

ORACLE®

マルチテナント・アーキテクチャ エッセンシャルズ

日本オラクル株式会社
オラクル・ユニバーシティ シニアインストラクタ
煤田弘法



 #odddtky

日本オラクル、今年最大の技術トレーニング・イベント

**Oracle DBA &
Developer Day 2013**

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。

Program Agenda

- マルチテナント・アーキテクチャの概要
- マルチテナント・アーキテクチャの管理
- CDB、PDBにおける表領域とユーザーの管理
- CDB、PDBのバックアップ、リカバリおよびフラッシュバック
- CDB、PDBでのリソース・マネージャの使用

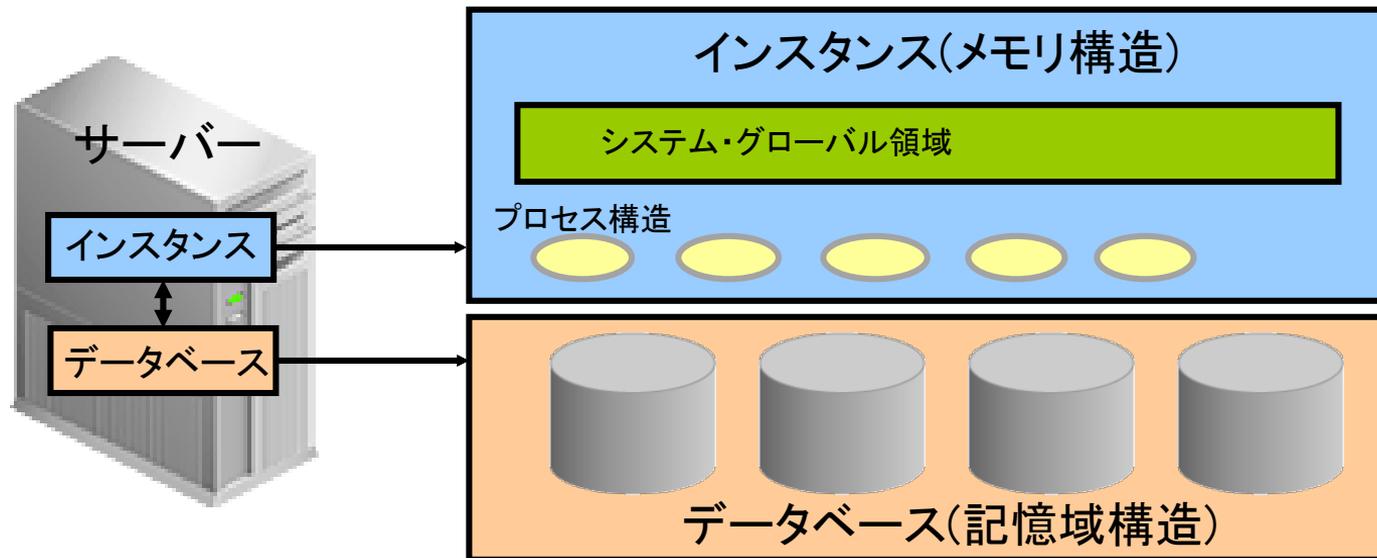
マルチテナント・アーキテクチャ の概要

課題

Oracle RDBMS上に構築された多数の部門アプリの課題

- それらデプロイされているハードウェアの大部分が使用されていない。
- インスタンスおよび記憶域のオーバーヘッドが原因で、多数の部門データベースを物理的に同じ記憶域サーバーに配置出来ない。
- 管理者が常駐して100%の注意力を注ぐ必要があるほど複雑ではない。
- すべてのパッチ適用やアップグレードに多大な時間を必要となる。 など

Oracle Database 11gリリース2

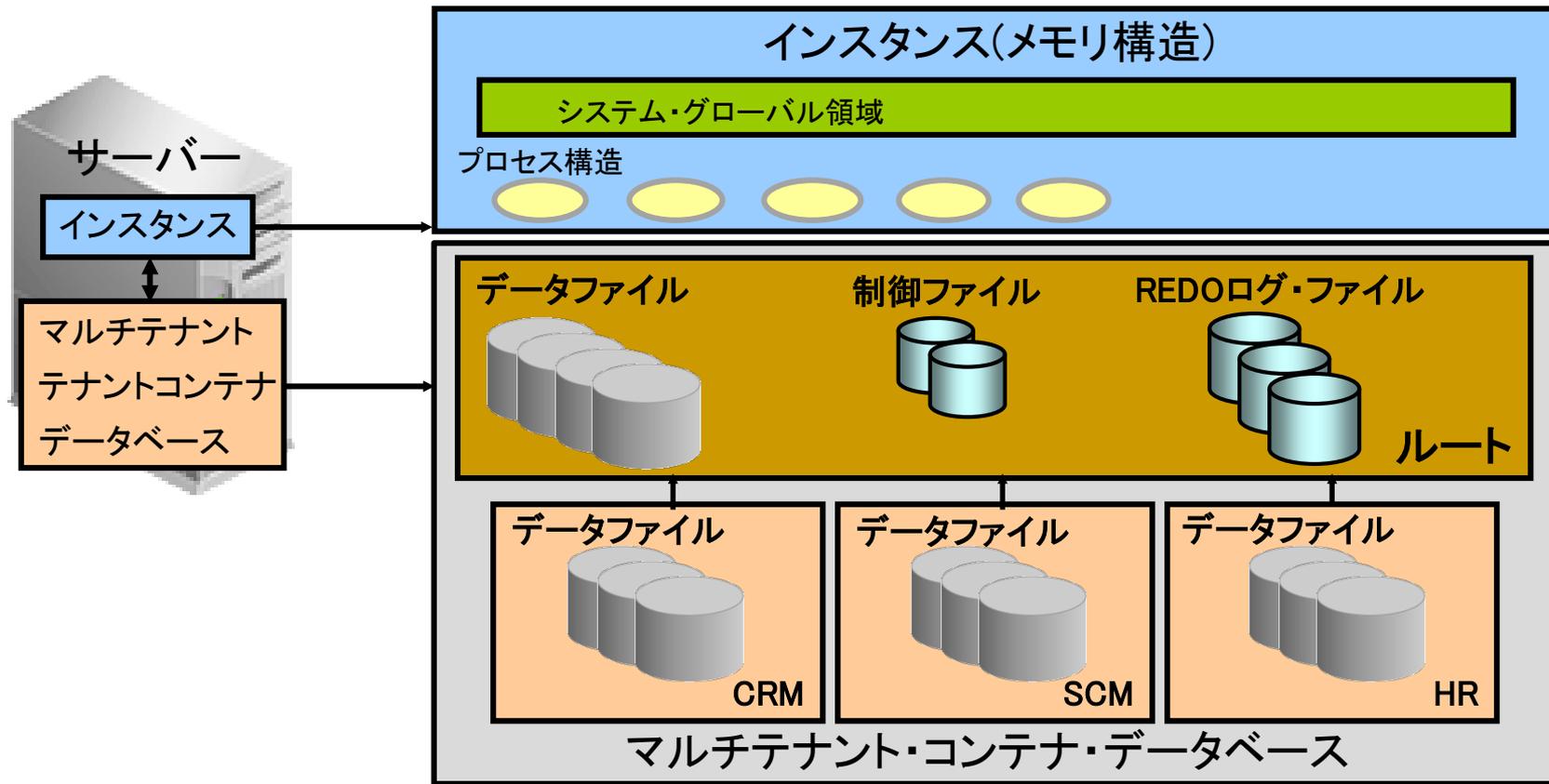


- 非CDBが複数存在しても何も共有されない。
 - バックグラウンド・プロセス、共有プロセス・メモリーが多い
 - Oracleメタデータのコピーが多い

マルチテナント・アーキテクチャのメリット

- 集中管理されたプラットフォームで複数データベースを稼働させコスト削減
 - インスタンスのオーバーヘッド、記憶域のコストが比較的小さい
- DBAリソース・コストを削減し、セキュリティを確保
 - アプリケーションの変更なし
 - 高速で簡単なプロビジョニング
 - パッチ適用およびアップグレードの時間短縮
- 非CDBとの完全な下位互換性を保証
- RACで完全に動作
- 集中管理および複数データベース管理を可能
 - バックアップ/リカバリ

マルチテナント・コンテナ・データベース



初期インストール

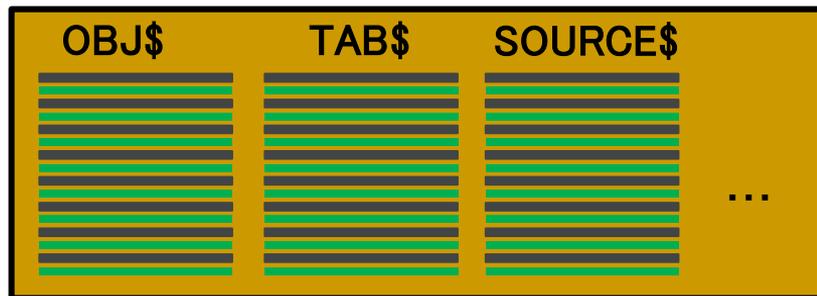
データベース作成後のオブジェクトは、Oracle提供オブジェクトのみ

| OBJ\$ | TAB\$ | SOURCE\$ | |
|-------|-------|----------|-----|
| ===== | ===== | ===== | |
| ===== | ===== | ===== | |
| ===== | ===== | ===== | |
| ===== | ===== | ===== | |
| ===== | ===== | ===== | ... |
| ===== | ===== | ===== | |
| ===== | ===== | ===== | |

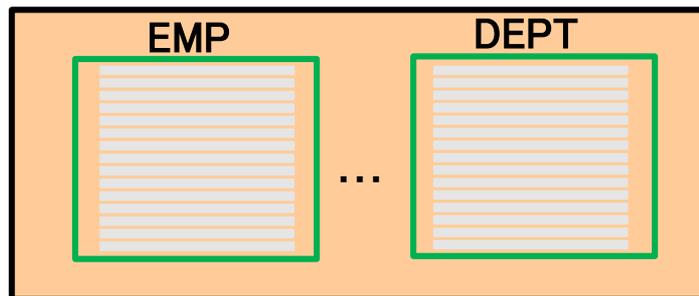
Oracleシステム・データ

ユーザー・データの追加

メタデータがデータ・ディクショナリ内のOracle提供データと混在する

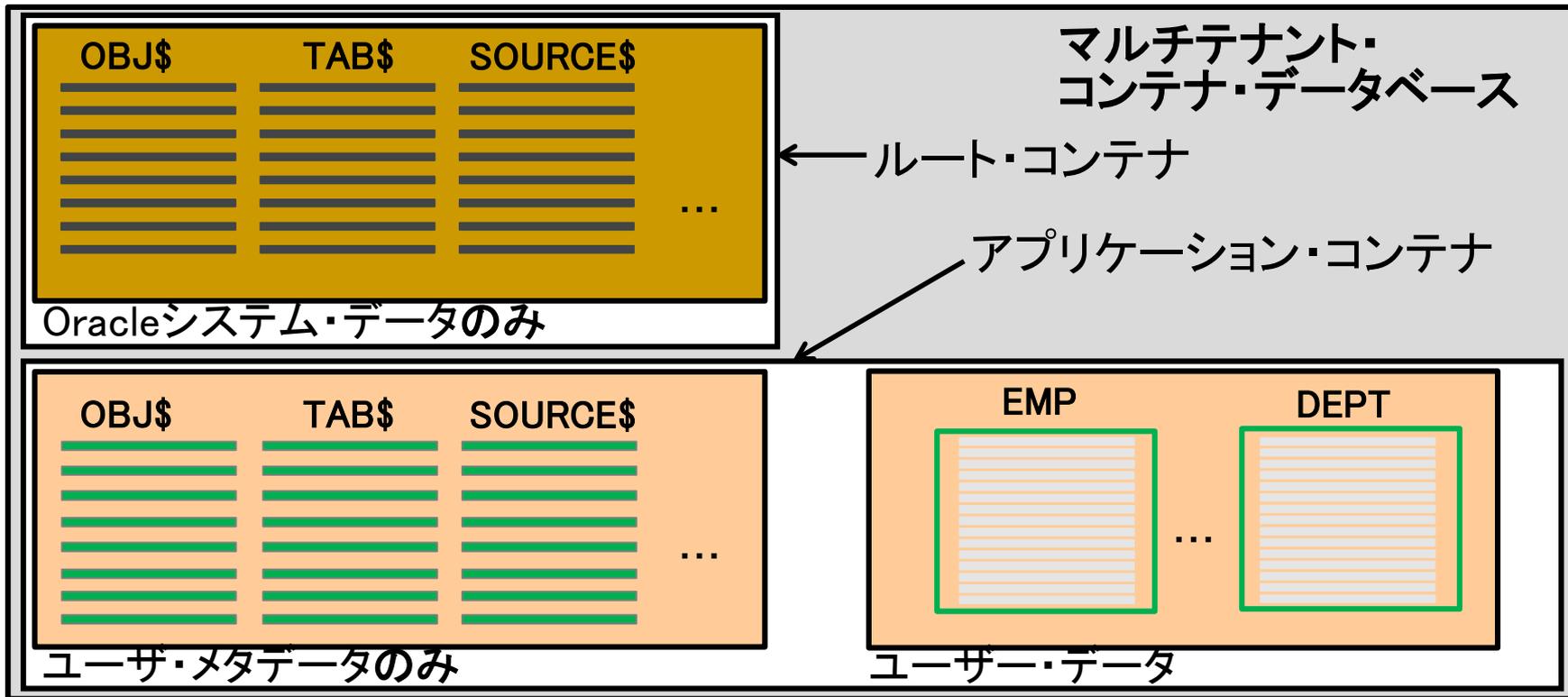


Oracleシステム・データとユーザ・メタデータの混在

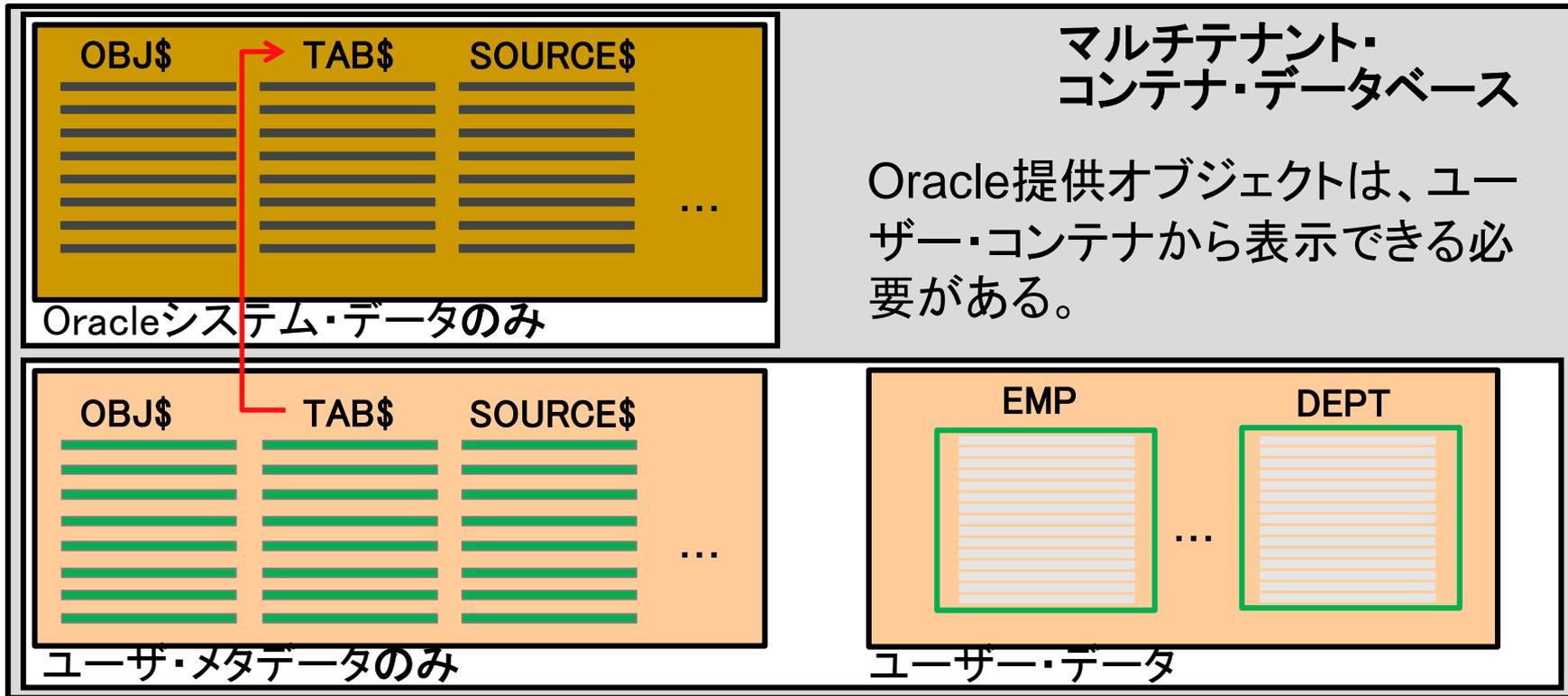


ユーザー・データ

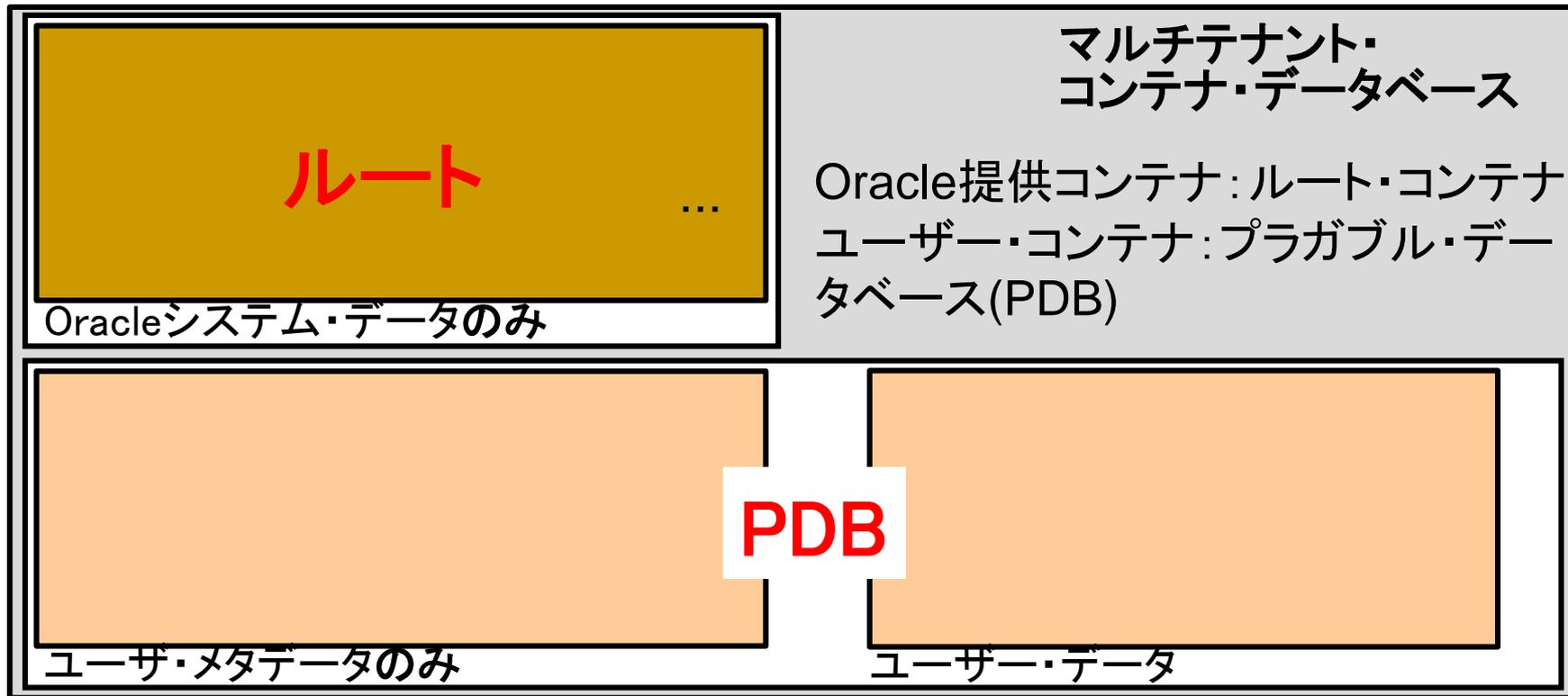
システム・データとユーザデータの分離



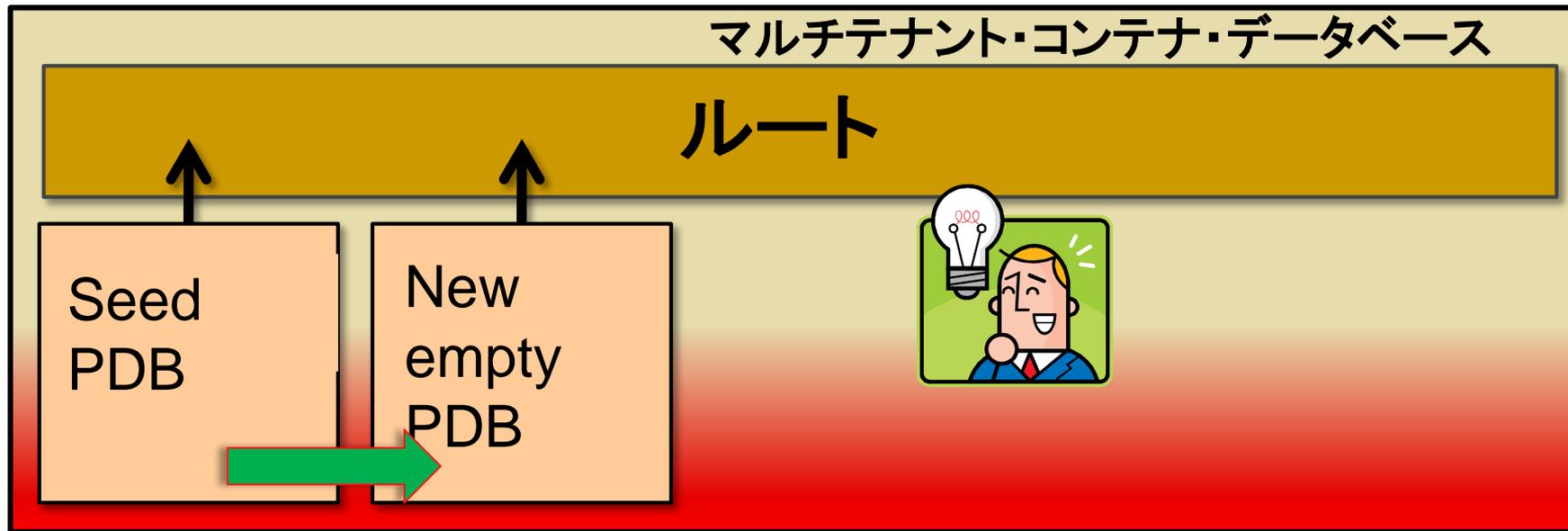
システム・データとユーザデータの分離



コンテナの命名



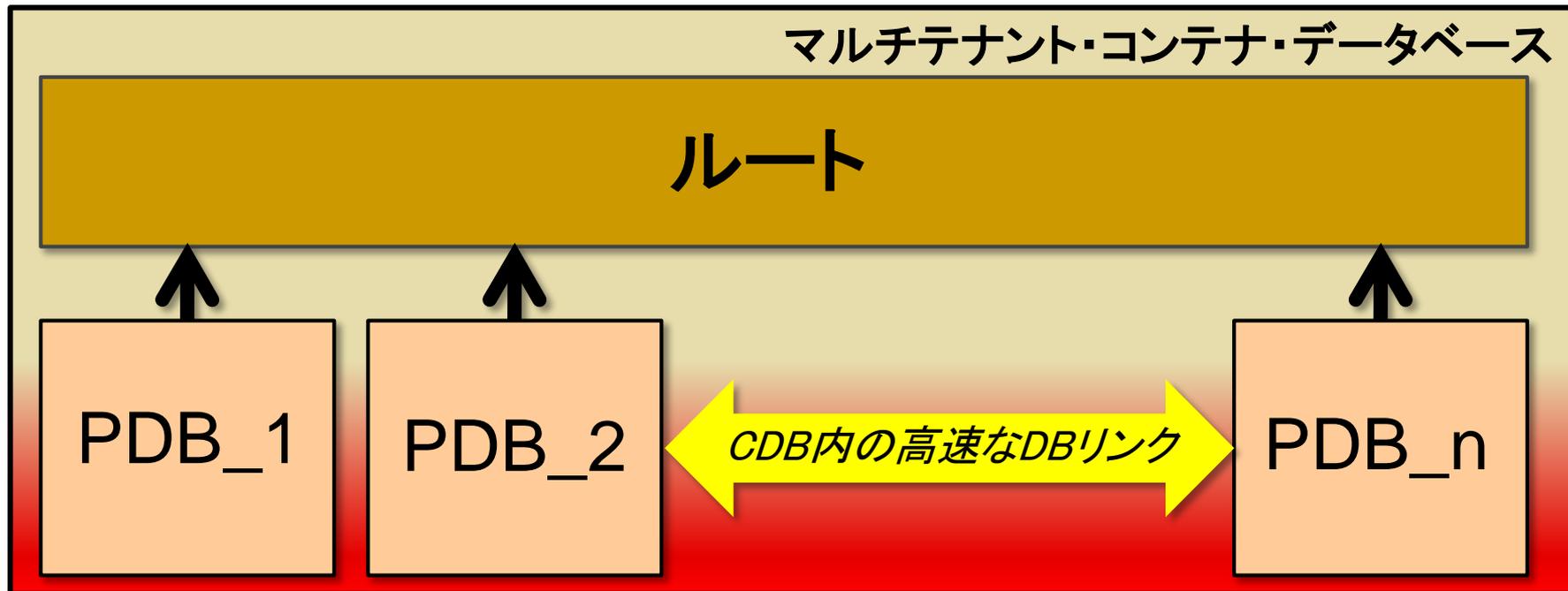
プラガブル・データベースのプロビジョニング



- PDB\$SEEDからの新しいPDBを作成
- 別のPDBからのPDBのクローニング
- 非CDBを接続
- 切断したPDBを別のCDBに接続

マルチテナント・コンテナ・データベース内の相互作用

データベース・リンクを使用してデータを共有



コンテナ

- ルート・コンテナ
 - CDB作成時に最初に作成される必須コンテナ
 - Oracleシステム提供の共通オブジェクトおよびメタデータ
 - Oracleシステム提供の共通ユーザーおよびロール
- プラガブル・データベース・コンテナ(PDB)
 - シードPDB(必須コンテナ、PDB\$SEED)
 - アプリケーション用コンテナ
- CDB内のPDBの制限は、シードを含めて253個
- CDB内のサービスの制限は512個

データ・ディクショナリ

CDB_xxx すべてのPDBのマルチテナント・コンテナ・データベース内のすべてのオブジェクト

DBA_xxx コンテナまたはプラグブル・データベース内のすべてのオブジェクト

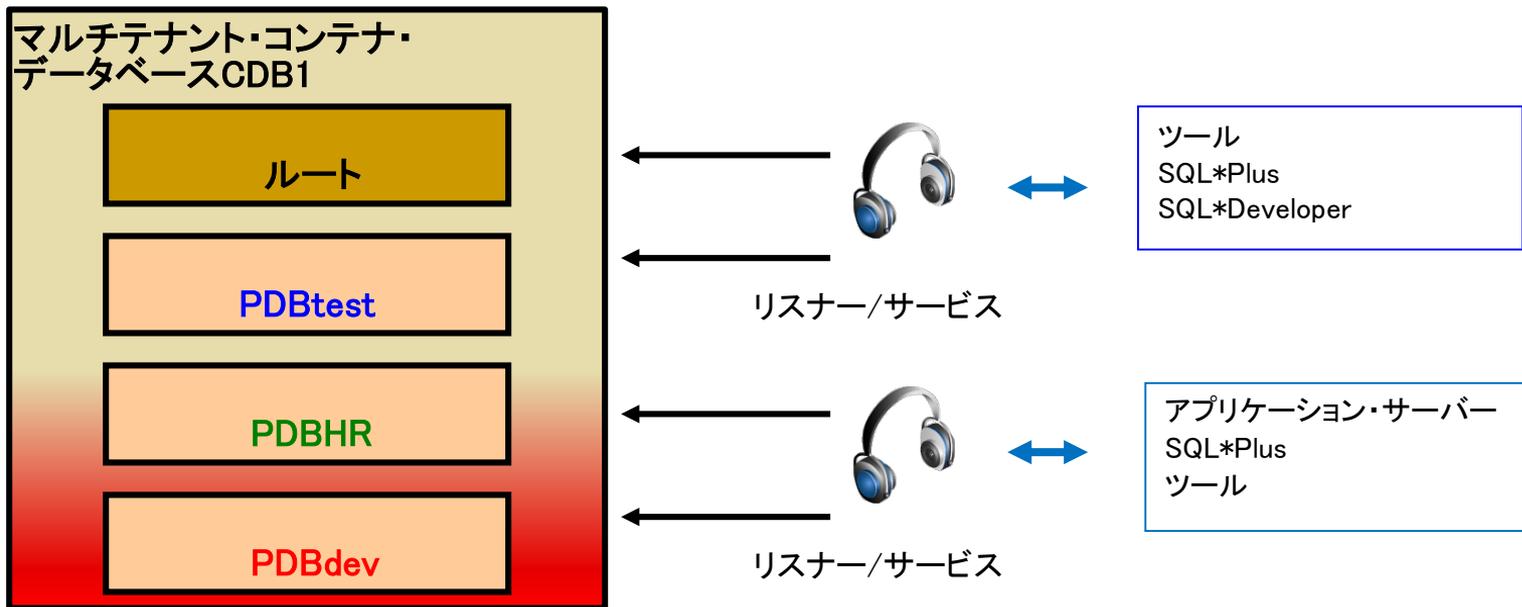
ALL_xxx 現行ユーザーがアクセスできるオブジェクト

USER_xxx 現行ユーザーによって所有されているオブジェクト

- CDB_pdfs: CDB内のすべてのPDB
- CDB_tablespaces: CDB内のすべての表領域
- CDB_users: CDB内のすべてのユーザー(共通およびローカル) など

マルチテナント・アーキテクチャ の管理

接続



- ルート: OS認証、サービス名
- PDB: サービス名(OS認証は使用不可)

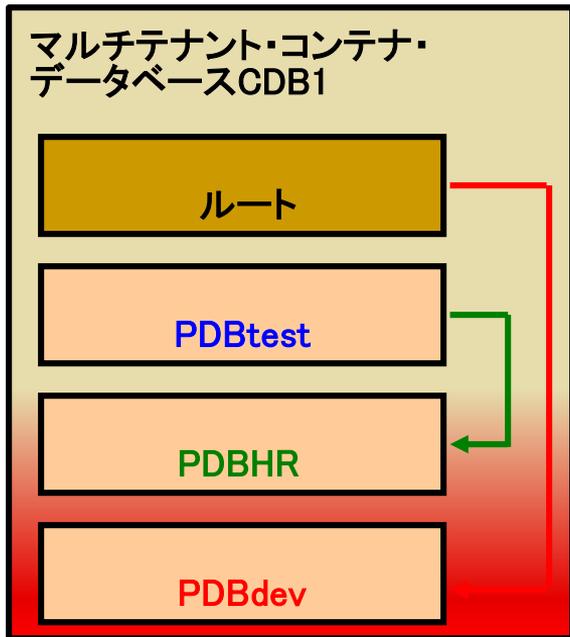
サービス

- それぞれのPDBにはデフォルト・サービスが存在する。
- サービス名はCDB間で一意である必要がある。

```
SQL> SELECT name, pdb FROM cdb_services;
```

```
SQL> CONNECT / AS SYSDBA  
SQL> CONNECT sys@PDBtest AS SYSDBA  
SQL> CONNECT local_user1 @hostname1:1525/PDBHR  
SQL> CONNECT common_user2@PDBdev  
SQL> SHOW CON_NAME
```

接続の切換え



- CDB内テナ間での接続切換え
 - 再接続

```
SQL> CONNECT / AS SYSDBA  
SQL> CONNECT local_user1 @PDBdev
```

- ALTER SESSION文の使用

```
SQL> CONNECT sys@PDBtest AS SYSDBA  
SQL> ALTER SESSION SET CONTAINER=PDBHR;  
SQL> SHOW CON_NAME  
SQL> ALTER SESSION SET  
CONTAINER=CDB$ROOT;
```

CDBのオープン

```
SQL> CONNECT sys@CDB1 AS  
      SYSDBA  
SQL> STARTUP
```

OPEN

- ルートがオープンされます。
- PDBは**読取り専用のシードを除き、マウントされたままです。**

MOUNT

- インスタンスのCDB制御ファイルがオープンされます。
- ルートがマウントされます。
- PDBがマウントされます。

NOMOUNT

インスタンスが
起動します。

SHUTDOWN

PDBのオープン

```
SQL> ALTER PLUGGABLE  
2> DATABASE ALL OPEN;
```

```
SQL> CONNECT sys@CDB1 AS SYSDBA  
SQL> ALTER PLUGGABLE  
2> DATABASE PDB1 OPEN;
```

PDB OPEN

PDBは読取り/書込み
でオープンされる。
(シードは除く)

OPEN

- ルートがオープンされます。
- PDBは読取り専用のシードを除き、マウントされたままです。

MOUNT

- インスタンスのCDB制御ファイルがオープンされます。
- ルートがマウントされます。
- PDBがマウントされます。

NOMOUNT

インスタンスが
起動します。

SHUTDOWN

PDBのクローズ

```
SQL> CONNECT / AS SYSDBA
SQL> ALTER PLUGGABLE DATABASE
  2 pdb1 CLOSE IMMEDIATE;
SQL> ALTER PLUGGABLE DATABASE
  2 ALL CLOSE;
```

```
SQL> CONNECT sys@PDB1 AS SYSDBA
SQL> SHUTDOWN IMMEDIATE;
```

PDB CLOSE

PDBがクローズ
されます。

OPEN

- ルートがオープンされます。
- PDBは**読取り専用のシード**を除き、マウントされたままです。

MOUNT

- インスタンスのCDB制御ファイルがオープンされます。
- ルートがマウントされます。
- PDBがマウントされます。

NOMOUNT

インスタンスが
起動します。

SHUTDOWN

CDBのクローズ

```
SQL> CONNECT sys@CDB1 AS SYSDBA  
SQL> SHUTDOWN IMMEDIATE
```

- 全てのPDBがクローズされる。
- CDBがクローズ、ディスマウントされる。
- インスタンスが停止する。

```
SQL> CONNECT sys@PDB1 AS SYSDBA  
SQL> SHUTDOWN IMMEDIATE
```

- PDBがクローズされる。

PDBの自動オープン

データベース・イベント・トリガーによるPDBの自動オープン

```
SQL> CREATE TRIGGER Open_All_PDBs  
2 AFTER STARTUP ON DATABASE  
3 BEGIN  
4 EXECUTE IMMEDIATE 'ALTER PLUGGABLE DATABASE ALL OPNE';  
5 END Open_All_PDBs;  
6 /
```

Trigger created.

CDB、PDBにおける表領域と ユーザーの管理

CDBおよびPDBでの永続表領域の作成

非CDBと同様

- CDBへの永続表領域の作成

```
SQL> CONNECT system@CDB1
SQL> CREATE TABLESPACE tbs_CDB_users DATAFILE
 2  '/u1/app/oracle/oradata/cdb/cdb_users01.dbf'
 3  SIZE 100M;
```

- PDBへの永続表領域の作成

```
SQL> CONNECT system@PDB1
SQL> CREATE TABLESPACE tbs_PDB1_users DATAFILE
 2  '/u1/app/oracle/oradata/cdb/pdb1/users01.dbf'
 3  SIZE 100M;
```

ローカル・ユーザおよび共通ユーザ



- ローカル・ユーザ
 - 単一のPDB内に存在するユーザ
 - 従来のユーザ
 - ルートコンテナには作成出来ない

```
SQL> CREATE USER CANDY IDENTIFIED BY x;
```



- 共通ユーザ
 - ルートおよび全てのPDBに存在するユーザ
 - ルート・コンテナで作成する必要がある

```
SQL> CREATE USER C##_U IDENTIFIED BY x CONTAINER=ALL;
```

ローカル・ロールおよび共通ロール

- ローカル・ロール

- 単一のPDBのコンテキスト内に存在するロール
- 従来のロール
- ルート・コンテナには作成出来ない

```
SQL> CREATE ROLE L_HR CONTAINER=CURRENT;
```

- 共通ロール

- ルートおよび全てのPDBに存在するロール
- ルート・コンテナで作成する必要がある

```
SQL> CREATE ROLE C##_R1 CONTAINER=ALL;
```

ローカル権限および共通権限

- ローカル権限

- 単一のPDBのコンテキスト内で付与される権限
- 従来の権限

```
SQL> GRANT CREATE TABLE TO LOCAL_USER CONTAINER=CURRENT;
```

- 共通権限

- すべてのコンテナにわたって付与される権限
- 各コンテナで同じ共通ユーザーに権限を付与
- ルート・コンテナへ接続し、共通ユーザーに対して付与

```
SQL> GRANT CREATE SESSION TO C##_U CONTAINER=ALL;
```

CDB、PDBのバックアップ、 リカバリおよびフラッシュバック

CDB、PDBのバックアップ

```
$ export ORACLE_SID=cdb1  
$ rman TARGET /
```

```
RMAN> BACKUP DATABASE;  
RMAN> RECOVER DATABASE;
```

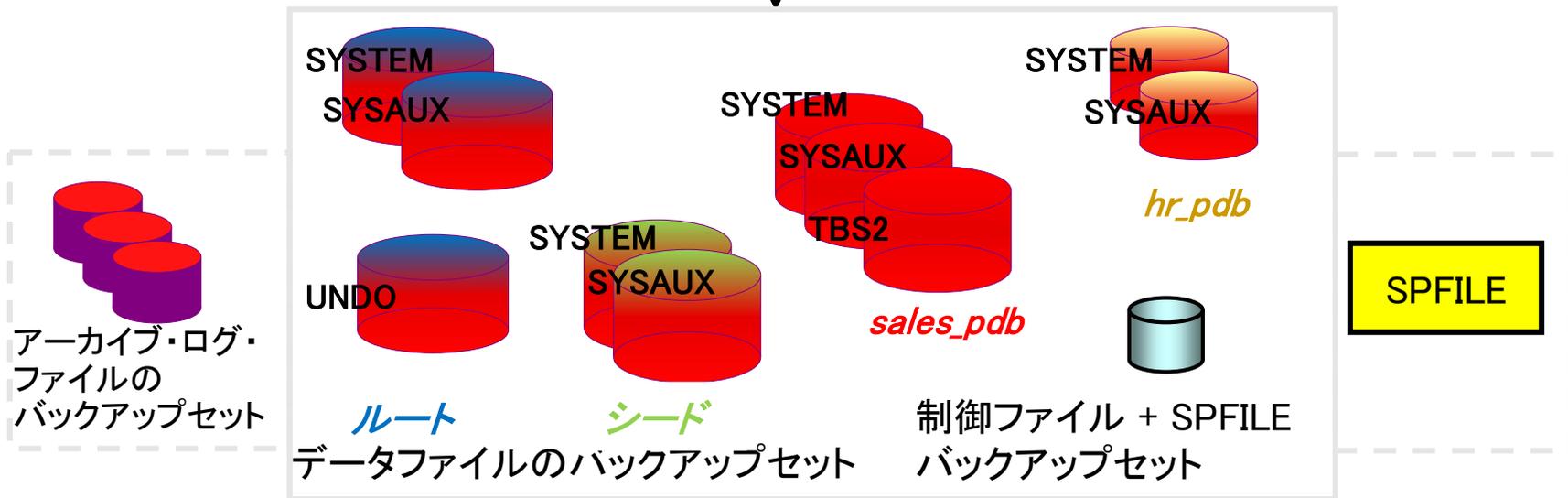
```
RMAN> BACKUP PLUGGABLE DATABASE hr_pdb, sales_pdb;  
RMAN> RECOVER PLUGGABLE DATABASE hr_pdb;
```

```
RMAN> BACKUP PLUGGABLE DATABASE "CDB$ROOT";
```

```
RMAN> BACKUP TABLESPACE sales_pdb:tbs2;  
RMAN> RESTORE TABLESPACE system;
```

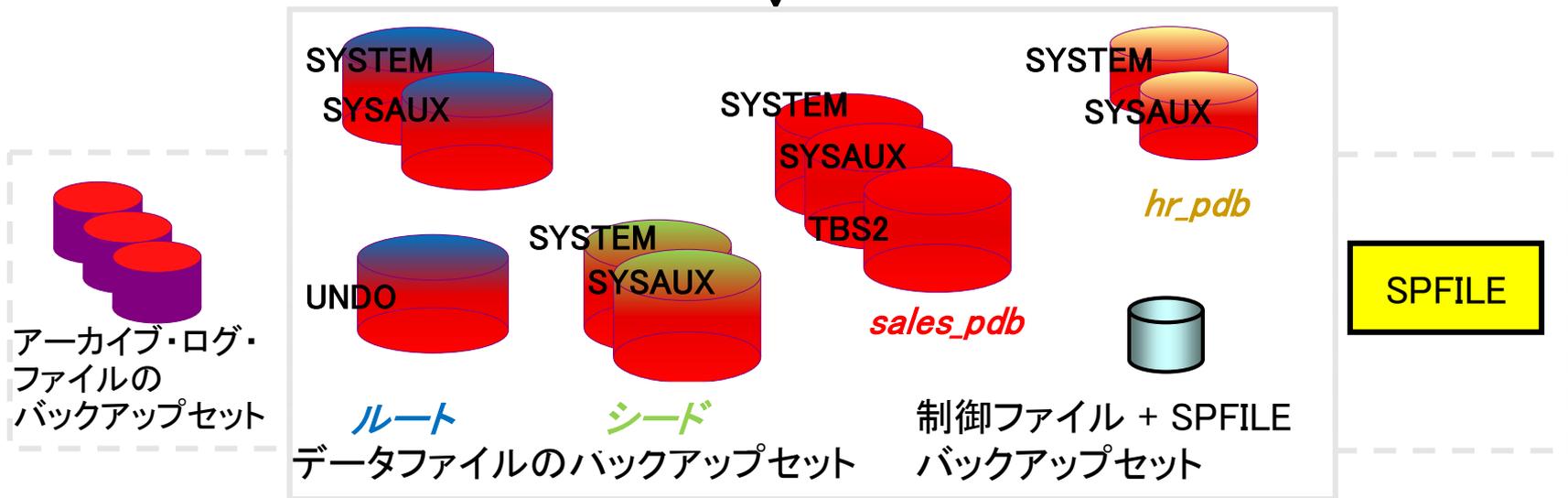
CDB全体のバックアップ

```
$ RMAN TARGET /  
RMAN> CONFIGURE CONTROLFILE AUTOBACKUP ON;  
RMAN> BACKUP DATABASE PLUS ARCHIVELOG;
```



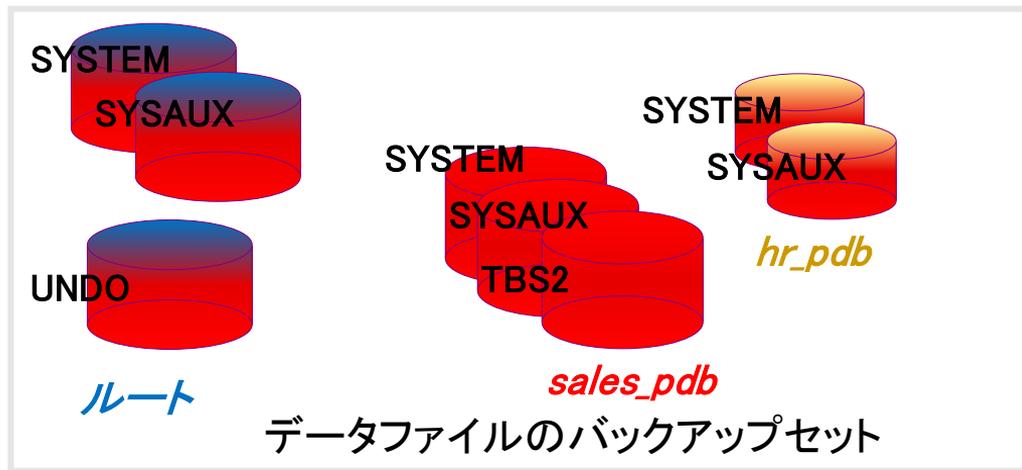
CDB部分バックアップ

```
RMAN> CONNECT TARGET /  
RMAN> BACKUP PLUGGABLE DATABASE "CDB$ROOT", sales_pdb;  
RMAN> BACKUP PLUGGABLE DATABASE hr_pdb PLUS ARCHIVELOG;
```



ユーザー管理のCDBホット・バックアップ

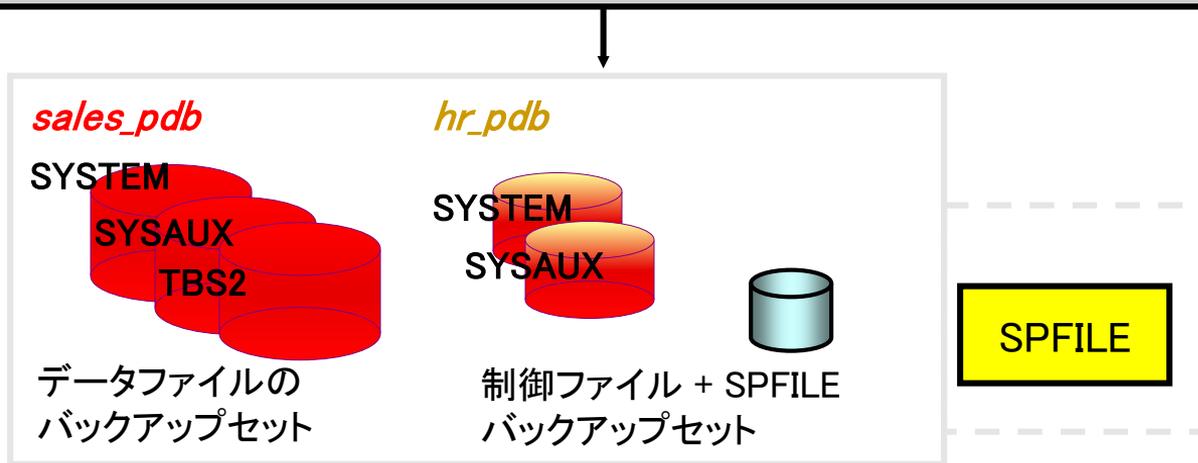
```
SQL> CONNECT / AS SYSDBA
SQL> ALTER DATABASE BEGIN BACKUP;
SQL> !cp datafiles /backup_dir
SQL> ALTER DATABASE END BACKUP;
```



PDBの全体バックアップ

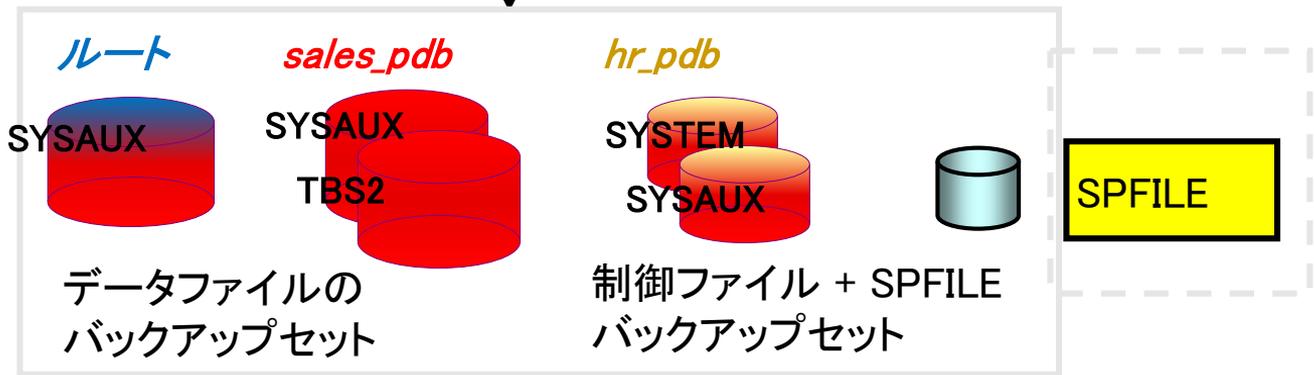
```
RMAN> CONNECT TARGET /  
RMAN> BACKUP PLUGGABLE DATABASE sales_pdb;
```

```
RMAN> BACKUP PLUGGABLE DATABASE sales_pdb, hr_pdb;
```



PDBの部分バックアップ

```
RMAN> CONNECT TARGET /  
RMAN> REPORT SCHEMA;  
RMAN> BACKUP TABLESPACE sales_pdb:tbs2;  
RMAN> BACKUP TABLESPACE hr_pdb:system,  
sales_pdb:sysaux;  
RMAN> BACKUP TABLESPACE sysaux, hr_pdb:sysaux;
```



インスタンス・リカバリ、完全リカバリ

- インスタンス・リカバリ: CDBレベルのみ
- 一時ファイルの自動的に再作成(CDBのオープン時)
- 完全なメディア・リカバリ
 - CDBはマウント: 非CDBの場合と同様
 - REDOログ・ファイル、制御ファイル、
 - SYSTEM / UNDOルート・データファイル
 - SYSTEM PDBデータファイル
 - PDBはオープン
 - SYSTEM以外のすべてのPDBデータファイル

不完全リカバリ、フラッシュバック・データベース

- 不完全なメディア・リカバリ
 - CDB全体が過去の時点に戻る。CDBはマウント。
 - PDB全体が過去の時点に戻る。PDBはマウント。
 - SYSTEM、UNDO、SYSAUX以外のすべての表領域に対するTSPITR
 - SYSTEM以外のすべてのPDBデータファイル。PDBはオープン。
- フラッシュバック・データベース: CDBはマウント。

PDB SYSTEMデータファイルのメディア・リカバリ

- PDBがオープンされている場合、CDBはマウントのみ行える。
- PDBがクローズされている場合、CDBはオープンしたままにできる。

```
RMAN> STARTUP MOUNT;      — クローズしているPDBでは不要—  
RMAN> RESTORE PLUGGABLE DATABASE sales_pdb;  
RMAN> RECOVER PLUGGABLE DATABASE sales_pdb;  
RMAN> ALTER DATABASE OPEN; — クローズしているPDBでは不要—  
RMAN> ALTER PLUGGABLE DATABASE sales_pdb OPEN;
```

PDBでの不完全リカバリ

- PDB PITR

```
RMAN> ALTER PLUGGABLE DATABASE PDB1 CLOSE;
RMAN> RUN {
    SET UNTIL SCN = 1851648 ;
    RESTORE pluggable DATABASE pdb1;
    RECOVER pluggable DATABASE pdb1
    AUXILIARY DESTINATION='/u01/app/oracle/oradata';
    ALTER PLUGGABLE DATABASE pdb1 OPEN RESETLOGS;
}
```

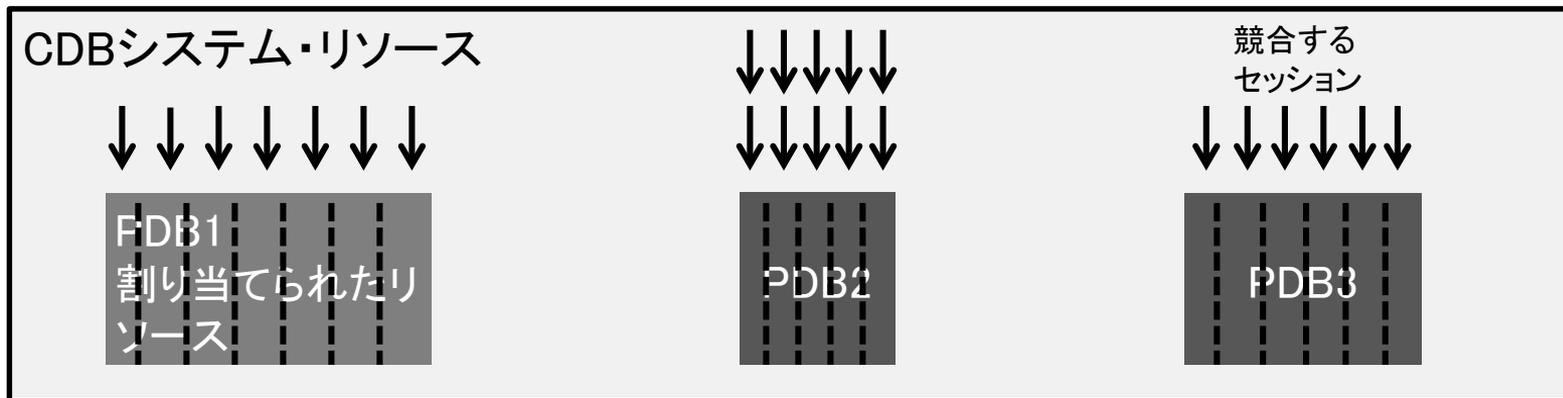
- PDB表領域PITR

```
RMAN> RECOVER TABLESPACE PDB1:TEST_TBS
2>    UNTIL SCN 832972
3>    AUXILIARY DESTINATION '/tmp/CDB1/reco';
RMAN> ALTER TABLESPACE PDB1:TEST_TBS ONLINE;
```

CDB、PDBでのリソース・ マネージャの使用

リソース・マネージャとプラグブル・データベース

- CDBでは、リソース・マネージャは次のリソースを管理可能
 - PDB間
 - 各PDB内

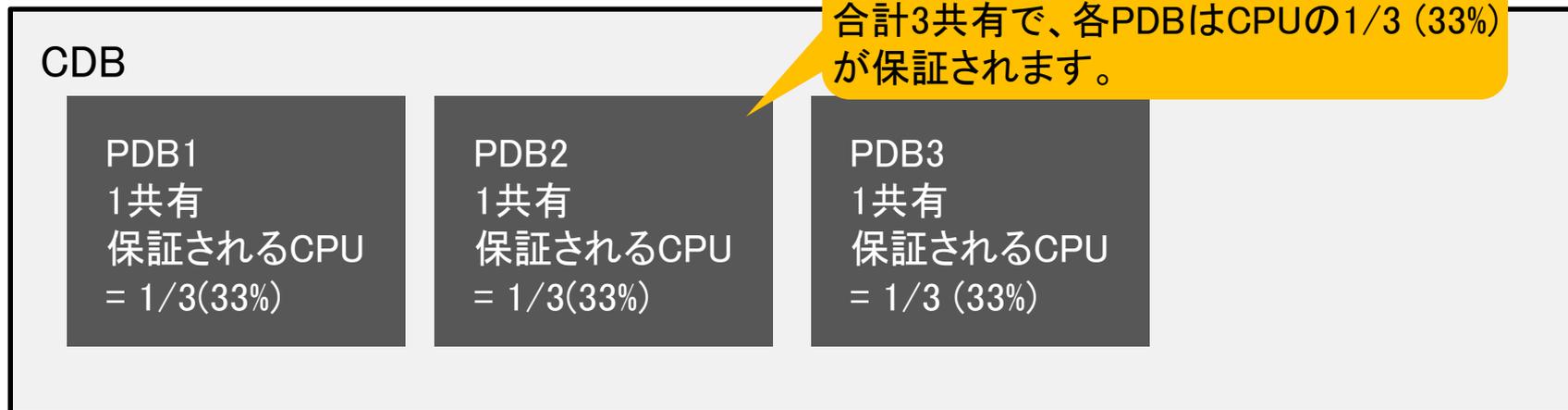


PDB間のリソースの管理

- PDBは、リソース(CPU、Exadata I/O、パラレル・サーバー)に対して競合。
 - 「共有」を使用して、各PDBにリソースを割り当てる
 - 「制限」を使用して、各PDBのリソース使用率を制限する

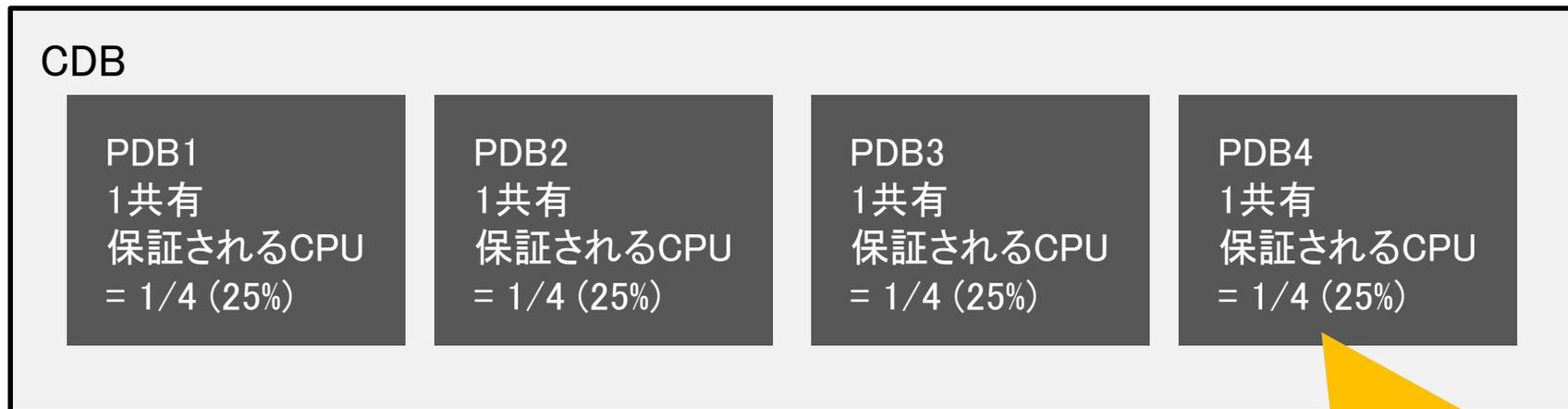


CDBリソース計画の基礎: 共有



- 共有を使用してリソースを指定
 - PDBが追加または削除された場合、再計算は必要ない。
 - PDB1はCPUの33%が保証されます。PDB1はCPUの100%に制限されます。

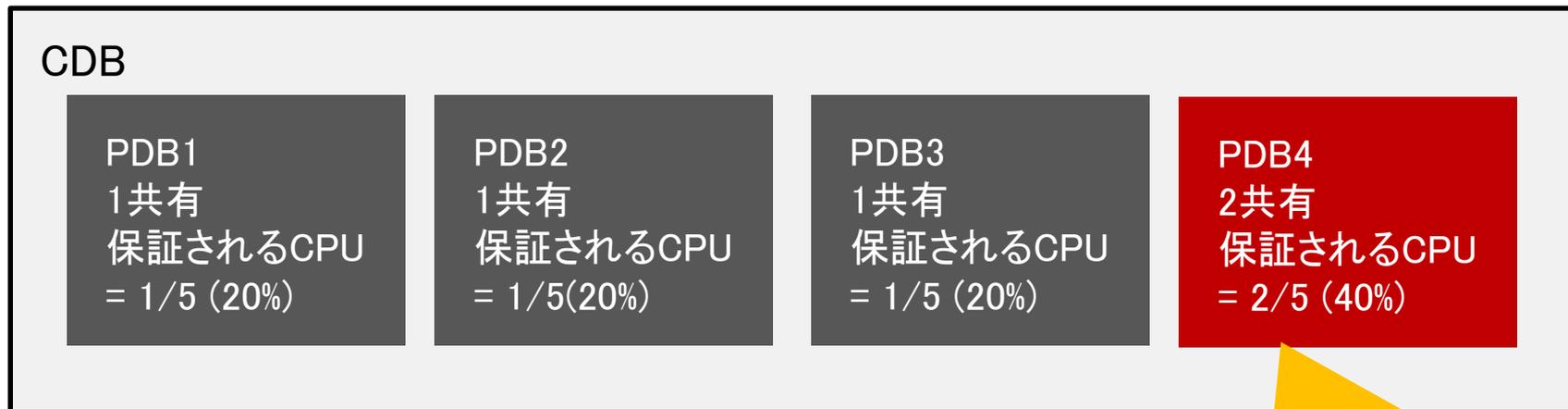
CDBリソース計画の基礎: 共有



- デフォルト割当て: 1共有

新しいPDBはデフォルト割当ての1共有を取得します。
合計4共有となり、各PDBはCPUの1/4 (25%)が保証されます。

CDBリソース計画の基礎: 共有



PDBがより重要な場合は、より多くの共有を明示的に割り当てることができます。その場合、そのPDBは他のPDBより多くのリソースを取得します。

CDBリソース計画の基礎: 制限

- PDBごとに次の2つの制限を定義可能
 - CPU、Exadata I/Oおよびパラレル・サーバーの使用率制限
 - 使用率制限を上書きするパラレル・サーバーの制限
- デフォルト値: 100%
- デフォルト値は変更可能

PDB1

`utilization_limit = 30`

`Parallel_server_limit = 50`

CDBリソース計画の作成: SQLの例

```
SQL> BEGIN
 2  DBMS_RESOURCE_MANAGER.CREATE_CDB_PLAN(
 3      plan    => 'daytime_plan',
 4      comment => 'CDB resource plan for mycdb');
 5  END;
```

```
BEGIN
  DBMS_RESOURCE_MANAGER.CREATE_CDB_PLAN_DIRECTIVE(
    plan            => 'daytime_plan',
    pluggable_database => 'hrpdb',
    shares          => 1,
    utilization_limit  => 70,
    parallel_server_limit => 70);
END;
```

PDB内のリソースの管理

- 非CDBデータベースでは、データベース内のワークロードはリソース計画によって管理
- PDBでも、ワークロードはリソース計画(PDBリソース計画とも呼ばれる)によって管理
- 次の違いを除き、機能はほぼ同じです。

| 非CDBデータベース | PDBデータベース |
|-----------------|----------------|
| 複数レベルのリソース計画 | 単一レベルのリソース計画のみ |
| 最大32コンシューマ・グループ | 最大8コンシューマ・グループ |
| サブプラン | サブプランなし |

Hardware and Software

ORACLE®

Engineered to Work Together

ORACLE®