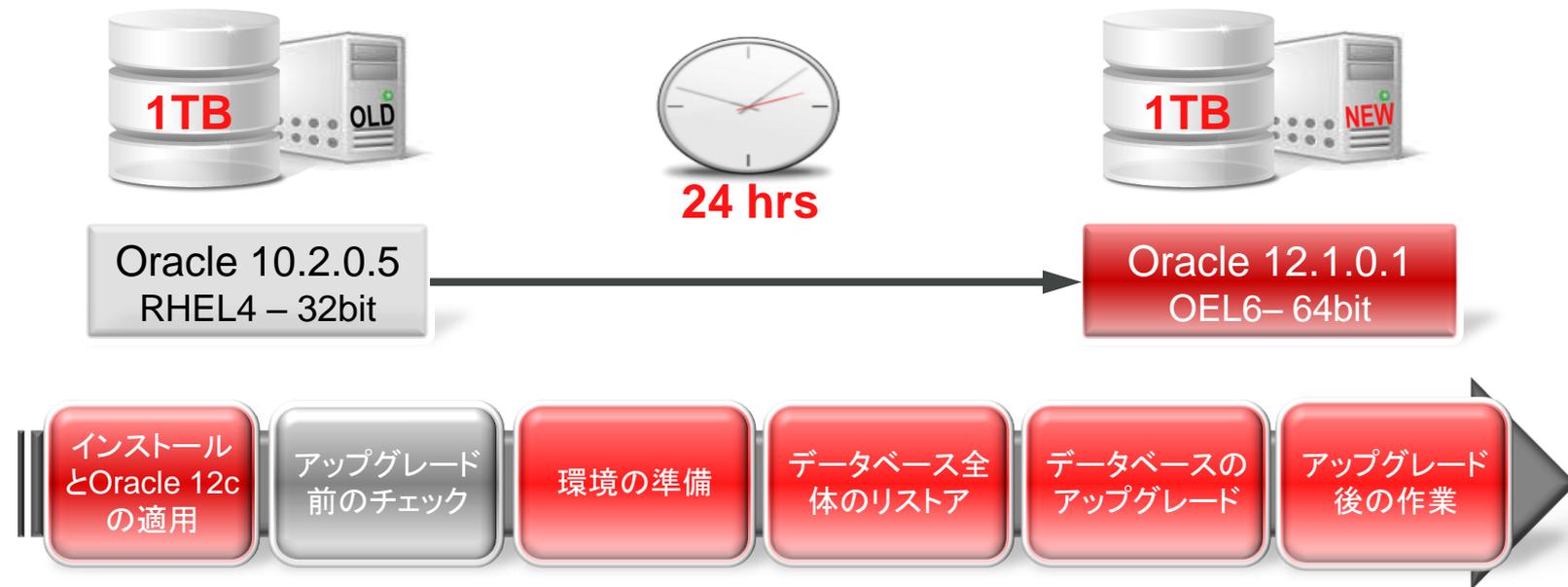
Oracle 10.2.0.5
RHEL4 – 32bit

Oracle 12.1.0.1
OEL6 – 64bit

ORACLE

サマリー

- 新サーバーへの移行を含むデータベース・アップグレード



アップグレードと CDB へのプラグ

アップグレードとCDB へのプラグ

- 12c にアップグレードしたデータベースをCDB に統合

実例 & 説明

1. ソース・データベースは、Oracle Database 12c へ事前にアップグレード
2. ソース・データベースを新しいPDB として、既存のCDB にプラグイン
3. 最大許容停止時間: 15分
4. Database size: 1TB



ORACLE

マルチテナント環境の概要

- マルチテナント環境の紹介
- プラガブル・データベースへの道
- プラガブル・データベースの扱い
- ラップアップ



マルチテナント環境 – 概要

- 統合による物理コストの削減
- メンテナンス・コストの削除
- 簡単で高速なプロビジョニング、クローニング、移行
- オプション名はOracle Multitenant



プラグブル・データベースの選択

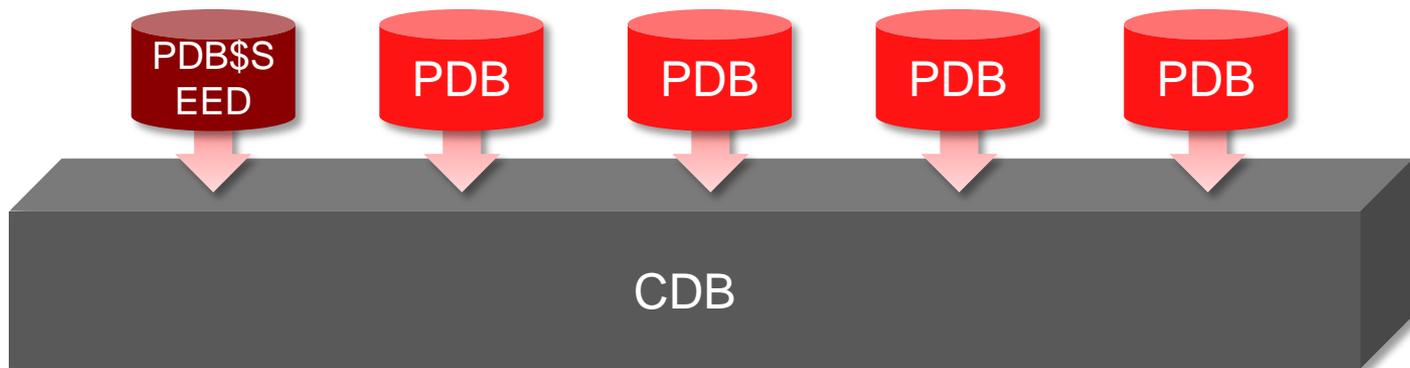
- プラグブル・データベースはまだ使わなくても良いが...
- Oracle Database 12c non-CDB は、これまで通り動く
 - 新しい機能も使えますが、そのためには、古い習慣を変える必要があります

PDB? NON-CDB?



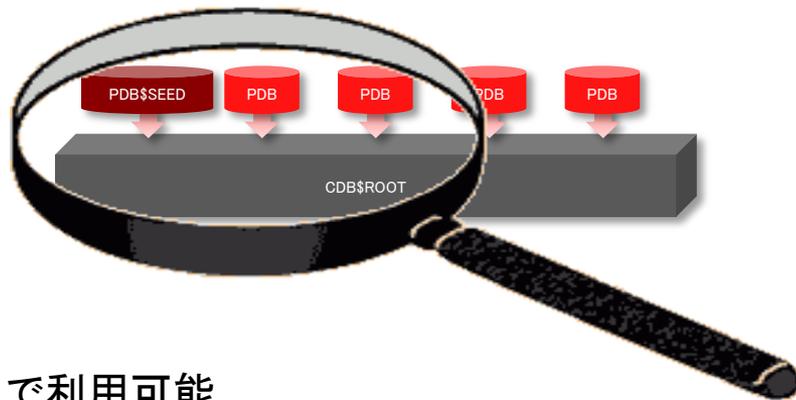
導入

- マルチテナント・コンテナ・データベース => CDB
 - CDB レベルの DBA 管理者
- CDB には、複数のプラグابل・データベースを含む
 - アプリケーションは、各PDB に接続
 - 1つの CDB で最大253 個のPDB を作成可能



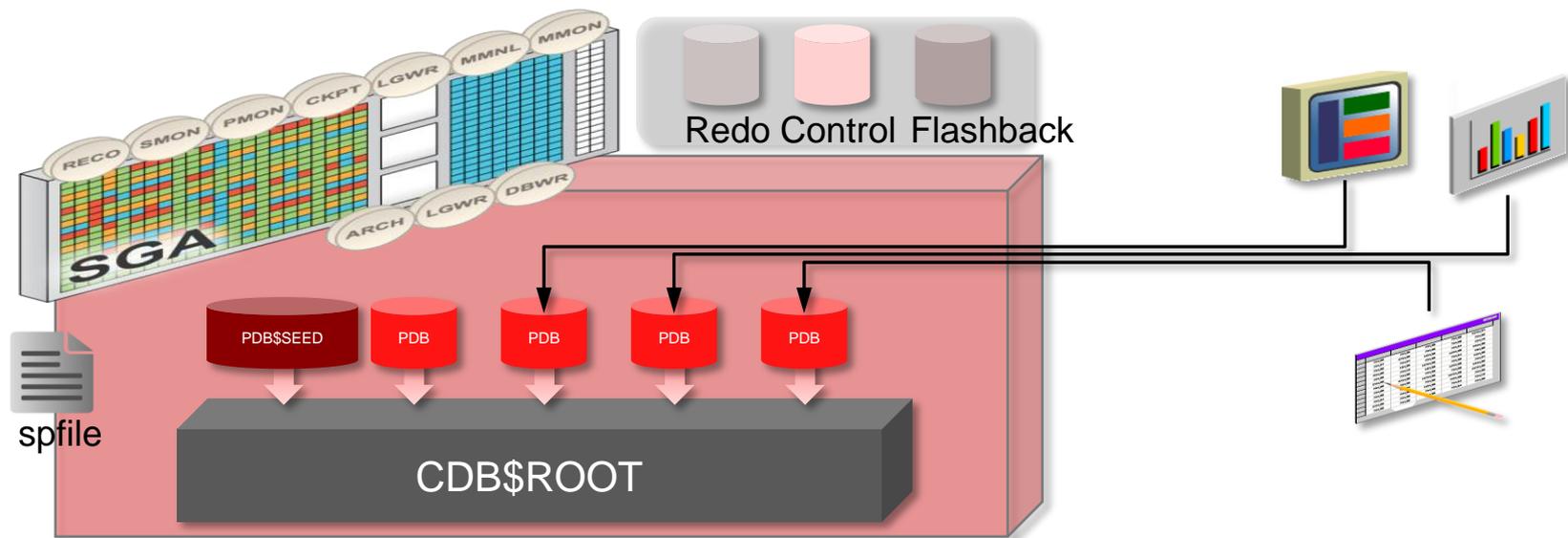
コンポーネント

- CDB って何？
 - PDBs
 - プラガブル・データベース
 - データやコードが含まれる
 - CDB\$ROOT
 - スキーマ、オブジェクトなどすべてのPDB で利用可能
 - データ・ディクショナリに PDB の情報を格納
 - CDB で 1つのみ
 - PDB\$SEED
 - PDB 作成のテンプレート
- PDB、CDB\$ROOT、PDB\$SEED すべてをまとめて、**コンテナ(containers)** と呼ぶ



内部では

- SGA 1つ、バックグラウンド・プロセス一式、サーバー・パラメータ・ファイル1つ
 - リソースを共有することによる恩恵
- アプリケーションは個々のPDB に接続



CDBs / PDBs 使用条件

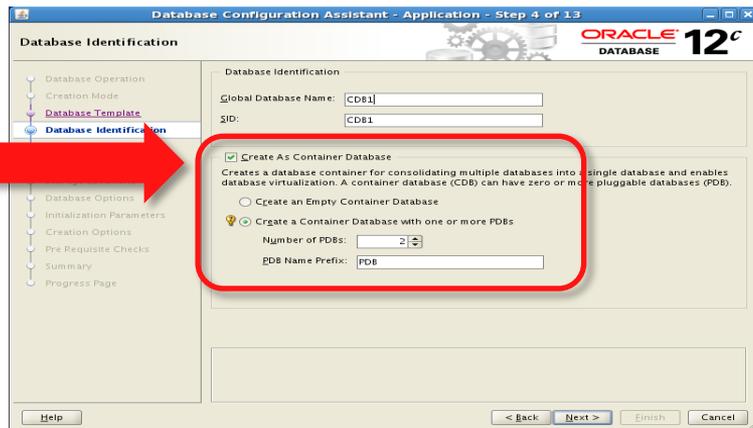
- COMPATIBLE ≥ 12.0.0
 - 注意: 12.0.0 と12.1.0 で違いなし
- ENABLE_PLUGGABLE_DATABASE=TRUE
 - Default: FALSE
 - CDB 作成時にのみ設定可能
 - その後のタイミングで設定しても動作しない
 - DBCA で作成時、コンテナ・データベースの作成を選択すれば、TRUE が設定されます

CDB の作成

- 方法は2つ

- DBCA

- 1 Click !



- コマンドラインでのCREATE DATABASE 文

- すべてのオプションを作成する必要があり、おすすめしません

```
CREATE DATABASE cdb12  
[...]  
ENABLE PLUGGABLE DATABASE  
SEED_FILE_NAME_CONVERT=('/oradata/cdb12/', '/oradata/pdbseed/')  
SYSTEM DATAFILES SIZE 125M AUTOEXTEND ON NEXT 10M MAXSIZE [...]
```

コマンドラインからのCDB 作成

- `catcon.pl` を利用したスクリプトの実行

```
$> perl catcon.pl -u SYS -d $ORACLE_HOME/rdbms/admin -e  
-s -b catalog catdb.sql
```

- `catdb.sql` はすべてのオプションですべてのスクリプトを実行

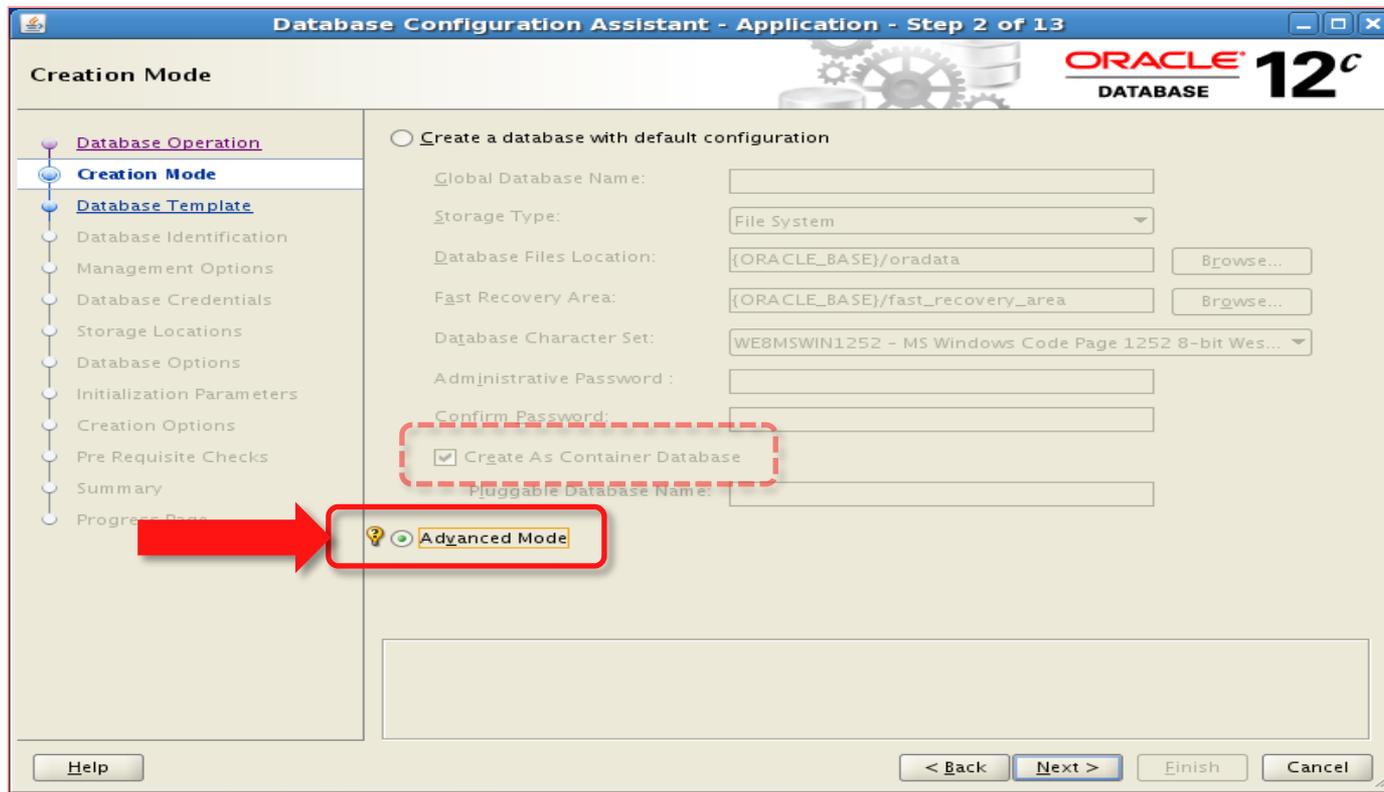
- `catcon.pl` の有用なオプション

- `-u` ユーザー名とパスワード
- `-d` 実行スクリプトを含むディレクトリ(デフォルト: 実行ディレクトリ)
- `-e` *Echo on*
- `-s` すべてのスクリプトの出力をスプール
- `-l` ログ・ファイルが書き込まれるディレクトリ(デフォルト: 実行ディレクトリ)
- `-b` ログ・ファイルのベース名 (必須オプション)

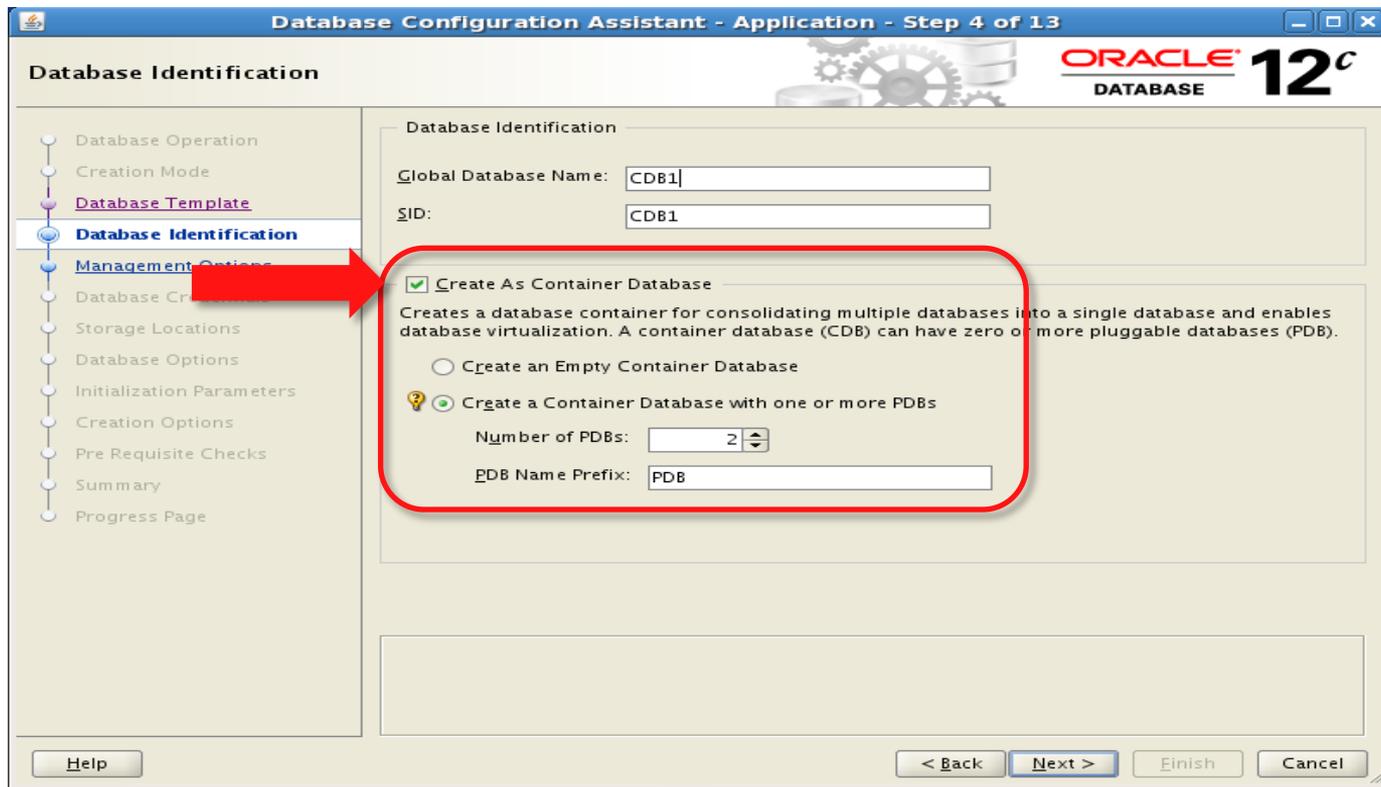
catcon.pl

- catcon.pl 利用時の考慮事項
 - catcon.pl は、PDB をopen しない
 - 推奨
 - catcon.pl を実行する前に、すべてのPDB をオープンする
 - ALTER PLUGGABLE DATABASE ALL OPEN;
 - Perl がPATH がない場合、下記のように実行
 - `$> $ORACLE_HOME/perl/bin/perl catcon.pl ...`

DBCA を利用したCDB の作成 (推奨)

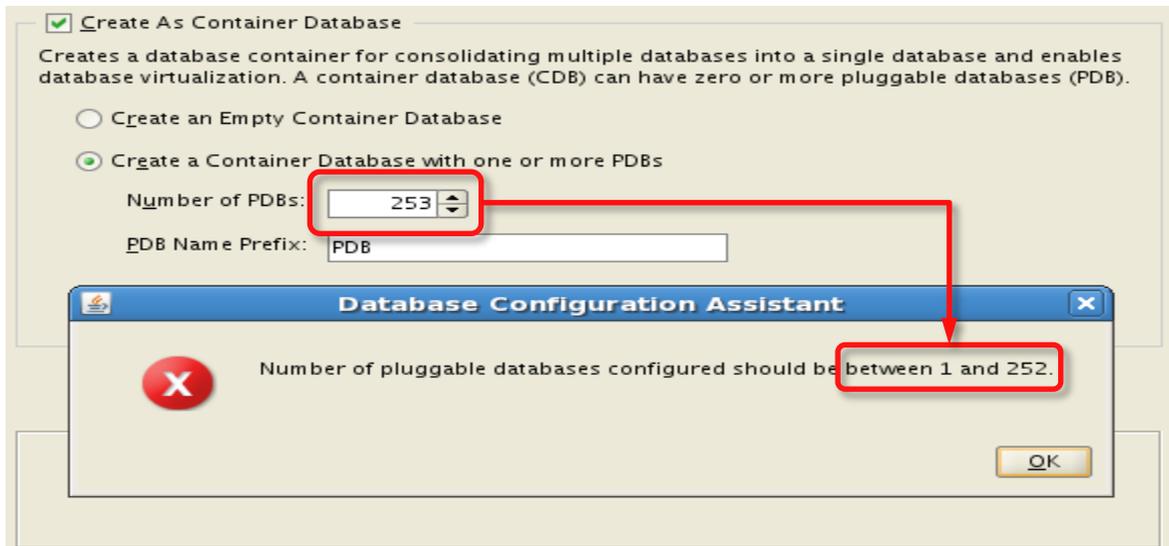


DBCA を利用したCDB の作成 (推奨)



CDB - PDB の制約

- CDB のキャラクタセットは、すべてのPDB で同じ
- 1つのCDB には、252 PDB とPDB\$SEED まで作成可



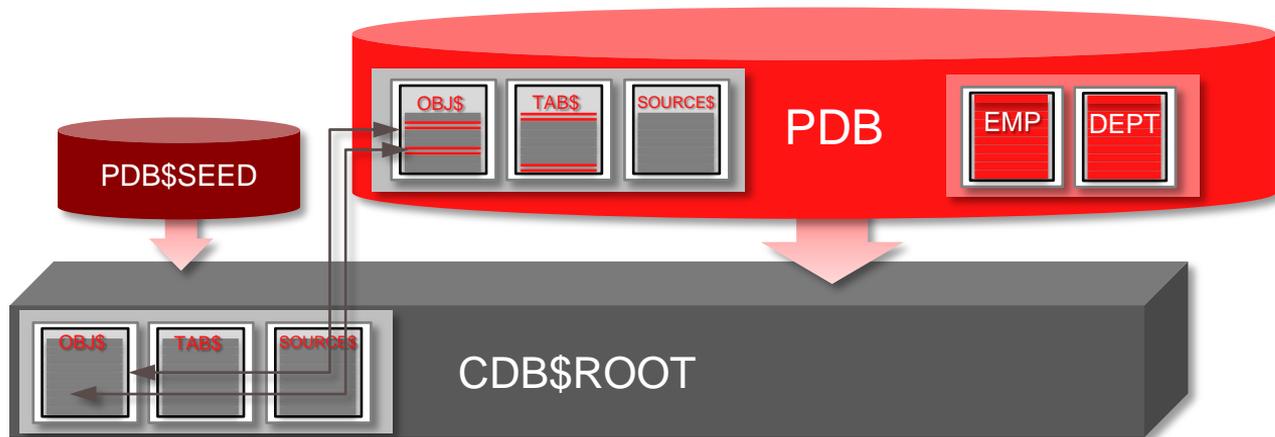
CDB - PDB: Who's who?

- CDB 作成後
 - データ・ディクショナリ
 - 共通ユーザー
 - デフォルトでは、ルートにユーザーデータはない



CDB - PDB: Who's who?

- PDB をCDB にプラグ後
 - データは PDB に存在
 - Oracle 固有のオブジェクトの実体は、CDB に格納
 - PDB からディクショナリ・オブジェクトに対しては、CDB を参照するように変換してアクセス



CDB - PDB: Who's who?

■ 共通ユーザー

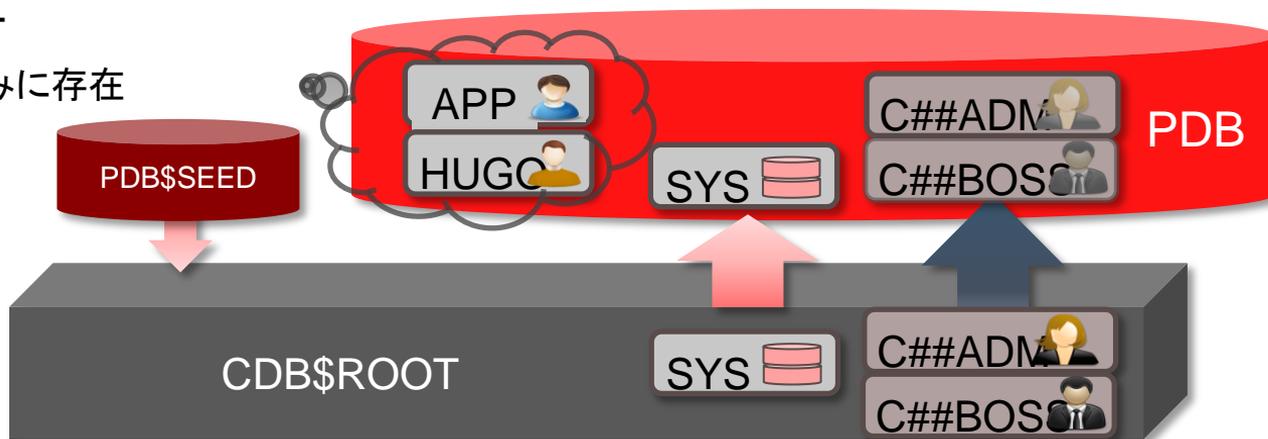
- CDB と現在および将来のPDB に存在
- Oracle が作成するユーザーは、すべて共通ユーザー
- plug/unplug/clone を実行するユーザー

- CDB1> create user **c##adm1** identified by a1;

- CDB1> create user **c##adm2** identified by a2 **container=all**;

■ ローカル・ユーザー

- 1つの PDB のみに存在



テナント間の接続先の切り替え

- PDB への切り替え

- SQL> ALTER SESSION SET CONTAINER=PDB1;

- CDB への切り替え

- SQL> ALTER SESSION SET CONTAINER=CDB\$ROOT;

- 今どこ？

- SQL> SHOW CON_ID

- SQL> SHOW CON_NAME

- SQL> SHOW PDBS

- これらは、接続しているPDB やCDB\$ROOT に接続していれば、PDB\$SEED を含むすべてのPDB を表示します

データ・ディクショナリ・ビュー

データ・ディクショナリ・ビューの種類

- データ・ディクショナリ・ビューの種類には、次の 4 つがある

CDB_xxx

- ルートおよび全 PDB 内のオブジェクトに関する情報

New

DBA_xxx

- ルートあるいは PDB 内のオブジェクトに関する情報

ALL_xxx

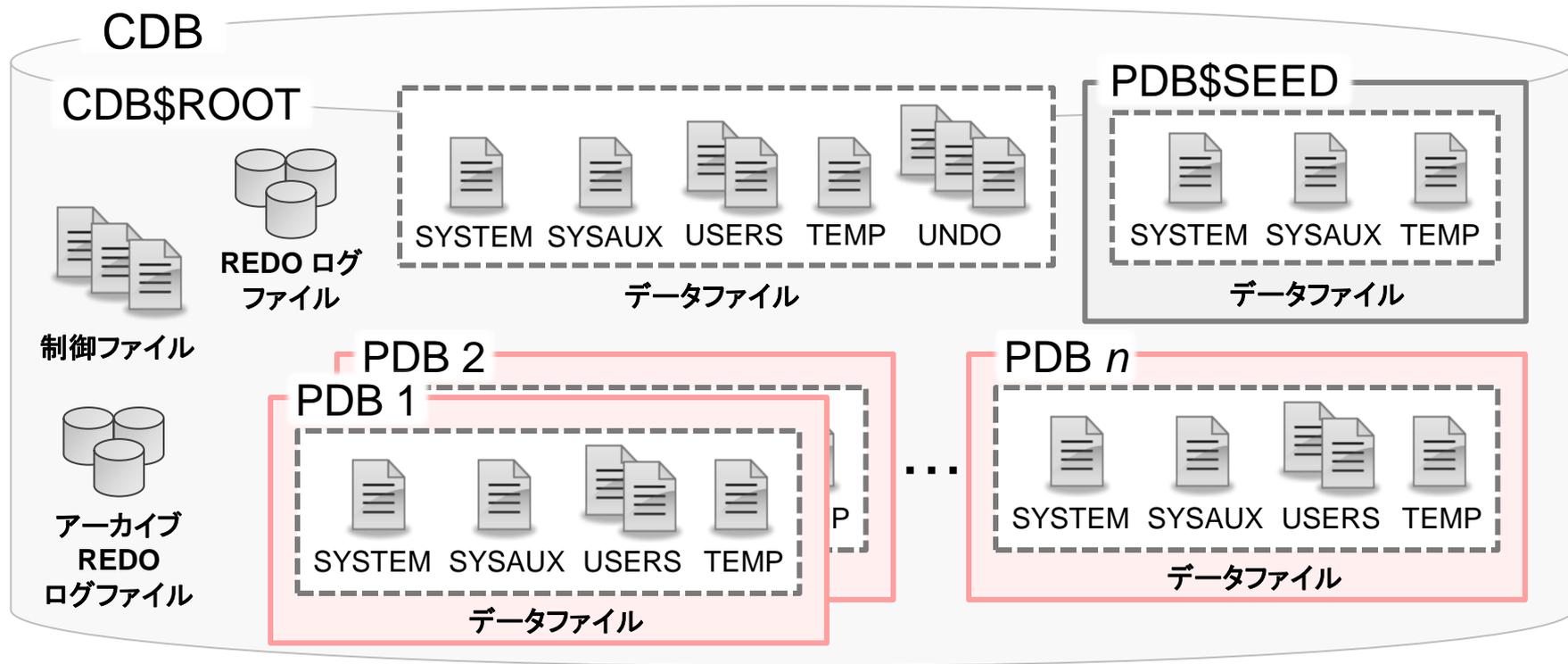
- ユーザーがアクセス可能なオブジェクトに関する情報

USER_xxx

- ユーザーが所有するオブジェクトに関する情報

マルチテナント・コンテナ・データベースの物理構造

データベース関連ファイル



マルチテナント環境の概要

- マルチテナント環境の紹介
- プラガブル・データベースへの道程
 - 新しいPDB の作成
 - PDB のクローニング
 - アップグレードし、PDB としてプラグイン
- ラップアップ



PDB の作成

- PDB\$SEED を利用した高速なプロビジョニング
 - CDB へログイン
 - PDB\$SEED を利用したプロビジョニング

```
create pluggable database PDB128
admin user PDB128adm identified by mypasswd
roles = (DBA)
```

- 初期化パラメータ : PDB_FILE_NAME_CONVERT を新しい場所にファイルを配置するために利用 -> PDB_FILE_NAME_CONVERT='strg1','strg2'
- 必要に応じて、ソース・データベースからデータをインポート
 - 10g 以降のリリースでは、Data Pump インポート(ダンプファイルもしくはNETWORK_LINK の利用)
 - 9i 以前のリリースでは、オリジナルのインポート
 - トランスポートابل表領域

マルチテナント環境の概要

- マルチテナント環境の紹介
- プラガブル・データベースへの道程
 - 新しい PDB の作成
 - PDB のクローニング
 - アップグレードし、PDB としてプラグイン
- ラップアップ



PDB のクローニング

- 同一 CDB 内(ローカル)、または異なる CDB 間 (リモート)で高速なクローニング
- ソースの PDB は、読み取り専用でオープン
- ローカル

```
create pluggable database PDB7 from PDB4;
```

- PDB_FILE_NAME_CONVERT を新しい場所にファイルを配置するために利用
 - » システム単位 (alter system) かセッション単位 (alter session) で指定可能

- リモート

```
create pluggable database PDB8 from  
PDB3@de.acme.cdb1;
```

- PDB8 を作りたい root と PDB3 が DB Link でつながっていることが必要

マルチテナント環境の概要

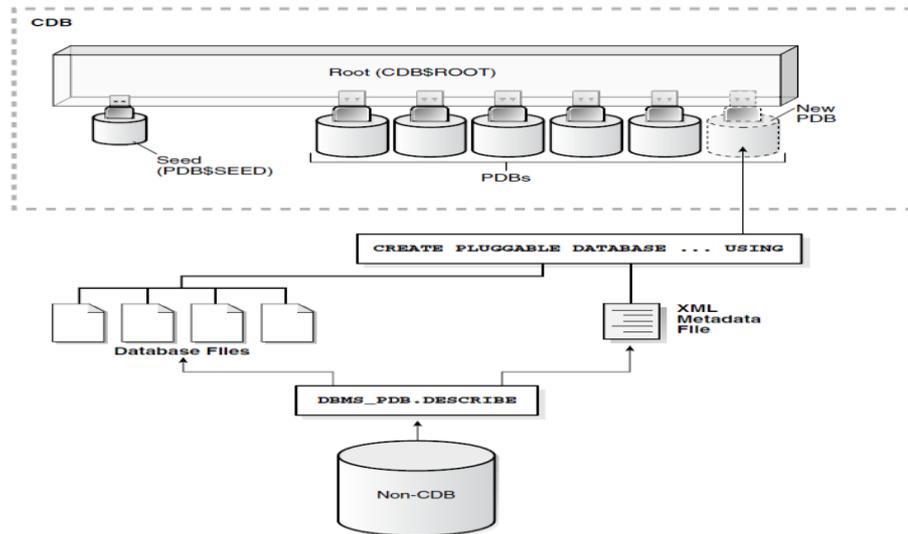
- マルチテナント環境の紹介
- プラガブル・データベースへの道程
 - 新しい PDB の作成
 - PDB のクローニング
 - アップグレードし、PDB としてプラグイン
- ラップアップ



まずアップグレードし、その後、プラグイン

- データベースをマルチテナント環境にプラグインするためには、初めに Oracle Database 12c にアップグレードする必要があります
 - アップグレード・ステップは、マルチテナントの概念とは独立
- アップグレード後

1. 読み取り専用モードでオープン
2. XML ファイル生成
3. データベースを目的の CDB にプラグ

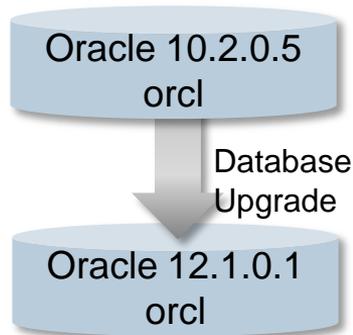


アップグレードとCDB へのプラグイン

■ ステップ1

- データベースをOracle 12c にアップグレード

```
oracle@localhost.localdomain: /u01/app/oracle/product/12.1.0/dbhome_1
File Edit View Terminal Tabs Help
[V112] oracle@localhost:/u01/app/oracle/product/12.1.0/db
$ pwd
/u01/app/oracle/product/12.1.0/dbhome_1/rdbms/admin
[V112] oracle@localhost:/u01/app/oracle/product/12.1.0/db
$ $ORACLE_HOME/perl/bin/perl catctl.pl -n catupgrd.sql
```

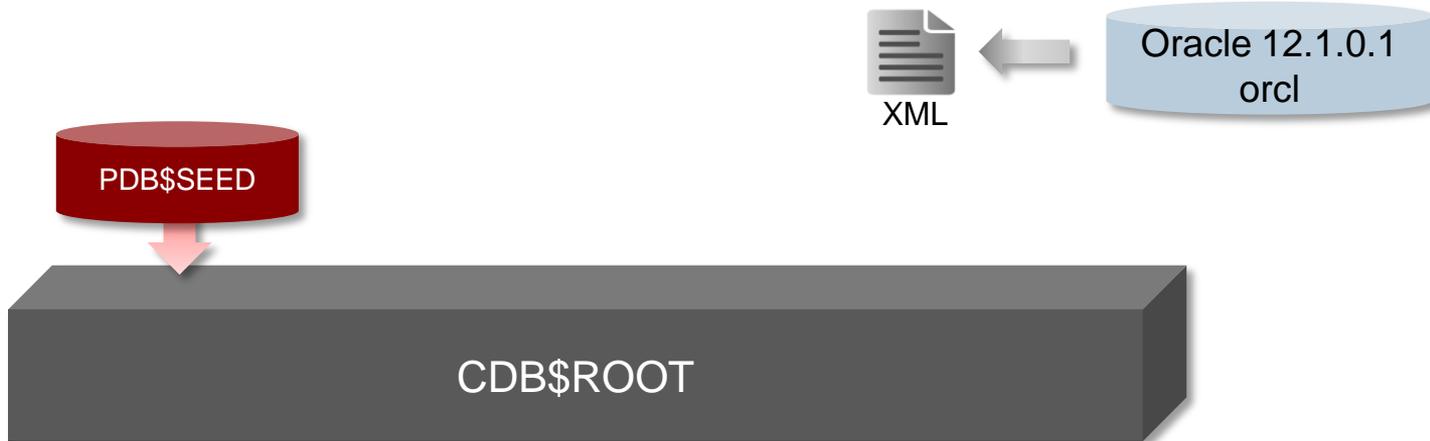


アップグレードとCDB へのプラグイン

■ ステップ2

- データベースを読み取り専用モードでオープン
- XML ファイルを作成

```
2> exec DBMS_PDB.DESCRIBE('/tmp/pdb1.xml');
```

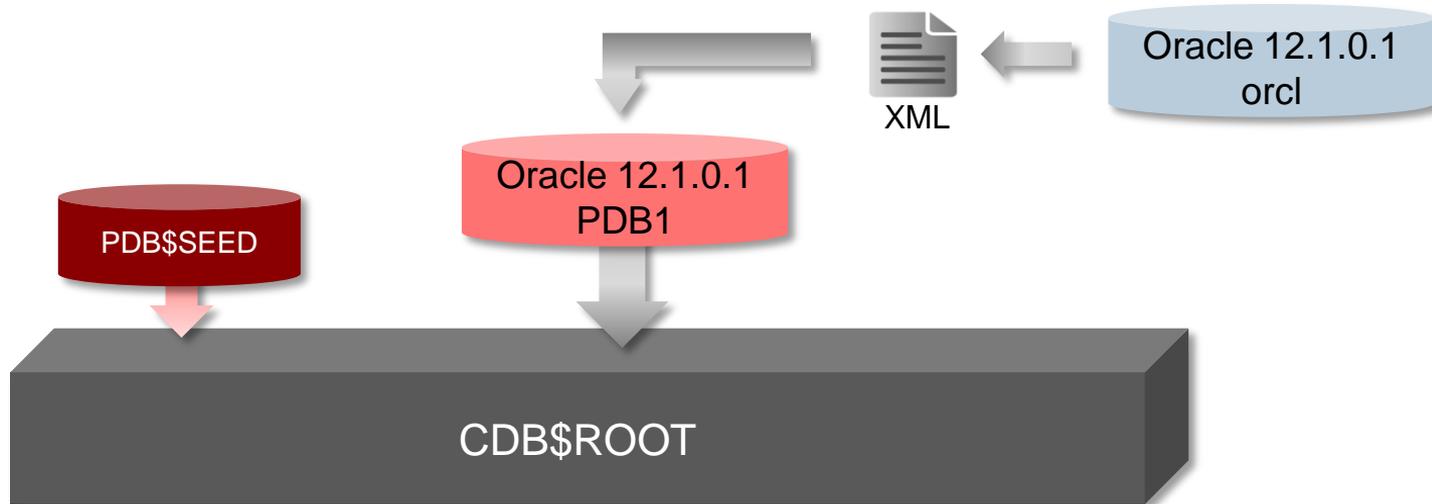


アップグレードとCDB へのプラグイン

■ ステップ3

- データベースを停止し、CDB にプラグ

```
SYS:CDB12> create pluggable database PDB1 using '/tmp/pdb1.xml' nocopy  
Pluggable database created. TEMPFILE REUSE;
```

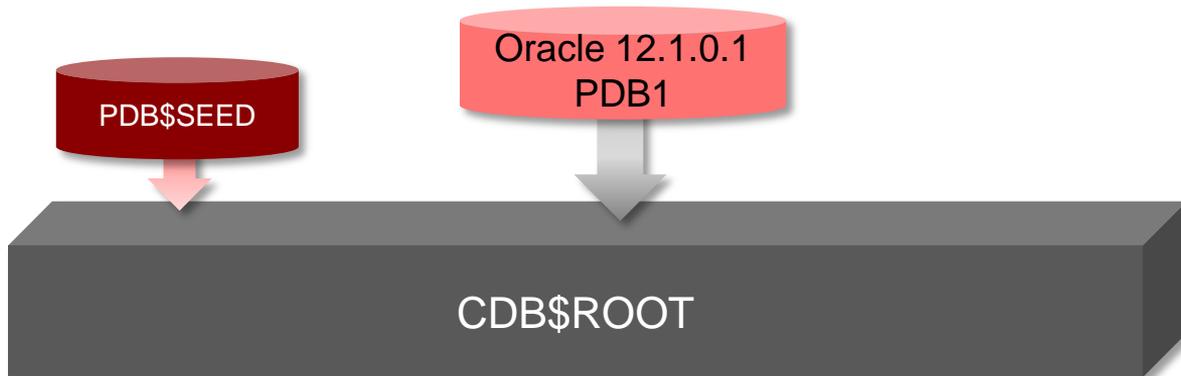


アップグレードとCDB へのプラグイン

■ ステップ4

- 接続記述子を設定し、接続記述子を利用して PDB に接続
 - `sqlplus "sys/oracle@PDB1 as sysdba"`
- ディクショナリへの登録 `*/rdbms/admin/noncdb_to_pdb.sql`
- PDB をオープン

```
SYS:PDB1 > startup  
Pluggable Database opened.
```



アップグレードとCDB へのプラグイン

■ サマリー

1. データベースをOracle 12c へアップグレード
2. データベースをread-only に設定
3. XML ファイルを生成
 - `SQL> exec DBMS_PDB.DESCRIBE('/tmp/pdb1.xml');`
4. データベースを停止
5. XML ファイルに従って、新しいPDB としてCDB にプラグイン
 - `SQL> create pluggable database PDB1 using '/tmp/pdb1.xml' nocopy tempfile reuse;`
6. 新しいPDB に接続して、ディクショナリへ登録
 - `sqlplus "sys/oracle@PDB12 as sysdba"`
 - `SQL> @?/rdbms/admin/noncdb_to_pdb.sql`
7. 新しいPDB を起動
 - `SQL> startup`

Agenda

- はじめに
- マルチテナント・アーキテクチャ概要
- Oracle Database 12c へのアップグレード事例
- Appendix: 12c 新機能

Oracle Database 12c 新機能

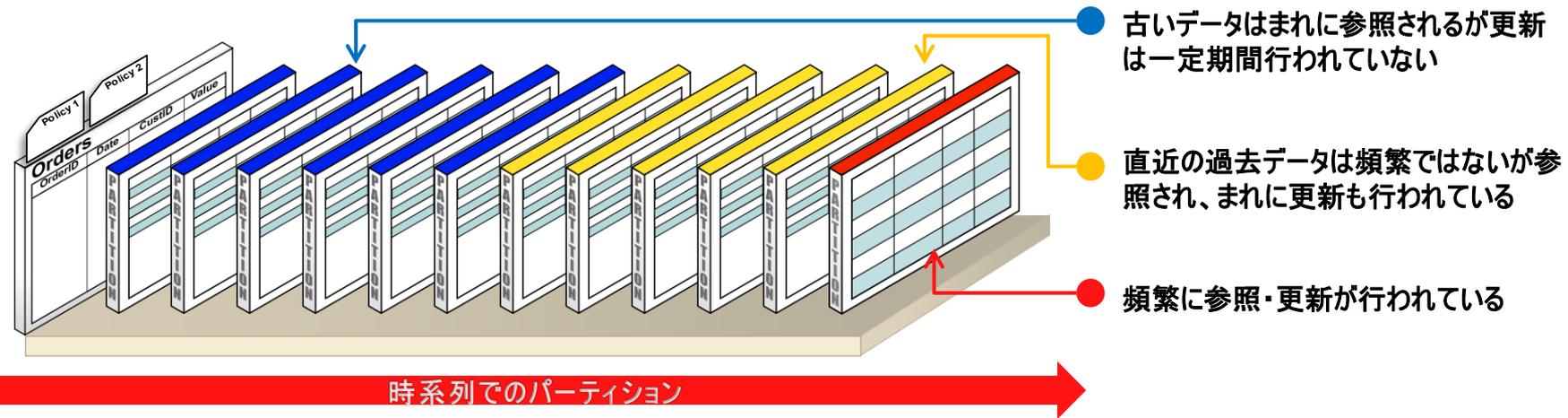
マルチテナント型のデータベース統合、クラウドに必要な可用性、運用の自動化を提供

Cloud & Consolidation	Multitenant Architecture DB内に仮想DBを複数持つ構成	Consolidated Testing 複数環境の合計負荷テスト	
Information Lifecycle Management	Heat Map データ利用頻度を目に見える形でレポートする	Automatic Data Optimization ポリシーベースでデータ運用を自動化する	In-Database Archiving DB内でデータをアーカイブと同等に見せる
Security	Data Redaction DBユーザ毎に動的マスキング	Privilege Analysis 特権ユーザの利用状況を分析レポートとして発行する	
Cluster	Flex ASM ASMインスタンスとDBインスタンスを分離して配置	Flex Cluster オラクルの新しいクラスタウェア、Hub&Spoke型のクラスタを構成	
High Availability	Far Sync (Data Guard) 最小負荷でデータロスの無い災対サイトを構築	Global Data Services 複製サイトをまとめてサービス化し、障害時の切替に負荷分散を適用	Application Continuity DBがダウンした際、成功するまでトランザクションをリプレイする

Heat Map

データの参照・更新の状況を保持し、情報ライフサイクル管理の基準となるデータを提供

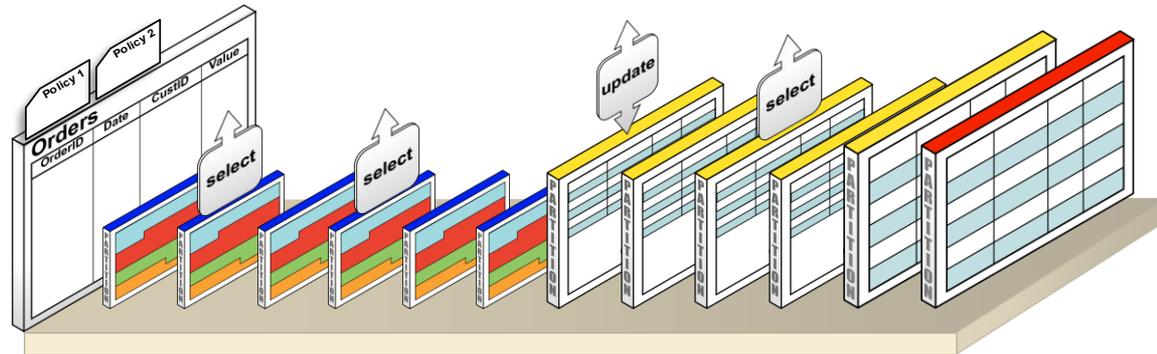
- データベース、パーティション、列、または、ブロック/エクステントのレベルで利用状況のレポートを発行します
- 本機能の利用には、Advanced Compression のライセンスが必要です



Automatic Data Optimization (ADO)

設定したポリシーに従って自動でデータ配置を最適化

- 表/表領域単位でポリシーを設定し、データの圧縮やレベル変更を自動で行うことができます
- ポリシーの条件は Heat Map が取得するデータ操作からの経過時間によりパーティション・表領域のレベルで設定されます
- 本機能の利用には Advanced Compression のライセンスが必要です



時系列でのパーティション

Policy1:

作成から90日後経ったら圧縮

```
ALTER TABLE sales
  ILM ADD CompressionPolicy
  COMPRESS Partitions for Query
  AFTER 90 days from creation;
```

Policy2:

180日以上変更がなければアーカイブ

```
ALTER TABLE sales
  ILM ADD MovePolicy
  TIER Partitions TO 'Archive_TBS'
  AFTER 180 days of no modification;
```

Data Redaction

アクセスするユーザーやアプリケーションに応じてデータをリダクション(編纂)して返す

- ユーザーに対しポリシーを設定することで、結果の返し方を設定します
- 結果の返し方は、伏字・置き換え・ランダムなどから選択して、Enterprise Manager から指定することができます
- 本機能の利用には Advanced Security のライセンスが必要です



CLIENT CONTACT SARA JONES	SOC. SEC. NO. ***_*_*_3428	DOB 01/01/01	PIN ****
CLIENT CONTACT JAMES SMITH	SOC. SEC. NO. ***_*_*_2256	DOB 01/01/01	PIN ****

● コールセンタ

業務に関係ない個人情報は参照させないため、社会保障番号やIDなどの情報は伏字にしてアプリケーションに表示

EMPLOYEE NAME SARA JONES	SOC. SEC.# 115-69-3428	DOB 11/06/71	PIN 5623
EMPLOYEE NAME JAMES SMITH	SOC. SEC.# 114_50_2256	DOB 01/12/68	PIN 2702

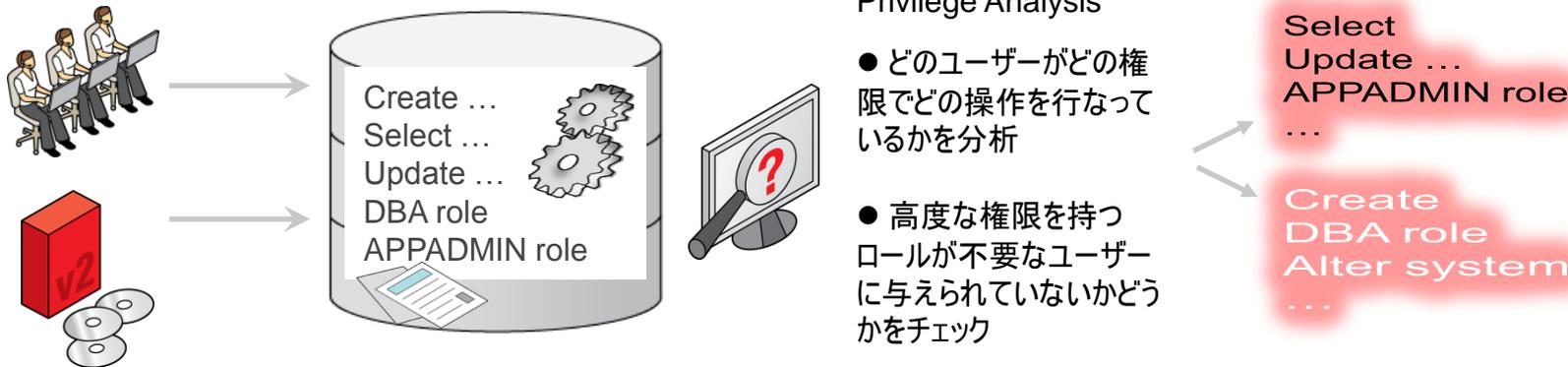
● 給与担当者

業務上必要となるので、全ての情報をアプリケーションに表示

Privilege Analysis

データベースでの最小権限許諾を遂行するためのチェック機能

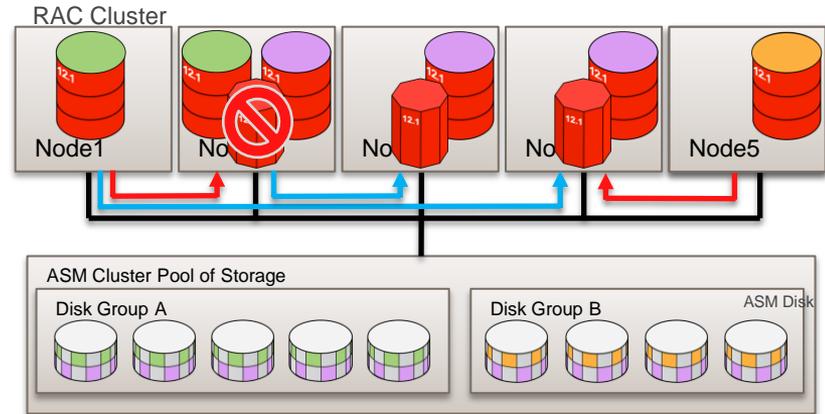
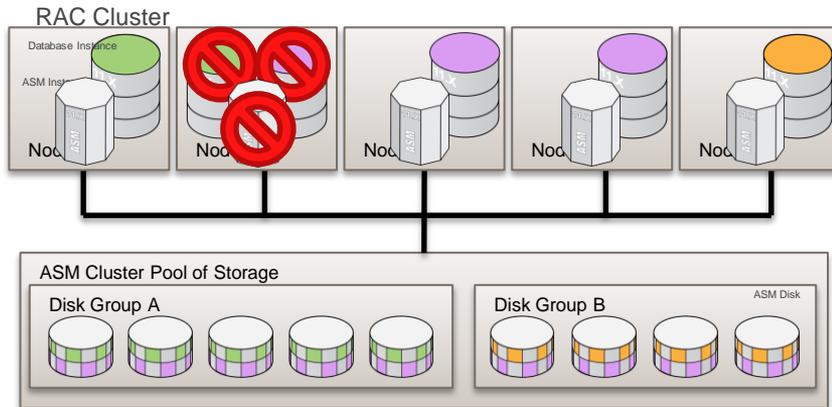
- 権限とロールの使用状況をキャプチャし結果を専用のディクショナリビューからレポート出力します
- 不必要な権限を実績にもとづいて剥奪していくことで、最小権限でのデータベース運営に近づけることができ、情報漏えいのリスクも最小化することができます
- 本機能の利用には Database Vault のライセンスが必要です



Flex ASM

ストレージ・グリッドとデータベース・グリッドを分離することで可用性を向上

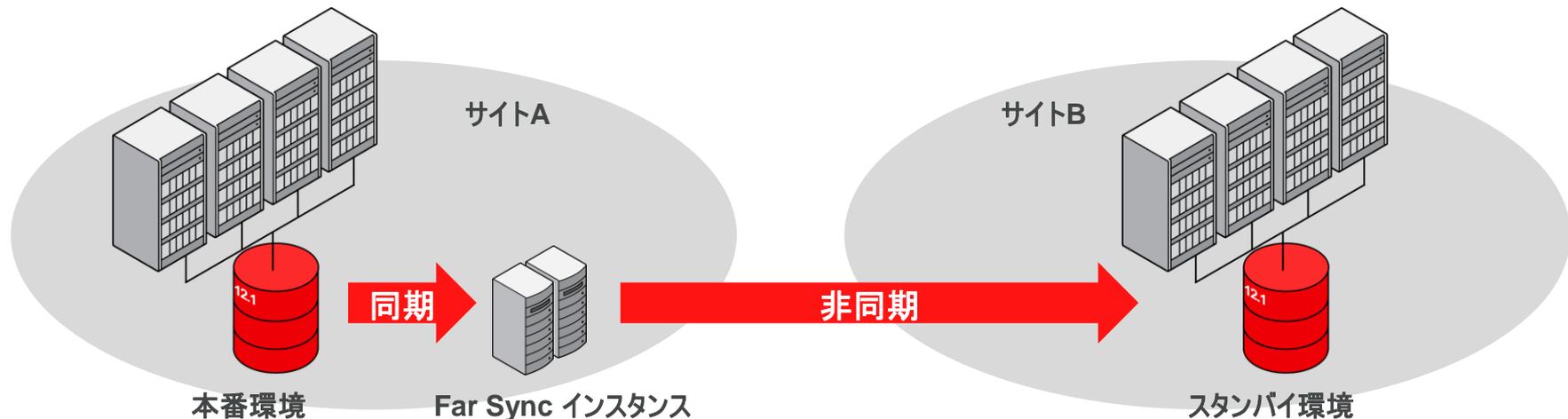
- データベースインスタンスとASMインスタンスの依存関係を取り除き、それぞれ独立したクラスタとして構成
- ASMインスタンスを持たないRACノードは他ノードのASMクライアントとして動作し、接続先のASMインスタンスがダウンした場合には、他のASMインスタンスに接続を切り替えます
- ASMインスタンスの障害にデータベースインスタンスが影響を受けなくなるため、データベースサーバは生きたままの運用が可能であり、更にASMインスタンスの可用性も確保されるようになります



Far Sync

災害対策サイトなどの遠隔スタンバイでゼロ・データロスを実現

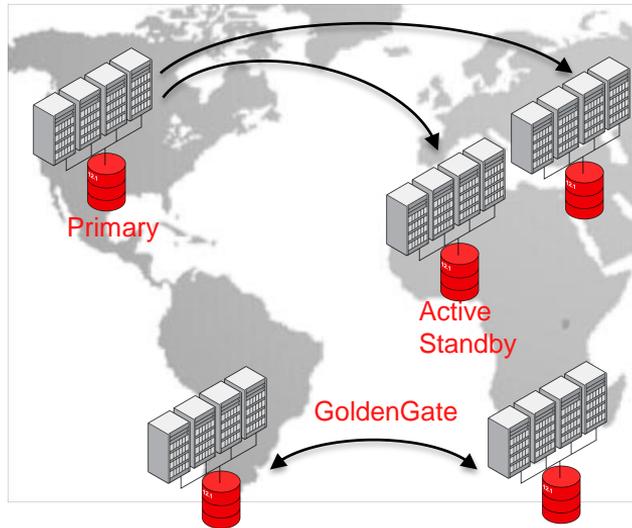
- 従来の同期転送でのネットワーク遅延やコミット処理への影響を最小化します
- Far Sync インスタンスと呼ばれるログ転送専用インスタンスから転送処理を行うため、同期転送時のネットワーク・レイテンシを最小限に押さえ、本番環境へのパフォーマンス影響を最小化した上でゼロデータロスの環境を構築します
- 本機能の利用には Active Data Guard のライセンスが必要です



Global Data Service (GDS)

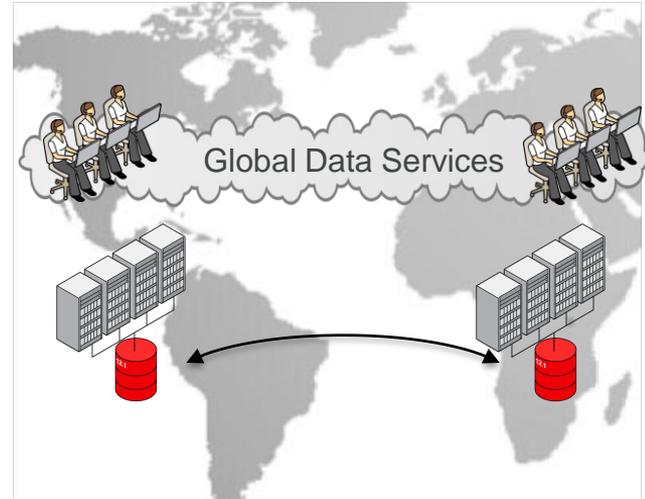
レプリケーション・サイト間でのフェイルオーバーとロードバランシングを実装

- データセンターをまたがる複数のサイト間をサービスとしてまとめ、障害発生時の切り替えをロードバランスを考慮した上で自動的に行い、手動による切り替えのラグや管理の負荷を下げることができます
- 複数のデータセンターにスタンバイ環境がまたがっているなどで、複数サイトでの障害などが想定される場合に有効です
- 本機能の利用には Active Data Guard が必要です



← 従来のバックアップサイト切替
組合せごとに複製や切替の
方法が異なり、また切替時の負荷
分散は手動管理となる

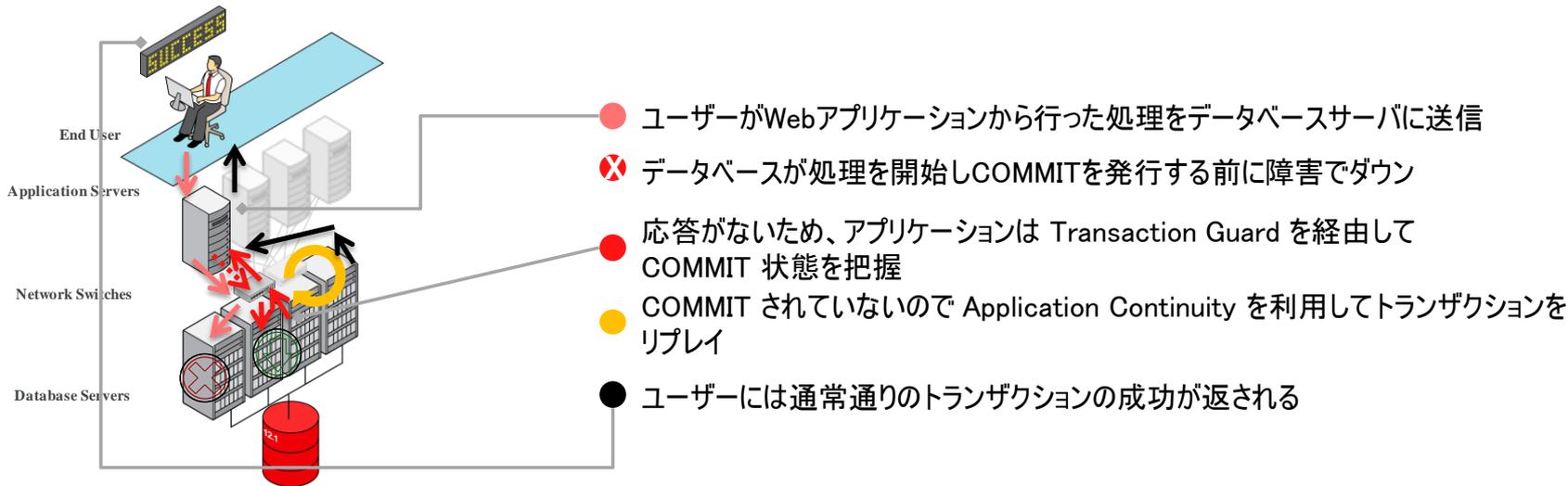
Global Data Service →
本番環境とスタンバイ環境を複
数まとめてサービスとして登録し、
サービス内で負荷分散を配慮し
た切替を行う



Application Continuity (AC)

障害時にトランザクションをリプレイ

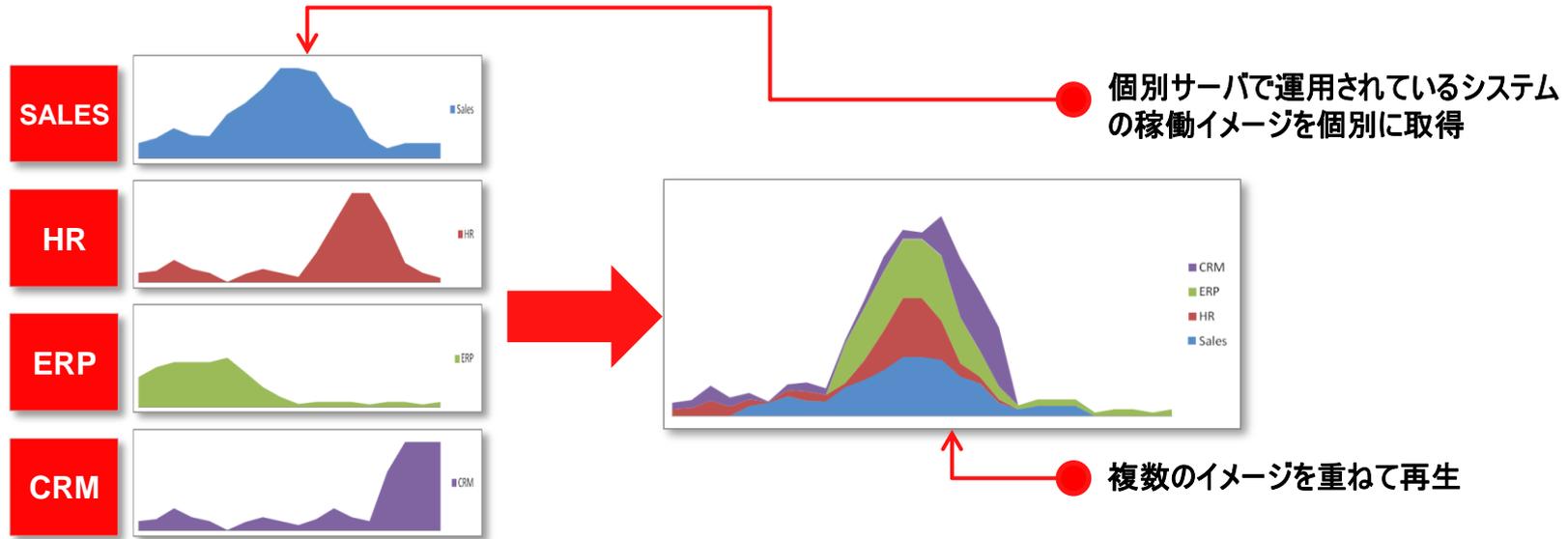
- 最後のトランザクションの結果を返すAPI であるTransaction Guard と連携し必要に応じてトランザクションを再生します
- アプリケーションのエラー処理をコミット状況に応じて分けることができるため、エラー処理のパターンを明確に定義付けることができ、結果として障害時の二重処理や未処理が減るためにユーザーエクスペリエンスを向上させることができます



Consolidated Testing

システム統合後の稼働イメージを再生

- 個別に稼働しているシステムの実際の稼働状態をそれぞれ記録し、統合後の状態として合わせて再生することで負荷状況のシミュレートを行う機能です
- 本機能の利用には、Real Application Testing ライセンスが必要です



Hardware and Software

ORACLE®

Engineered to Work Together

ORACLE®