



DOAG Expertenseminar Multitenant Database

Johannes Ahrends

- Oracle Spezialist seit 1992
 - 1992: Presales bei Oracle in Düsseldorf
 - 1999: Projektleiter bei Herrmann & Lenz Services GmbH
 - 2005: Technischer Direktor ADM Presales bei Quest Software GmbH
 - 2011: Geschäftsführer CarajanDB GmbH
- 2011 → Ernennung zum Oracle ACE
- Autor der Bücher:
 - Oracle9i für den DBA, Oracle10g für den DBA, Oracle 11g Release 2 für den DBA
- DOAG Themenverantwortlicher Datenbankadministration, Standard Edition
- DOAG Botschafter
- Hobbies:
 - Drachen steigen lassen (Kiting) draußen wie drinnen (Indoorkiting)
 - Motorradfahren (nur draußen)
 - Singen (überall)



- Architektur
- Erstellen CDB / PDB
- Administration
- Rollen, Rechte
- PDB Parameter
- Cloning, Snapshot, Proxy
- Backup und Recovery
- Migration und Upgrade
- Patching
- Data Guard
- Application Container

Architektur

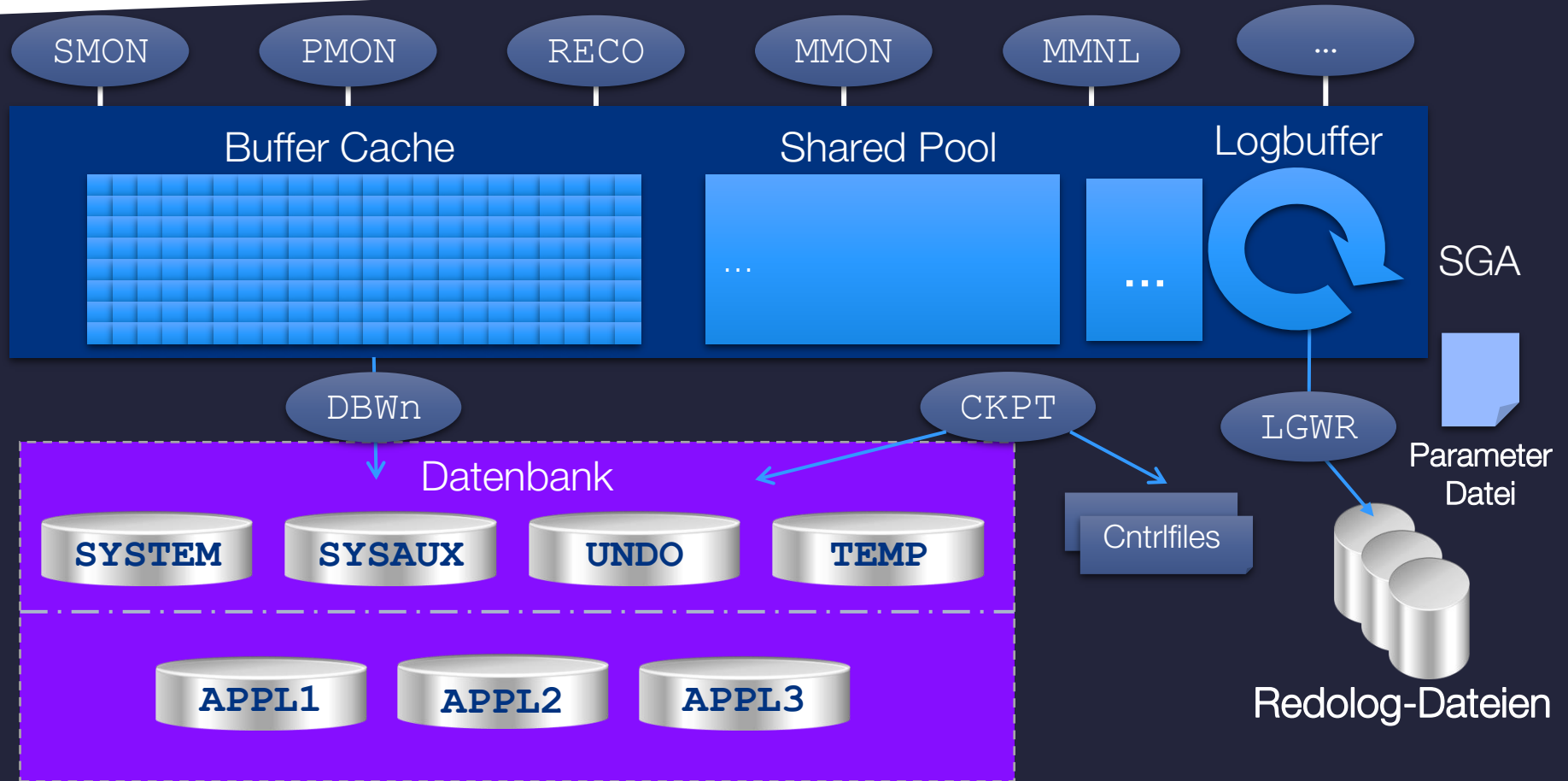
- 12.1.0.1 Eine neue Datenbankarchitektur hält Einzug: Multitenant
Besser nicht einsetzen!
- 12.1.0.2 Long Term Support Release
Multitenant brauchbar, erste Projekte
- 12.2.0.1 Flashback Pluggable Database möglich
viele neue Optionen für PDBs (Proxy PDB, Application Container, ...)
- 18c PDB Lockdown Profiles, Refreshable PDB Switchover
- 19c Drei Pluggable Databases in allen Editionen frei
- 21c Nur noch Multitenant möglich (Desupport NON-CDB)

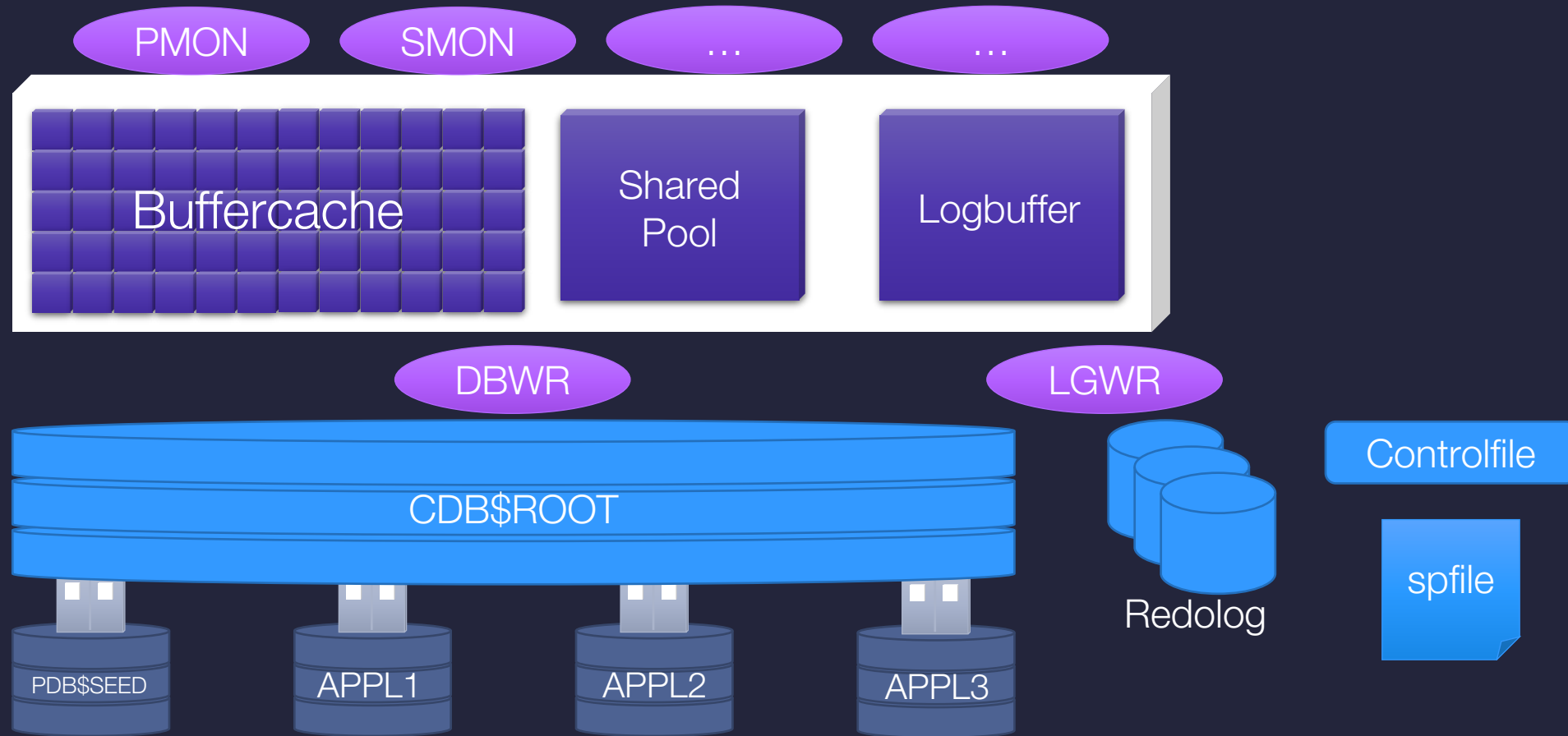
- NON-CDB
 - Architektur bis Oracle 11.2
- Multitenant Architektur
 - Alternative Architektur ab Version 12.1.1 für alle Editionen
- Multitenant Option
 - Kostenpflichtig für die Enterprise Edition
 - Bis zu 252 PDBs pro CDB (Exadata ab 12.2 4096 PDBs)
- The non-CDB architecture was deprecated in Oracle Database 12c. It can be desupported and unavailable in a release after Oracle Database 12c Release Oracle recommends use of the CDB architecture.
- The non-CDB architecture will be desupported from Oracle 20c on (OOW 2019 Dominic Gilles)

Feature / Option / Pack	SE2	EE	EE-ES	DBCS SE	DBCS EE	DBCS EE-HP	DBCS EE-EP	ExaCS	Notes
Oracle Multitenant - # of PDBs	3	252	4096	3	3	4096	4096	4096	The number value in each column indicates the maximum number of pluggable databases (PDBs) that can be created for each offering. • For all offerings, if you are not licensed for Oracle Multitenant, then you may have up to 3 PDBs in a given container database at any time. • EE: Extra cost option; if you are licensed for Oracle Multitenant, then you can create up to 252 PDBs. • EE-ES: Extra cost option; if you are licensed for Oracle Multitenant, then you can create up to 4096 PDBs. • DBCS EE-HP, DBCS EE-EP, and ExaCS: Included option; you can create up to 4096 PDBs.
CDB Fleet Management	N	N	Y	N	Y	Y	Y	Y	
PDB Snapshot Carousel	N	N	Y	N	Y	Y	Y	Y	
Refreshable PDB switchover	N	N	Y	Y	Y	Y	Y	Y	

Quelle: Oracle Database Database Licensing Information User Manual 19c E94254-10 September 2019

NON-CDB Database





Root Container (CDB)

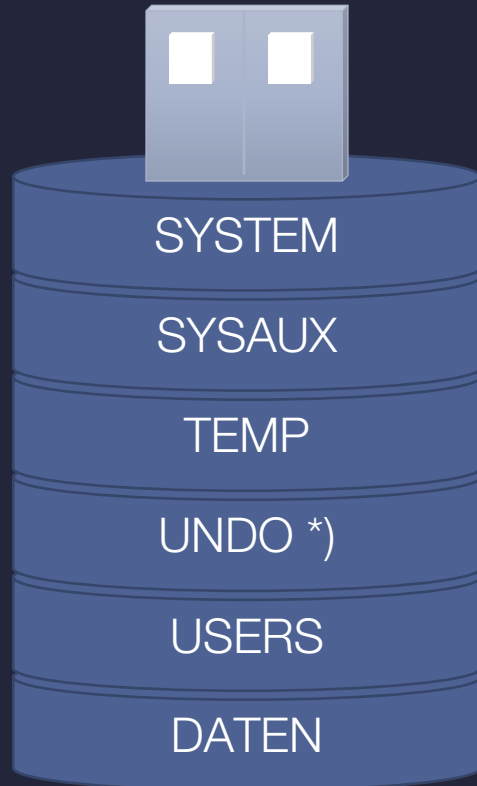


Controlfile

spfile

- Komponenten für den Aufbau der Instanz
 - Controlfiles
 - spfile
 - Connect an die Instanz nur über die CDB möglich
- Verwaltung der Redolog-Dateien
 - Vollständiges Recovery der Datenbank nur über die CDB möglich

Pluggable Database (PDB)



- In sich geschlossenes System
 - Data Dictionary
 - Tablespaces
 - Schemata
 - Objekte
 - Public Objekte
 - Workload Repository
 - Lokales Undo (nicht immer)
 - Parameter

Warum Pluggable Database?

- Konsolidierung von Datenbanken
- Eine gemeinsame Instanz
- Ein gemeinsames Redo Management
- Separate Tablespace Verwaltung
- Separate Schema Verwaltung
- Separate User / Rollen-Verwaltung

- **CONTAINER** → Eigenständiger Bereich einer Datenbank
(kann PDB oder CDB sein)
Besteht mindestens aus SYSTEM und SYSAUX Tablespace
- **CDB** → Container Database (**CDB\$ROOT**)
- **PDB** → Pluggable Database
- **CON_ID** → Identifier für jede Datenbank
 - **0** → Gesamtheit aller Datenbanken
 - **1** → **CDB**
 - **2** → **PDB\$SEED**
 - **>2** → PDB
- **GUID** → eindeutiger Bezeichner von CDB und PDB

Erstellen einer CDB

Erstellen einer CDB

- Eine CDB kann nur neu erstellt werden
 - Keine Migration möglich
- Upgrade und Patching auf CDB-Ebene

Database Configuration Assistant - Create a database - Step 2 of 14@lindenberg.carajandb.local

19c ORACLE Database

Select Database Creation Mode

Database Operation

- Creation Mode
- Deployment Type
- Database Identification
- Storage Option
- Fast Recovery Option
- Database Options
- Configuration Options
- Management Options
- User Credentials
- Creation Option
- Summary
- Progress Page
- Finish

☐ Typical configuration

Global database name: orcl.carajandb.local

Storage type: File System

Database files location: {ORACLE_BASE}/oradata/{DB_UNIQUE_NAME}

Fast Recovery Area (FRA): {ORACLE_BASE}/fast_recovery_area/{DB_UNIQUE_NAME}

Database character set: AL32UTF8 - Unicode UTF-8 Universal character set

Administrative password:

Confirm password:

☒ Create as Container database

Pluggable database name:

☒ **Advanced configuration**

Database Configuration Assistant - Create a database - Step 3 of 14@lindenberg.carajandb.local

19c ORACLE Database

Select Database Deployment Type

Select the type of database you want to create.

Database type:

Configuration type:

Select a template for your database.

Templates that include datafiles contain pre-created databases. They allow you to create a new database quickly. Use templates without datafiles only when necessary, such as when you need to change attributes like block size that cannot be altered after database creation.

	Template name	Include datafiles	Details
☐	Data Warehouse	Yes	View details
☒	Custom Database	No	View details
☐	General Purpose or Transaction Processing	Yes	View details

Template location: /u01/app/oracle/product/19/dbhome_1/assistants/dbca/templates

Database Configuration Assistant - Create a database - Step 4 of 14@lindenberg.carajandb.local

Specify Database Identification Details

19c ORACLE Database

Provide a unique database identifier information. An Oracle database is uniquely identified by a Global database name, typically of the form "name.domain".

Global database name:

SID:

Service name:

☒ **Create as Container database**

A Container database can be used for consolidating multiple databases into a single database, and it enables database virtualization. A Container database (CDB) can have zero or more pluggable databases (PDB).

☒ **Use Local Undo tablespace for PDBs**

☒ **Create an empty Container database**

☐ Create a Container database with one or more PDBs

Number of PDBs:

PDB name:

Help < Back Next > Finish Cancel

DBCA Oracle Managed Files (1)

Database Configuration Assistant - Create 'UDO' database - Step 5 of 14@lindenberg.carajandb.local

19c ORACLE Database

Select Database Storage Option

- Database Operation
- Creation Mode
- Deployment Type
- Database Identification
- Storage Option**
- Fast Recovery Option
- Database Options
- Configuration Options
- Management Options
- User Credentials
- Creation Option
- Summary
- Progress Page
- Finish

☐ Use template file for database storage attributes
Storage type and location for database files will be picked up from the specified template (Custom Database).

☒ Use following for the database storage attributes
All the database files will be put at the specified location below. You can customize the name and location of each datafile in the subsequent screen.

Database files storage type:

Database files location:

Oracle Managed files option will enable Oracle to automatically generate the names of the datafiles for simplified database management.

☒ **Use Oracle-Managed Files (OMF)**

Database Configuration Assistant - Create 'UDO' database - Step 5 of 14@lindenberg.carajandb.local

19c ORACLE Database

Select Database Storage Option

- Database Operation
- Creation Mode
- Deployment Type
- Database Identification
- Storage Option**
- Fast Recovery Option
- Database Options
- Configuration Options
- Management Options
- User Credentials
- Creation Option
- Summary
- Progress Page
- Finish

☐ Use template file for database storage attributes
Storage type and location for database files will be picked up from the specified template (Custom Database).

☒ Use following for the database storage attributes
All the database files will be put at the specified location below. You can customize the name and location of each datafile in the subsequent screen.

Database files storage type:

Database files location:

Oracle Managed files option will enable Oracle to automatically generate the names of the datafiles for simplified database management.

☒ **Use Oracle-Managed Files (OMF)**

DBCA Oracle Managed Files (2)

Database Configuration Assistant - Create 'UDO' database - Step 6 of 14@lindenberg.carajandb.local

19^c ORACLE[®] Database

Select Fast Recovery Option

Choose the recovery options for the database.

☒ **Specify Fast Recovery Area**

Recovery files storage type: File System

Fast Recovery Area: **{ORACLE_BASE}/fast_recovery_area/{DB_UNIQUE_NAME}** Browse...

Fast Recovery Area size: 12561 MB

☐ Enable archiving Edit archive mode parameters...

Navigation: < Back Next > Finish Cancel

Database Configuration Assistant - Step 6 of 14@lindenberg.carajandb.local

19^c ORACLE[®] Database

options for the database.

Fast Recovery Area

Storage type: File System

Fast Recovery Area: /u03/fast_recovery_area/ Browse...

Fast Recovery Area size: 20000 MB

☐ Enable archiving Edit archive mode parameters...

Navigation: < Back Next > Finish Cancel

Database Configuration Assistant - Create 'UDO' database - Step 8 of 15@lindenberg.carajandb.local

19c ORACLE Database

Select Database Options

- Database Operation
- Creation Mode
- Deployment Type
- Database Identification
- Storage Option
- Fast Recovery Option
- Network Configuration
- Database Options**
- Configuration Options
- Management Options
- User Credentials
- Creation Option
- Summary
- Progress Page
- Finish

Database components

Select the standard database components you want to configure for use in your database. Oracle recommends that you always install these components in your database. Deselecting these components may cause you to no longer be able to choose some components on the subsequent page.

Select Component	Tablespace	Include in PDBs
☐ Oracle JVM	SYSTEM	☐
☒ Oracle Text	SYSAUX	☒
☐ Oracle Multimedia	SYSAUX	☐
☐ Oracle OLAP	SYSAUX	☐
☐ Oracle Spatial	SYSAUX	☐
☐ Oracle Label Security	SYSTEM	☐
☐ Oracle Application Express	SYSAUX	☐
☐ Oracle Database Vault	SYSAUX	☐

Help < Back Next > Finish Cancel

Database Configuration Assistant - Create 'UDO' database - Step 12 of 15@lindenberg.carajandb.local

Select Database Creation Option

Select the database creation options.

☐ Create database

Specify the SQL scripts you want to run after the database is created, listed below.

Post DB creation scripts:

☐ Save as a database template

Template name:

Template location:

Description:

☒ **Generate database creation scripts**

Destination directory:

Following advanced configuration options can be used to configure and customize database storage locations.

All Initialization Parameters...

[All initialization parameters@lindenberg.carajandb.local](#)

Update the initialization parameters only when it is required. Refer to the Oracle documentation to learn more about each initialization parameter and its valid set of values.

(Storage related parameter(s) value is shown in MB) ☐ Show advanced parameters

Name	Value	Include in spfile	Category
processes	300	☒	Processes and Sessions
pga_aggregate_target	200	☒	Sort, Hash Joins, Bitmap Indexes
nls_territory	AMERICA	☒	NLS
db_recovery_file_dest_size	20000	☒	File Configuration
open_cursors	300	☒	Cursors and Library Cache
compatible	19.0.0	☒	Miscellaneous
db_name	UDO	☒	Database Identification
db_recovery_file_dest	/u03/fast_recovery_area/	☒	File Configuration
cluster_database	FALSE	☐	Cluster Database
log_archive_dest_1		☐	Archive
log_archive_dest_2		☐	Archive
shared_servers	1	☐	Shared Server
control_files	("/u02/oradata/{DB_UNIQUE_NAME}/control01.ctl, /u02/oradata/{DB_UNIQUE_NAME}/control02.ctl)	☐	File Configuration
db_create_online_log_dest_2		☐	File Configuration
db_create_online_log_dest_1		☐	File Configuration
db_unique_name	UDO_S1	☐	Standby Database
remote_listener		☐	Network Registration
db_domain		☐	Database Identification
sessions	504	☐	Processes and Sessions

Description:

db_unique_name: Database Unique Name. Default Value: Database instances: the value of DB_NAME. Oracle Automatic Storage Management instances: +ASM.

```
% cat /u01/app/oracle/admin/UDO_S1/scripts/init.ora
...
#####
# Database Identification
#####
db_name="UDO"
db_unique_name="UDO_S1"
#####
# File Configuration
#####
db_create_file_dest="/u02/oradata"
db_recovery_file_dest="/u03/fast_recovery_area/"
db_recovery_file_dest_size=20000m
#####
# Miscellaneous
#####
compatible=19.0.0
diagnostic_dest=/u01/app/oracle
enable_pluggable_database=true
...
```

```
% cat /u01/app/oracle/admin/UDO_S1/scripts/createDB.sql
...
CREATE DATABASE "UDO"
...
DATAFILE SIZE 700M AUTOEXTEND ON NEXT 100M MAXSIZE 2000M EXTENT MANAGEMENT LOCAL
SYSAUX DATAFILE SIZE 500M AUTOEXTEND ON NEXT 100M MAXSIZE 2000M
DEFAULT TEMPORARY TABLESPACE TEMP TEMPFILE SIZE 100M AUTOEXTEND ON NEXT 100M MAXSIZE 2000M
UNDO TABLESPACE "UNDOTBS1" DATAFILE SIZE 200M AUTOEXTEND ON NEXT 100M MAXSIZE 2000M
DEFAULT TABLESPACE "USERS" LOGGING DATAFILE SIZE 100M AUTOEXTEND ON NEXT 100M MAXSIZE 2000M
CHARACTER SET AL32UTF8
NATIONAL CHARACTER SET AL16UTF16
LOGFILE GROUP 1 SIZE 200M,
GROUP 2 SIZE 200M,
GROUP 3 SIZE 200M
USER SYS IDENTIFIED BY "&&sysPassword" USER SYSTEM IDENTIFIED BY "&&systemPassword"
enable pluggable database
LOCAL UNDO ON;
```

Create Database Skript - NONOMF

```
% cat /u01/app/oracle/admin/FRANZ_S1/scripts/createDB.sql
...
CREATE DATABASE "FRANZ"
...
DATAFILE '/u02/oradata/FRANZ/system01.dbf' SIZE 700M REUSE AUTOEXTEND ON NEXT 100M MAXSIZE
2000M EXTENT MANAGEMENT LOCAL
...
SMALLFILE TABLESPACE "USERS" LOGGING DATAFILE '/u02/oradata/FRANZ/users01.dbf' SIZE 100M
REUSE AUTOEXTEND ON NEXT 100M MAXSIZE 2000M
...
enable pluggable database
seed
file_name_convert=('/u02/oradata/FRANZ/system01.dbf', '/u02/oradata/FRANZ/pdbseed/system01.db
f', '/u02/oradata/FRANZ/sysaux01.dbf', '/u02/oradata/FRANZ/pdbseed/sysaux01.dbf', '/u02/oradata
/FRANZ/temp01.dbf', '/u02/oradata/FRANZ/pdbseed/temp01.dbf', '/u02/oradata/FRANZ/undotbs01.dbf
', '/u02/oradata/FRANZ/pdbseed/undotbs01.dbf')
LOCAL UNDO ON;
spool off
```

- Besser:

```
% cat /u01/app/oracle/admin/FRANZ_S1/scripts/createDB.sql
...
CREATE DATABASE "FRANZ"
...
DATAFILE '/u02/oradata/FRANZ/system01.dbf' SIZE 700M REUSE AUTOEXTEND ON NEXT 100M MAXSIZE
2000M EXTENT MANAGEMENT LOCAL
...
SMALLFILE TABLESPACE "USERS" LOGGING DATAFILE '/u02/oradata/FRANZ/users01.dbf' SIZE 100M
REUSE AUTOEXTEND ON NEXT 100M MAXSIZE 2000M
...
enable pluggable database
seed file_name_convert=('FRANZ','FRANZ/pdbseed')
LOCAL UNDO ON;
spool off
```

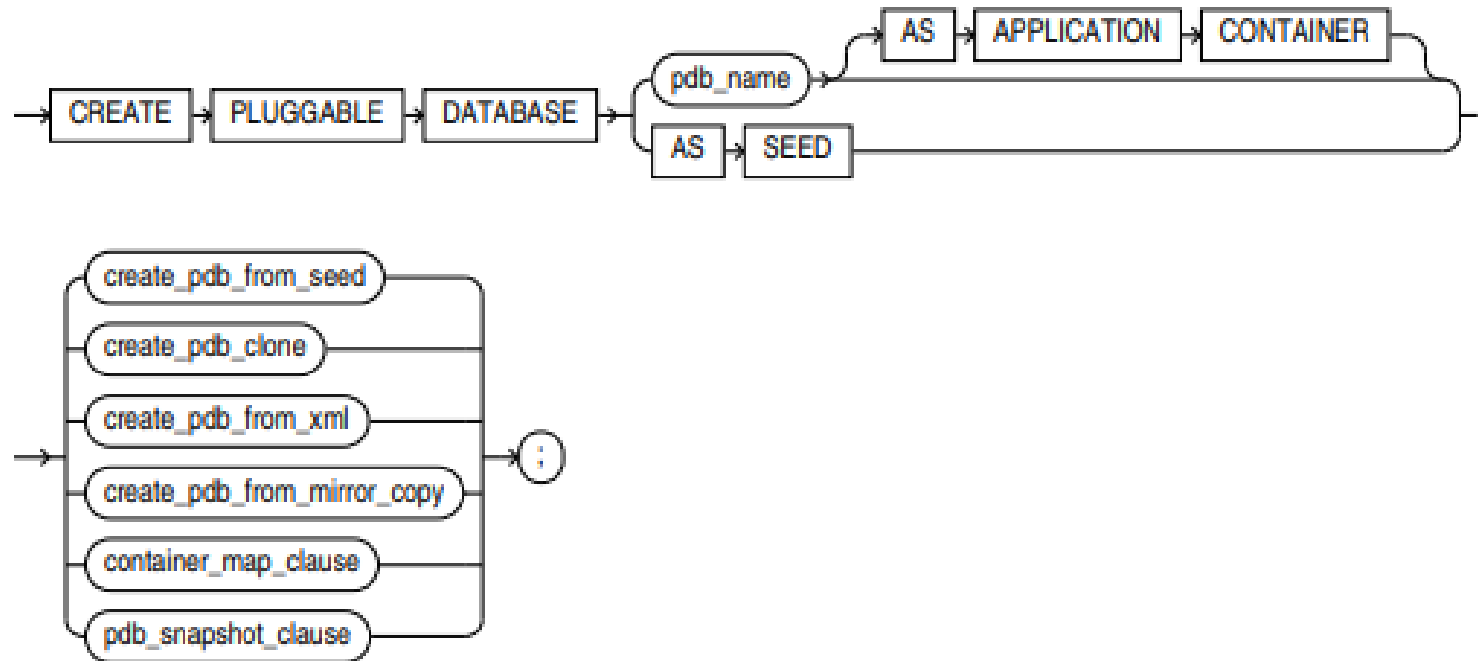
Erstellen einer PDB

- Eine PDB kann nicht erstellt werden!
 - PDBs können ausschließlich kopiert / geklont werden
 - Ausnahme: Migration NON-CDB → PDB
- Den Befehl „Plugin“ gibt es nicht!
 - Eine PDB kann nur mit dem Befehl „CREATE“ in einer CDB angelegt werden
- Den Befehl „Unplug“ gibt es!
 - Mit dem Befehl wird die PDB aus der CDB herausgenommen
 - Der Befehl ist irreversibel

Create Pluggable Database

Syntax

create_pluggable_database::=



MOS Note:2378735.1 – “`_ORACLE_SCRIPT`”=TRUE PARAMETER Should not be Invoked by Users

Per Development, setting `_ORACLE_SCRIPT` to `TRUE` is something to be used only by Oracle internally. There may be a case where a note says to use it explicitly, but you should not use it outside the context of that note. In general is not to be set explicitly by users.

...

As soon as you do DDLs, the objects will have an `ORACLE_MAINTAINED=Y` flag. And this potentially leads to issues during your next upgrade.

Quelle: https://mikedietrichde.com/2020/02/10/be-aware-when-you-use-_oracle_script-in-scripts

- Pluggable Database Template
- existiert immer
- kann nicht / eingeschränkt geändert werden
- Read Only geöffnet
- Für Patching read-write geöffnet
 - Bis 12.2 `ALTER SYSTEM SET "_ORACLE_SCRIPT"=TRUE;`

- Enthält ein Basis-Data Dictionary
- Enthält die notwendigen Tablespaces
 - SYSTEM
 - SYSAUX
 - UNDO
- Kann ein Default Tablespace enthalten
 - Muss beim Erstellen der CDB mit angelegt worden sein

Create PDB from PDB\$SEED - OMF

```
SQL> CREATE PLUGGABLE DATABASE cello  
      ADMIN USER pdbadmin IDENTIFIED BY manager;
```

- Bei Oracle Managed Files keine weiteren Angaben notwendig

- Lokaler PDB Administrator (kein Common User)

```
SQL> SELECT con_id,grantee,granted_role FROM cdb_role_privs WHERE grantee='PDBADMIN'
```

CON_ID	GRANTEE	GRANTED_ROLE
3	PDBADMIN	PDB_DBA

```
SQL> SELECT con_id,grantee,privilege FROM cdb_sys_privs WHERE grantee='PDBADMIN';
```

```
no rows selected
```

```
SQL> SELECT con_id,grantee,table_name,privilege FROM cdb_tab_privs WHERE grantee='PDB_DBA'
```

CON_ID	GRANTEE	TABLE_NAME	PRIVILEGE
1	PDB_DBA	PDB_PLUG_IN_VIOLATIONS	SELECT
1	PDB_DBA	PDB_ALERTS	SELECT
3	PDB_DBA	PDB_PLUG_IN_VIOLATIONS	SELECT
3	PDB_DBA	PDB_ALERTS	SELECT

```
SQL> SELECT con_id,grantee,privilege FROM cdb_sys_privs WHERE grantee='PDB_DBA';
```

CON_ID	GRANTEE	PRIVILEGE
3	PDB_DBA	CREATE PLUGGABLE DATABASE
3	PDB_DBA	CREATE SESSION

```
SQL> SELECT con_id,grantee,granted_role FROM cdb_role_privs WHERE grantee='PDB_DBA';
```

```
no rows selected
```

```
SQL> SELECT con_id, tablespace_name, file_name FROM cdb_data_files ORDER BY 1,2;
```

CON_ID	TABLESPACE_NAME	FILE_NAME
1	SYSAUX	/u02/oradata/UDO_S1/datafile/o1_mf_sysaux_h522h356_.dbf
1	SYSTEM	/u02/oradata/UDO_S1/datafile/o1_mf_system_h522gqm_.dbf
1	UNDOTBS1	/u02/oradata/UDO_S1/datafile/o1_mf_undotbs1_h522h7yv_.dbf
1	USERS	/u02/oradata/UDO_S1/datafile/o1_mf_users_h522h9fb_.dbf
2	SYSAUX	/u02/oradata/UDO_S1/9F290C149A1D0795E055FB909B80DF97/datafile/o1_mf_sysaux_h522h4rb_.dbf
2	SYSTEM	/u02/oradata/UDO_S1/9F290C149A1D0795E055FB909B80DF97/datafile/o1_mf_system_h522gtfk_.dbf
2	UNDOTBS1	/u02/oradata/UDO_S1/9F290C149A1D0795E055FB909B80DF97/datafile/o1_mf_undotbs1_h522h8w7_.dbf
2	USERS	/u02/oradata/UDO_S1/9F290C149A1D0795E055FB909B80DF97/datafile/o1_mf_users_h522h9x4_.dbf
3	SYSAUX	/u02/oradata/UDO_S1/9F2C4A9C6B704912E055FB909B80DF97/datafile/o1_mf_sysaux_h52j25f3_.dbf
3	SYSTEM	/u02/oradata/UDO_S1/9F2C4A9C6B704912E055FB909B80DF97/datafile/o1_mf_system_h52j25dc_.dbf
3	UNDOTBS1	/u02/oradata/UDO_S1/9F2C4A9C6B704912E055FB909B80DF97/datafile/o1_mf_undotbs1_h52j25f4_.dbf
3	USERS	/u02/oradata/UDO_S1/9F2C4A9C6B704912E055FB909B80DF97/datafile/o1_mf_users_h52j25f5_.dbf

- Schwierig, wenn PDBs und CDBs unterschiedliche Filesysteme bekommen sollen

Create PDB from PDB\$SEED - NONOMF

```
SQL> CREATE PLUGGABLE DATABASE horsti  
      ADMIN USER pdbadmin IDENTIFIED BY manager  
      FILE_NAME_CONVERT=('pdbseed','horsti');
```

- FILE_NAME_CONVERT zwingend notwendig

Datafile Layout NON-OMF

```
SQL> SELECT con_id, tablespace_name, file_name FROM cdb_data_files ORDER BY 1,2;
```

CON_ID	TABLESPACE_NAME	FILE_NAME
1	SYSAUX	/u02/oradata/FRANZ/sysaux01.dbf
1	SYSTEM	/u02/oradata/FRANZ/system01.dbf
1	UNDOTBS1	/u02/oradata/FRANZ/undotbs01.dbf
1	USERS	/u02/oradata/FRANZ/users01.dbf
4	SYSAUX	/u02/oradata/FRANZ/horsti/sysaux01.dbf
4	SYSTEM	/u02/oradata/FRANZ/horsti/system01.dbf
4	UNDOTBS1	/u02/oradata/FRANZ/horsti/undotbs01.dbf
4	USERS	/u02/oradata/FRANZ/horsti/users01.dbf

DEFAULT TABLESPACE

- Default Tablespace wird standardmäßig nicht mit angelegt
- Ausnahme: Ändern des CREATE DATABASE Befehls
- Beim Erstellen einer PDB:

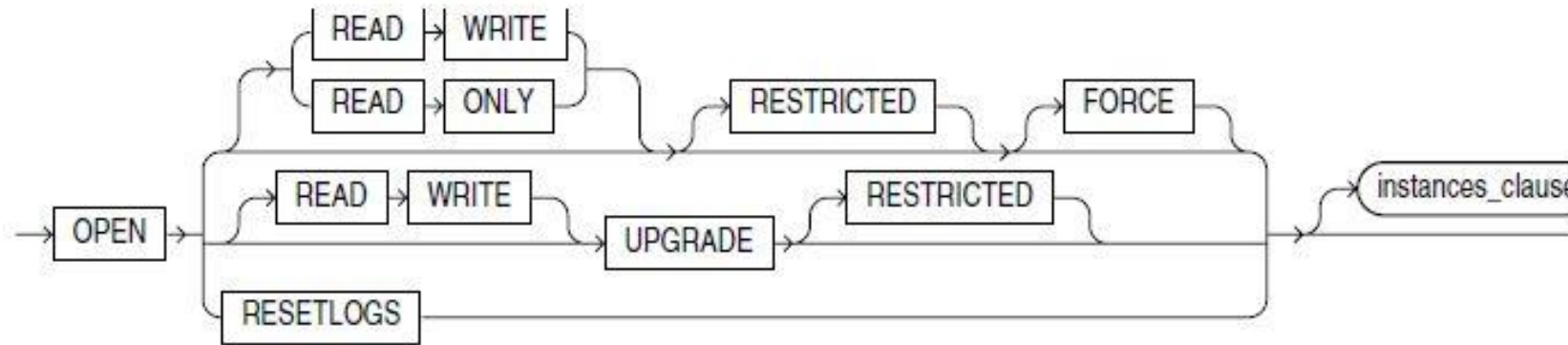
```
SQL> CREATE PLUGGABLE DATABASE ...  
      DEFAULT TABLESPACE USERS ...
```

PDB Management

- Öffnen einer oder mehrerer Pluggable Databases

```
ALTER PLUGGABLE DATABASE [<PDB>,<PDB>,ALL] [EXCEPT <PDB>] OPEN ...
```

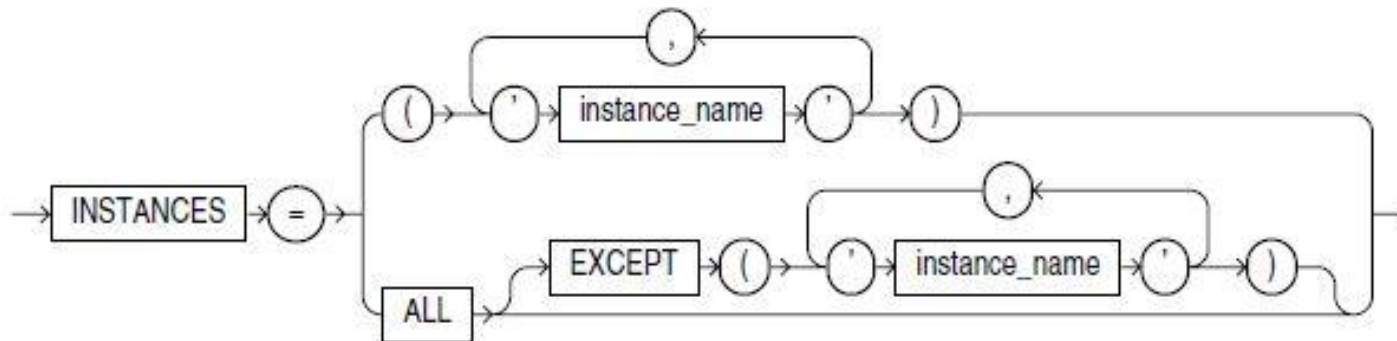
pdb_open::=



- Beim ersten Öffnen wird die Verbindung zum Data Dictionary der CDB hergestellt
→ dauert länger!

- Bei RAC Datenbanken
 - Angabe von Instanzen
 - ... außer

instances_clause::=



PDB öffnen – aus der PDB heraus

- <PLUGGABLE> kann weggelassen werden

```
ALTER PLUGGABLE DATABASE OPEN ...
```

- Aus der CDB nicht möglich!
 - STARTUP PLUGGABLE DATABASE gibt es nicht
- Aus der Pluggable Database heraus:

```
STARTUP
```

BITTE NICHT VERWENDEN!

- Öffnen einer einzelnen PDB

```
SQL> ALTER PLUGGABLE DATABASE cello OPEN INSTANCES=ALL;
```

- Öffnen aller PDBs

```
SQL> ALTER PLUGGABLE DATABASE all OPEN INSTANCES=ALL;
```

- Schließen und Öffnen einer PDBs READ ONLY

```
SQL> ALTER PLUGGABLE DATABASE cello OPEN READ ONLY FORCE INSTANCES=ALL;
```

- Ab 12.1.0.2 → derzeitiger Status der PDB über Instanz-Restart sichern

```
SQL> ALTER PLUGGABLE DATABASE [<PDB>,<PDB>,ALL] [EXCEPT <PDB>] SAVE STATE INSTANCES=...
```

- Rückgängig machen mit

```
SQL> ALTER PLUGGABLE DATABASE [<PDB>,<PDB>,ALL] [EXCEPT <PDB>] DISCARD STATE
```

- Abfrage mit:

```
SQL> SELECT con_id, con_name, instance_name, state, restricted  
       FROM cdb_pdb_saved_states;
```

CON_ID	CON_NAME	INSTANCE_NAME	STATE	RES
3	CELLO	UD01	OPEN	NO
3	CELLO	UD02	OPEN	NO

- Aus der CDB nicht möglich!
 - SHUTDOWN PLUGGABLE DATABASE gibt es nicht
- Aus der Pluggable Database heraus:

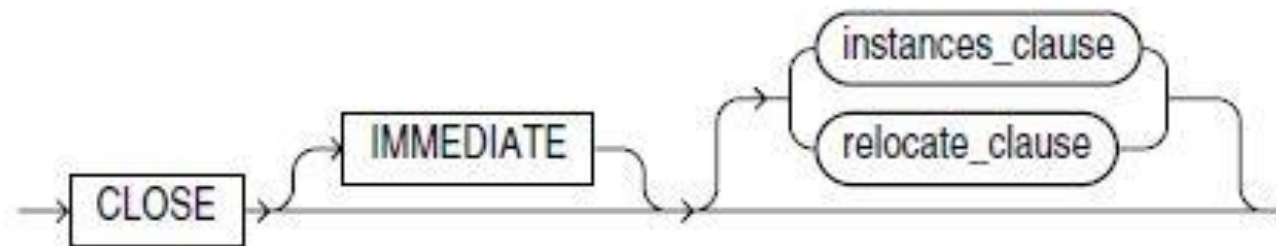
```
SHUTDOWN
```

BITTE NICHT VERWENDEN!

- Wesentlich umfangreicherer Befehlssatz
 - Schließen einer oder mehrerer Pluggable Databases

```
ALTER PLUGGABLE DATABASE [<PDB>,<PDB>,ALL] [EXCEPT <PDB>] CLOSE ...
```

pdb_close::=



PDB schließen – aus der PDB heraus

- <PLUGGABLE> kann weggelassen werden

```
ALTER PLUGGABLE DATABASE CLOSE ...
```

- oder

```
ALTER DATABASE CLOSE ...
```

- Es gibt zwar den Befehl „UNPLUG“ aber nicht „PLUG“
- „UNPLUG“ ist nur aus der CDB möglich
- Nur eine nicht geöffnete PDB kann „UNPLUGGED“ werden
- Die Metainformation wird in eine XML-Datei gespeichert (Endung .xml zwingend)

```
ALTER PLUGGABLE DATABASE <PDB> UNPLUG INTO '<filename>.xml';
```

- Eine „UNPLUGGED“ Database wird trotzdem noch mit gesichert

- Schließen der PDB

```
SQL> ALTER PLUGGABLE DATABASE cello CLOSE IMMEDIATE;
```

- Unplug der PDB

```
SQL> ALTER PLUGGABLE DATABASE cello UNPLUG INTO '/u02/oradata/cello.xml';
```

- Dateiänderung „.pdb“

```
SQL> ALTER PLUGGABLE DATABASE cello UNPLUG INTO '/u02/oradata/UDO/cello.pdb';

% ls -l /u02/oradata/UDO/cello.pdb
-rw-r--r-- 1 oracle oinstall 158679850 Dec 10 17:07 /u02/oradata/UDO/cello.pdb

% unzip -l /u02/oradata/UDO/cello.pdb
Archive:  cello.pdb
  Length      Date    Time    Name
  -----  -
272629760  12-10-2020  17:06    o1_mf_system_hx3skvqh_.dbf
340787200  12-10-2020  17:06    o1_mf_sysaux_hx3skvqr_.dbf
209715200  12-10-2020  17:06    o1_mf_undotbs1_hx3skvqs_.dbf
104857600  12-10-2020  17:07    o1_mf_users_hx3skvqv_.dbf
      7624  12-10-2020  17:07    /u02/oradata/UDO/cello.xml
  -----
927997384
                    5 files
```


- Information über „UNPLUGGED“ PDB

```
SQL> SELECT pdb_id, pdb_name, guid, status, con_id  
FROM cdb_pdb;
```

PDB_ID	PDB_NAME	GUID	STATUS	CON_ID
2	PDB\$SEED	9F290C149A1D0795E055FB909B80DF97	NORMAL	2
3	CELLO	9F2C4A9C6B704912E055FB909B80DF97	NORMAL	3
4	DORIA	9F3AC1DE46E50A89E0554D86E6126571	NORMAL	4
6	ANDREA	9F3AC1DE46ED0A89E0554D86E6126571	UNPLUGGED	6

- Keine Information über Status (Mount, Open, ...)

- V\$PDBS → welche PDBs gibt es (außer CDB)

```
SQL> SELECT con_id, name, open_mode  
FROM v$pdb;
```

CON_ID	NAME	OPEN_MODE
2	PDB\$SEED	READ ONLY
3	CELLO	READ WRITE
4	DORIA	READ WRITE
5	ANDREA	MOUNTED

- Keine Informationen über „*UNPLUGGED*“ PDBs

- v\$containers → welche PDBs gibt es (inclusive CDB)

```
SQL> SELECT con_id, name, guid, open_mode, restricted  
FROM v$containers;
```

CON_ID	NAME	GUID	OPEN_MODE	RES
1	CDB\$ROOT	9F290C149A1C0795E055FB909B80DF97	READ WRITE	NO
2	PDB\$SEED	9F290C149A1D0795E055FB909B80DF97	READ ONLY	NO
3	CELLO	9F2C4A9C6B704912E055FB909B80DF97	READ WRITE	NO
4	DORIA	9F3AC1DE46E50A89E0554D86E6126571	READ WRITE	NO
5	ANDREA	9F3AC1DE46ED0A89E0554D86E6126571	MOUNTED	

- Keine Informationen über „*UNPLUGGED*“ PDBs

- show pdbs

```
SQL> show pdbs
```

CON_ID	CON_NAME	OPEN	MODE	RESTRICTED
2	PDB\$SEED	READ ONLY		NO
3	CELLO	READ WRITE		NO
4	DORIA	READ WRITE		NO
5	ANDREA	MOUNTED		

- Keine Informationen über „*UNPLUGGED*“ PDBs

- Unplug nur möglich, wenn die PDB einmal geöffnet (READ ONLY, READ WRITE) war

```
SQL> ALTER PLUGGABLE DATABASE andrea UNPLUG INTO '/u02/oradata/andrea.xml';  
ALTER PLUGGABLE DATABASE andrea UNPLUG INTO '/u02/oradata/andrea.xml'  
*  
ERROR at line 1:  
ORA-65085: cannot open pluggable database in read only mode
```

- Informationen über die „*unplugged*“ PDB bleibt im Controlfile vorhanden

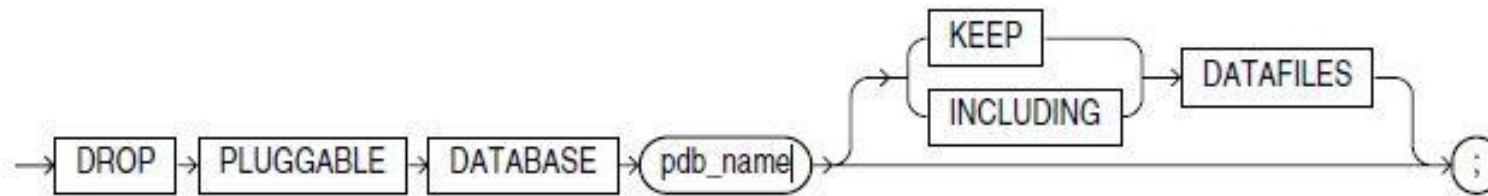
```
SELECT p.pdb_name, t.name, p.status, p.pdb_id
FROM v$tablespace t, cdb_pdb$seeds p
WHERE t.con_id = p.pdb_id ORDER BY 4;
```

PDB_NAME	NAME	STATUS	PDB_ID
-----	-----	-----	-----
PDB\$SEED	TEMP	NORMAL	2
PDB\$SEED	UNDOTBS1	NORMAL	2
...			
DORIA	TEMP	NORMAL	4
DORIA	USERS	NORMAL	4
ANDREA	SYSTEM	UNPLUGGED	6
ANDREA	SYSAUX	UNPLUGGED	6
ANDREA	UNDOTBS1	UNPLUGGED	6
ANDREA	TEMP	UNPLUGGED	6
ANDREA	USERS	UNPLUGGED	6

Drop Pluggable Database

- Nach dem „UNPLUG“ nur noch ein „DROP“ möglich

drop_pluggable_database::=



- Default: KEEP DATAFILES

```
SQL> DROP PLUGGABLE DATABASE andrea;
```

- Löscht nur den Eintrag der Pluggable Database im Data Dictionary
- Datafiles bleiben bestehen

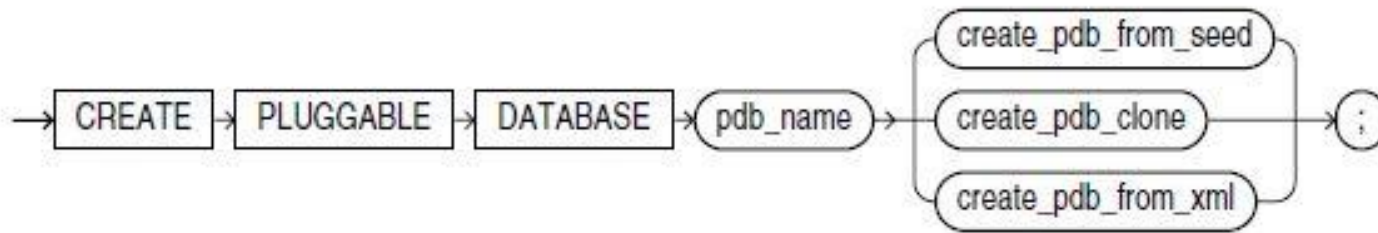
DROP PLUGGABLE DATABASE

- Datafiles werden nicht mehr mit gesichert
- Nach dem DROP PLUGGABLE DATABASE kann die PDB wieder „eingehängt“ werden

... und was ist mit „PLUGIN“?

- ALTER PLUGGABLE DATABASE PLUG ... gibt es nicht
- Statt dessen: CREATE PLUGGABLE DATABASE

create_pluggable_database::=



```
SQL> CREATE PLUGGABLE DATABASE andrea USING '/u02/oradata/andrea.xml';
```

- VORSICHT: Damit werden alle Datafiles kopiert

```
SQL> CREATE PLUGGABLE DATABASE andrea USING '/u02/oradata/andrea.xml' NOCOPY;
```

Neuerstellung mit pdb-File

```
SQL> ALTER PLUGGABLE DATABASE cello UNPLUG INTO '/u02/oradata/UDO/cello.pdb';
```

```
SQL> DROP PLUGGABLE DATABASE cello INCLUDING DATAFILES;
```

```
SQL> CREATE PLUGGABLE DATABASE rudi USING '/u02/oradata/rudi.pdb';
```

Nach Recreate

- Alte Backups der PDB sind nicht mehr nutzbar!
- Status der PDB ist nicht mehr gesetzt (SAVE STATE)
- Aber: Parameter der PDB bleiben gesetzt

- Bei OMF wird die GUID wiederverwendet
- Problem, wenn man die Option „NOCOPY“ vergessen hat

```
SQL> CREATE PLUGGABLE DATABASE andrea USING '/u02/oradata/andrea.xml';
```

```
% ls -l /u02/oradata/UDO_S1/9F3AC1DE46ED0A89E0554D86E6126571/datafile/  
total 1884280
```

```
-rw-r----- 1 oracle oinstall 262152192 Feb 23 12:37 o1_mf_sysaux_h54o19pb_.dbf  
-rw-r----- 1 oracle oinstall 262152192 Feb 23 12:44 o1_mf_sysaux_h54sdvkv_.dbf  
-rw-r----- 1 oracle oinstall 325066752 Feb 23 12:37 o1_mf_system_h54o19p3_.dbf  
-rw-r----- 1 oracle oinstall 325066752 Feb 23 12:44 o1_mf_system_h54sdvkm_.dbf  
-rw-r----- 1 oracle oinstall 104865792 Feb 23 12:44 o1_mf_temp_h54sdvky_.dbf  
-rw-r----- 1 oracle oinstall 272637952 Feb 23 12:37 o1_mf_undotbs1_h54o19pc_.dbf  
-rw-r----- 1 oracle oinstall 272637952 Feb 23 12:44 o1_mf_undotbs1_h54sdvkv_.dbf  
-rw-r----- 1 oracle oinstall 104865792 Feb 23 12:37 o1_mf_users_h54o19pd_.dbf  
-rw-r----- 1 oracle oinstall 104865792 Feb 23 12:44 o1_mf_users_h54sdvky_.dbf
```

- Indirektes Anmelden über die Container Datenbank:

```
$ sqlplus / as sysdba          # CDB
SQL> ALTER SESSION SET CONTAINER= ...;
```

- „Normale“ Anmeldung an die PDB nur über entsprechenden Servicenamen

```
$ sqlplus username/password@TNS-Alias
```

- Auf welcher PDB / CDB bin ich?

```
SQL> show con_name
```

```
CON_NAME
```

```
-----
```

```
CDB$ROOT
```

Skripte

- Beispiel:

```
SELECT tablespace_name,  
       trunc(sum(bytes)/1024/1024) Mbytes,  
       trunc(sum(decode(maxbytes,0,bytes,maxbytes)/1024/1024)) maxmbytes  
FROM dba_data_files  
GROUP BY tablespace_name;
```

TABLESPACE_NAME	MBYTES	MAXMBYTES
-----	-----	-----
SYSAUX	870	32767
UNDOTBS1	920	32767
USERS	5	32767
SYSTEM	800	32767

- Ist das richtig? – Es kommt darauf an!
- Es handelt sich hier wahrscheinlich um eine CDB, weil der UNDO-Tablespace angezeigt wird

- Ist das hier besser?

```
SELECT tablespace_name,  
       trunc(sum(bytes)/1024/1024) Mbytes,  
       trunc(sum(decode(maxbytes,0,bytes,maxbytes)/1024/1024)) maxmbytes  
FROM   cdb_data_files  
GROUP BY tablespace_name;
```

TABLESPACE_NAME	MBYTES	MAXMBYTES
USERS	405	38911
UNDOTBS1	920	32767
SYSTEM	1320	98303
SYSAUX	2070	98303

- Die MBYTES scheinen okay aber MAXMBYTES sehen merkwürdig aus!

Vielleicht passt das hier?

```
SELECT con_id, tablespace_name,  
       trunc(sum(bytes)/1024/1024) Mbytes,  
       trunc(sum(decode(maxbytes,0,bytes,maxbytes)/1024/1024)) maxmbytes  
FROM   cdb_data_files  
GROUP  BY con_id, tablespace_name  
ORDER  BY con_id;
```

CON_ID	TABLESPACE_NAME	MBYTES	MAXMBYTES
1	USERS	5	32767
1	UNDOTBS1	920	32767
1	SYSTEM	800	32767
1	SYSAUX	870	32767
3	USERS	100	5120
3	SYSAUX	600	32767
3	SYSTEM	260	32767
4	USERS	300	1024
4	SYSTEM	260	32767
4	SYSAUX	600	32767

Example 19-7 Showing the Data Files for Each PDB in a CDB

This example queries the `DBA_PDBS` and `CDB_DATA_FILES` views to show the name and location of each data file for all of the PDBs in a CDB, including the PDB seed.

```
COLUMN PDB_ID FORMAT 999
COLUMN PDB_NAME FORMAT A8
COLUMN FILE_ID FORMAT 9999
COLUMN TABLESPACE_NAME FORMAT A10
COLUMN FILE_NAME FORMAT A45

SELECT p.PDB_ID, p.PDB_NAME, d.FILE_ID, d.TABLESPACE_NAME, d.FILE_NAME
FROM DBA_PDBS p, CDB_DATA_FILES d
WHERE p.PDB_ID = d.CON_ID
ORDER BY p.PDB_ID;
```

Sample output:

PDB_ID	PDB_NAME	FILE_ID	TABLESPACE_NAME	FILE_NAME
2	PDB\$SEED	6	SYSAUX	/disk1/oracle/dbs/pdbseed/cdbl_ax.f
2	PDB\$SEED	5	SYSTEM	/disk1/oracle/dbs/pdbseed/cdbl_db.f
3	HRPDB	9	SYSAUX	/disk1/oracle/dbs/hrpdb/hrpdb_ax.f
3	HRPDB	8	SYSTEM	/disk1/oracle/dbs/hrpdb/hrpdb_db.f
3	HRPDB	13	USER	/disk1/oracle/dbs/hrpdb/hrpdb_usr.dbf
4	SALESPDB	15	SYSTEM	/disk1/oracle/dbs/salespdb/salespdb_db.f
4	SALESPDB	16	SYSAUX	/disk1/oracle/dbs/salespdb/salespdb_ax.f
4	SALESPDB	18	USER	/disk1/oracle/dbs/salespdb/salespdb_usr.dbf

Oracle Database Oracle® Multitenant Administrator's Guide 19c E96136-04 June 2019

- Default: PDB\$SEED wird bei allen CDB-Views ignoriert
- Laut Oracle muss das so sein (warum auch immer)
- Ausschalten mit:
 - 12.1: `exclude_seed_cdb_view = FALSE`
 - 12.2: `_exclude_seed_cdb_view = FALSE`
- Setting this parameter to FALSE would return results for the seed database when querying against the CDB views

```
ALTER SYSTEM SET "_exclude_seed_cdb_view"=FALSE;
SELECT con_id, tablespace_name,
       trunc(sum(bytes)/1024/1024) Mbytes,
       trunc(sum(decode(maxbytes,0,bytes,maxbytes)/1024/1024)) maxmbytes
FROM   cdb_data_files
GROUP  BY con_id, tablespace_name
ORDER  BY con_id;
```

CON_ID	TABLESPACE_NAME	MBYTES	MAXMBYTES
1	USERS	5	32767
1	UNDOTBS1	920	32767
1	SYSTEM	800	32767
1	SYSAUX	870	32767
2	SYSTEM	250	32767
2	SYSAUX	550	32767
3	USERS	100	5120
3	SYSAUX	600	32767
3	SYSTEM	260	32767
4	USERS	300	1024
4	SYSTEM	260	32767
4	SYSAUX	600	32767

Verzeichnisse

- Verzeichnisstruktur vorgeben:

- DB_CREATE_FILE_DEST → OMF Struktur für ALLE PDBs (spfile Parameter)
- CREATE_FILE_DEST → OMF Struktur für PDB
- FILE_NAME_CONVERT → Ändern des Standardpfades der Dateien
- PATH_PREFIX → Festlegen des Pfades für Directory Objekte
- PDB_FILE_NAME_CONVERT → Ändern von bestimmten Teilen der Verzeichnisstruktur

- Nicht Oracle Managed Files (OMF)
 - Ändert Teile des Pfades
 - Nur bei CREATE PLUGGABLE DATABASE

```
SQL> CREATE PLUGGABLE DATABASE cello  
      ADMIN USER pdbadmin IDENTIFIED BY manager  
      file_name_convert=('/u02/oradata/UDO/pdbseed', '/u03/oradata/pdbshared/UDO/cello');
```

- z.B. bei eigenem Filesystem für die PDB

- Nur Oracle Managed Files
- Ändern des Basisverzeichnisses für eine PDB (db_create_file_dest für PDB)

```
SQL> CREATE PLUGGABLE DATABASE doria
      ADMIN USER pdbadmin IDENTIFIED BY manager
      create_file_dest='/u03/oradata';
```

```
SQL> SELECT con_id, tablespace_name, file_name FROM cdb_data_files ORDER BY 1,2;
```

CON_ID	TABLESPACE_NAME	FILE_NAME
1	SYSAUX	/u02/oradata/UDO_S1/datafile/o1_mf_sysaux_h522h356_.dbf
...		
2	USERS	/u02/oradata/UDO_S1/9F290C149A1D0795E055FB909B80DF97/datafile/o1_mf_users_h522h9x4_.dbf
3	SYSAUX	/u02/oradata/UDO_S1/9F2C4A9C6B704912E055FB909B80DF97/datafile/o1_mf_sysaux_h52j25f3_.dbf
3	SYSTEM	/u02/oradata/UDO_S1/9F2C4A9C6B704912E055FB909B80DF97/datafile/o1_mf_system_h52j25dc_.dbf
3	UNDOTBS1	/u02/oradata/UDO_S1/9F2C4A9C6B704912E055FB909B80DF97/datafile/o1_mf_undotbs1_h52j25f4_.dbf
3	USERS	/u02/oradata/UDO_S1/9F2C4A9C6B704912E055FB909B80DF97/datafile/o1_mf_users_h52j25f5_.dbf
4	SYSAUX	/u03/oradata /UDO_S1/9F3AC1DE46E50A89E0554D86E6126571/datafile/o1_mf_sysaux_h54n42cg_.dbf
4	SYSTEM	/u03/oradata/UDO_S1/9F3AC1DE46E50A89E0554D86E6126571/datafile/o1_mf_system_h54n42c4_.dbf
4	UNDOTBS1	/u03/oradata/UDO_S1/9F3AC1DE46E50A89E0554D86E6126571/datafile/o1_mf_undotbs1_h54n42cj_.dbf
4	USERS	/u03/oradata/UDO_S1/9F3AC1DE46E50A89E0554D86E6126571/datafile/o1_mf_users_h54n42ck_.dbf

- Nur bei NON-OMF
 - Directories können nur in dem angegebenen Verzeichnis erstellt werden
 - Hat keine Auswirkung auf Datafiles oder Tempfiles
 - Gilt nicht für „Oracle Maintained“ Directories
 - Kann nachträglich nicht geändert werden

```
SQL> CREATE PLUGGABLE DATABASE cello
      ADMIN USER pdbadmin IDENTIFIED BY manager
      file_name_convert=('/u02/oradata/UDO/pdbseed','/u03/oradata/pdbshared/UDO/cello')
      path_prefix='/u04/directories/cello';
```

```
SQL> SELECT * FROM database_properties WHERE property_name='PATH_PREFIX'
```

PROPERTY_NAME	PROPERTY_VALUE	DESCRIPTION
PATH_PREFIX	/u04/directories/cello	All paths for objects such as directories are relative to this

- Oracle Managed Files
 - Kein db_unique_name

```
SQL> CREATE PLUGGABLE DATABASE andrea  
      ADMIN USER pdbadmin IDENTIFIED BY manager  
      path_prefix='/u04/directories/andrea';
```

```
SQL> SELECT * FROM database_properties WHERE property_name='PATH_PREFIX'
```

PROPERTY_NAME	PROPERTY_VALUE	DESCRIPTION
PATH_PREFIX	/u04/directories/andrea	All paths for objects such as directories are relative to this

- Nicht Oracle Managed Files
- Geeignet für Cloning von PDBs

```
SQL> ALTER SESSION SET
      pdb_file_name_convert='/u02/oradata/UDO/pdbseed','/u03/oradata/UDO/rudi';

SQL> CREATE PLUGGABLE DATABASE rudi
      ADMIN USER pdbadmin IDENTIFIED BY manager;

SQL> SELECT con_id, tablespace_name, file_name FROM cdb_data_files ORDER BY 1,2;
```

CON_ID	TABLESPACE_NAME	FILE_NAME
...		
2	SYSAUX	/u02/oradata/UDO/pdbseed/sysaux01.dbf
2	SYSTEM	/u02/oradata/UDO/pdbseed/system01.dbf
...		
5	SYSAUX	/u03/oradata/UDO/rudi/sysaux01.dbf
5	SYSTEM	/u03/oradata/UDO/rudi/system01.dbf
5	UNDOTBS1	/u03/oradata/UDO/rudi/undotbs01.dbf
5	USERS	/u03/oradata/UDO/rudi/users01.dbf

Oracle Managed Files

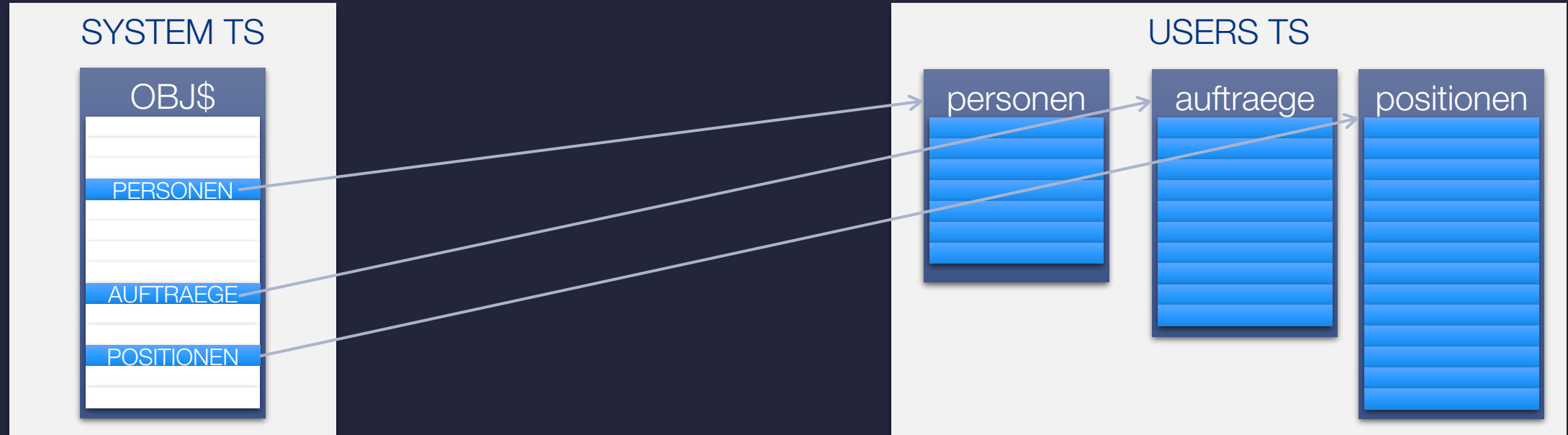
- Vorteile:
 - Sehr einfache Syntax
 - Wird auf der Standby Seite ebenfalls angelegt
- Nachteile:
 - Trennung von CDB und PDB schwierig (DB_UNIQUE_NAME immer im Pfad)
 - „Leichen“ beim Erstellen von PDBs von Manifest (NOCOPY)

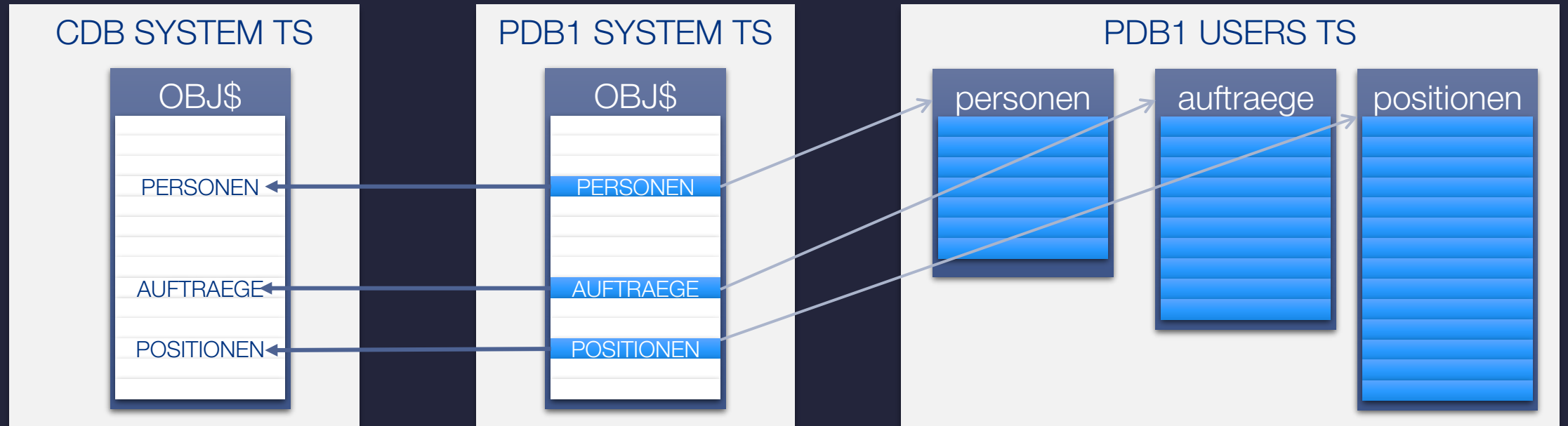
NON Oracle Managed Files

- Vorteile:
 - Trennung von CDB und PDB
 - Einfache Dateinamen
- Nachteile:
 - Aufwändigere Syntax
 - Filesystem wird auf der Standby Seite nicht angelegt

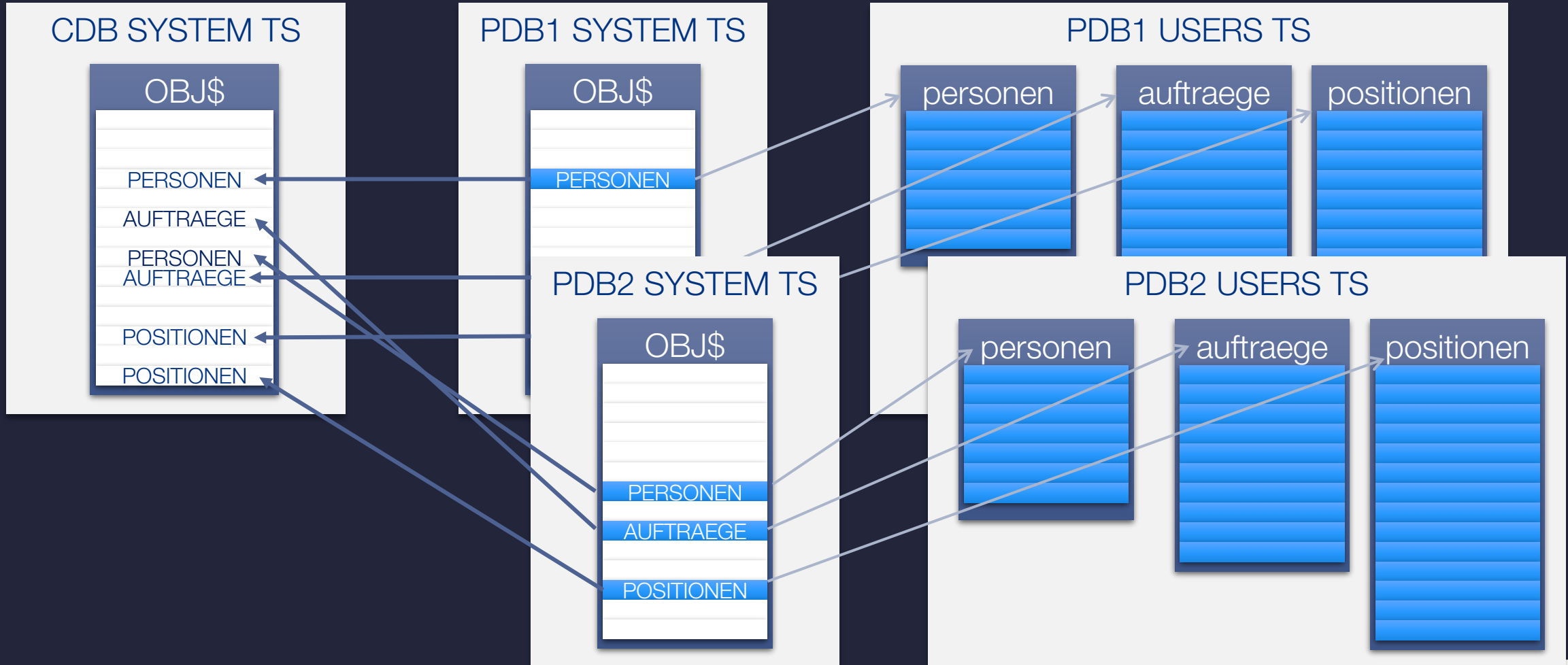
Data Dictionary

- CDB_<view> → Sicht auf alle Container-Daten aus der CDB genau wie DBA_<view> in der PDB
CON_ID wird mit angezeigt
- DBA_<view> → Sicht auf die lokalen Objekte (egal ob CDB oder PDB) als DBA
- ALL_<view> → Sicht auf alle Objekte, auf die man zugreifen darf (nur lokal in der PDB bzw. CDB)
- USER_<view> → Sicht des Benutzers auf seine eigenen Objekte





PDB Data Dictionary



- Data Dictionary Metadaten liegen in der CDB\$ROOT
 - Z.B. TAB\$
- PDBs haben einen Link in diese Root-Tabellen (Metadata Link)
- Data Dictionary Daten können sowohl im Root als auch in der PDB liegen
 - COMMON USER → Root
 - Local Users → PDB
 - PDB bekommt die Root-Daten vererbt (Read Only) und hat seine eigenen Daten

OWNER	OBJECT_NAME	OBJECT_ID	OBJECT_TYPE	SHARING	CON_ID
SYS	TAB\$	4	TABLE	METADATA LINK	1
SYS	WRM\$_DATABASE_INSTANCE	7778	TABLE	METADATA LINK	1
SYS	DBA_HIST_DATABASE_INSTANCE	7799	VIEW	OBJECT LINK	1
SYS	TAB\$	4	TABLE	METADATA LINK	2
SYS	WRM\$_DATABASE_INSTANCE	7776	TABLE	METADATA LINK	2
SYS	DBA_HIST_DATABASE_INSTANCE	7797	VIEW	OBJECT LINK	2
SYS	TAB\$	4	TABLE	METADATA LINK	3
SYS	WRM\$_DATABASE_INSTANCE	7776	TABLE	METADATA LINK	3
SYS	DBA_HIST_DATABASE_INSTANCE	7797	VIEW	OBJECT LINK	3
DEMO	PERSONEN	19938	TABLE	NONE	3
SYS	TAB\$	4	TABLE	METADATA LINK	4
SYS	WRM\$_DATABASE_INSTANCE	7776	TABLE	METADATA LINK	4
SYS	DBA_HIST_DATABASE_INSTANCE	7797	VIEW	OBJECT LINK	4
DEMO	PERSONEN	19881	TABLE	NONE	4
SYS	TAB\$	4	TABLE	METADATA LINK	5
SYS	WRM\$_DATABASE_INSTANCE	7776	TABLE	METADATA LINK	5
SYS	DBA_HIST_DATABASE_INSTANCE	7797	VIEW	OBJECT LINK	5
PUBLIC	DBA_HIST_DATABASE_INSTANCE	7798	SYNONYM	METADATA LINK	5
DEMO	PERSONEN	19938	TABLE	NONE	5

- Abfrage 1

```
SELECT obj#, owner#, name FROM obj$ WHERE name = 'PERSONEN';
```

- CDB

```
no rows selected
```

- PDB1

OBJ#	OWNER#	NAME
19938	69	PERSONEN

- PDB2

OBJ#	OWNER#	NAME
19881	66	PERSONEN

- Abfrage 2

```
SELECT username, user_id, common FROM dba_users WHERE username IN ('DEMO','SYSTEM');
```

- CDB

USERNAME	USER_ID	COMMON
SYSTEM	8	YES

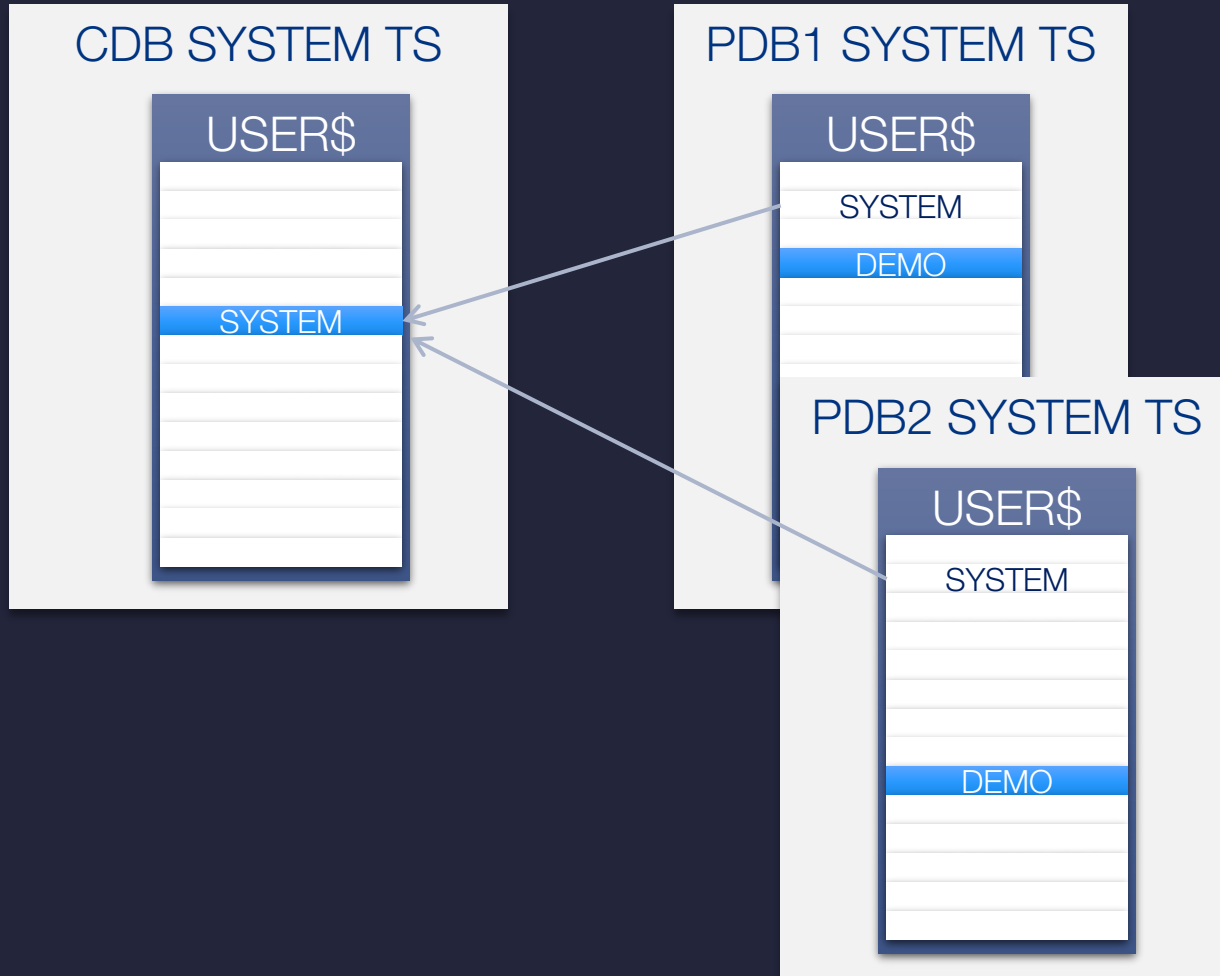
- PDB1

USERNAME	USER_ID	COMMON
SYSTEM	8	YES
DEMO	69	NO

- PDB2

USERNAME	USER_ID	COMMON
SYSTEM	8	YES
DEMO	69	NO

„COMMON“ Users



- Abfrage 3

```
SELECT username, user_id, common FROM cdb_users WHERE username IN ('DEMO','SYSTEM');
```

- CDB

USERNAME	USER_ID	COMMON
-----	-----	-----
SYSTEM	8	YES
DEMO	66	NO
SYSTEM	8	YES
SYSTEM	8	YES
DEMO	69	NO
SYSTEM	8	YES
SYSTEM	8	YES
DEMO	69	NO

- Abfrage 4

```
SELECT owner, object_name, object_id, con_id FROM cdb_objects  
WHERE object_name = 'PERSONEN';
```

- CDB

OWNER	OBJECT_NAME	OBJECT_ID	CON_ID
DEMO	PERSONEN	91897	3
DEMO	PERSONEN	92039	4

- PDB1

OWNER	OBJECT_NAME	OBJECT_ID	CON_ID
DEMO	PERSONEN	91897	3

- PDB2

OWNER	OBJECT_NAME	OBJECT_ID	CON_ID
DEMO	PERSONEN	92039	3

- SELECT COUNT (*) FROM ...

Tabelle	CDB	PLUGJA21	PLUGJA22	PJA12CD1	PDB\$SEED
USER\$	121	123	122	122	121
TAB\$	2362	2364	2363	2363	2362
IND\$	4240	4244	4244	4244	4240
OBJ\$	91372	90828	90803	91157	90773
COL\$	106677	106694	106692	106692	106678
CDB_USERS	179	37	36	36	35
CDB_TABLES	11579	2317	2316	2316	2315
CDB_INDEXES	21212	4244	4244	4244	4240
CDB_OBJECTS	454953	90874	90801	91155	90693
CDB_TAB_COLUMNS	521624	104332	104330	104330	104316

```
$ sqlplus / as sysdba -- Connect an CDB
```

```
SQL> SELECT username, user_id, common, oracle_maintained  
       FROM dba_users;
```

USERNAME	USER_ID	COM	O
-----	-----	---	-
SYS	0	YES	Y
SYSTEM	9	YES	Y
XS\$NULL	2147483638	YES	Y
SYSBACKUP	2147483617	YES	Y
SYSRAC	2147483620	YES	Y
SYSKM	2147483619	YES	Y
OUTLN	13	YES	Y
SYS\$UMF	50	YES	Y
SYSDG	2147483618	YES	Y
DBSNMP	58	YES	Y
...			
DIP	25	YES	Y
ORACLE_OCM	41	YES	Y

Anzeige „NON-COMMON“ Users

```
$ sqlplus / as sysdba -- Connect an CDB
```

```
SQL> SELECT username, user_id, common  
       FROM dba_users WHERE common = 'NO';
```

```
no rows selected
```

```
SQL> ALTER SESSION SET CONTAINER = cello;
```

```
SQL> SELECT username, user_id, common  
       FROM dba_users WHERE common = 'NO';
```

USERNAME	USER_ID	COM
DEMO	103	NO
PDBADMIN	102	NO
JOHANNES	104	NO

```
SQL> ALTER SESSION SET CONTAINER = andrea;
```

```
SQL> SELECT username, user_id, common  
       FROM dba_users WHERE common = 'NO';
```

USERNAME	USER_ID	COM
DEMO	103	NO
PDBADMIN	102	NO

- Profiles werden auf CDB-Ebene gesetzt
- C## als Prefix

```
SQL> CREATE PROFILE c##techuser LIMIT  
      FAILED_LOGIN_ATTEMPTS 5  
      PASSWORD_LIFE_TIME 60  
      PASSWORD_REUSE_TIME 60  
      PASSWORD_REUSE_MAX 5  
      PASSWORD_VERIFY_FUNCTION ora12c_verify_function  
      PASSWORD_LOCK_TIME 1/24  
      PASSWORD_GRACE_TIME 10  
      INACTIVE_ACCOUNT_TIME 30;
```

- Rolle wird in der CDB definiert und ist automatisch in allen PDBs verfügbar
- Privilegien in den PDBs können unterschiedlich sein
- Voraussetzung: Role beginnt mit "C##" (Serverparameter)

```
SQL> CREATE ROLE c##dba CONTAINER=ALL;
```

```
Role created.
```

```
SQL> GRANT CREATE SESSION, ALTER SESSION TO c##dba CONTAINER=ALL;
```

```
Grant succeeded.
```

- Voraussetzung wie bei der Role

```
SQL> CREATE USER c##dbauser IDENTIFIED BY "oracle";
```

```
SQL> GRANT c##dba TO c##dbauser;
```

```
SQL> CONNECT c##dbauser/oracle
```

```
SQL> ALTER SESSION SET CONTAINER=doria;
```

```
ERROR:
```

```
ORA-01031: insufficient privileges
```

- Privileg SET CONTAINER

```
SQL> GRANT SET CONTAINER TO c##dbauser CONTAINER=ALL;
```

```
SQL> ALTER SESSION SET CONTAINER=cello;  
Session altered.
```

- ALTER SESSION Privileg reicht nicht aus
- Privileg kann nur an User vergeben werden!
- Hier muss die Option „CONTAINER=...“ angegeben werden!

Default Tablespace

- Jeder User hat
 - einen Default Tablespace
 - einen Default Temporary Tablespace
- Common User:
 - Die Tablespaces müssen in allen (!) PDBs existieren

- Schemaverwaltung in der PDB genau wie bei NON-CDB Datenbank
- Globalen Usern können in der PDB weitere Rechte gegeben werden

```
SQL> CONNECT / AS SYSDBA
Connected.
SQL> ALTER SESSION SET CONTAINER=cello;
Session altered.
SQL> GRANT CREATE SESSION TO c##dbauser; -- User
Grant succeeded.
Oder:
SQL> grant create session to c##dba;      -- Role
Grant succeeded.

Aber nicht:
SQL> GRANT CREATE SESSION TO c##dba CONTAINER=ALL;
GRANT CREATE SESSION TO c##dba CONTAINER=ALL
*
ERROR at line 1:
ORA-65050: Common DDLs only allowed in CDB$ROOT
```

- Vergabe von Objektprivilegen an Common User ist nicht möglich

```
SQL> GRANT SELECT ON demo.personen TO c##dbauser CONTAINER=ALL;  
ERROR at line 1:  
ORA-00942: table or view does not exist
```

- Gründe:
 - Objekt müsste in ALLEN PDBs und in der CDB existieren
 - In der CDB kann aber ein User nicht ohne Prefix „c##“ angelegt werden
 - Selbst wenn das Objekt überall existieren würden (z.B. SYNONYM) → ORA-65033 (a common privilege may not be granted or revoked on a local object)

- Tabellen und Views als „CONTAINER_DATA“ gekennzeichnet
- Zugriff nur mit entsprechenden Privilegien

```
SQL> SELECT view_name FROM cdb_views WHERE container_data='Y';
```

```
VIEW_NAME
```

```
-----
```

```
...
```

```
DBA_PDBS
```

```
...
```

```
GV_$PDBS
```

```
...
```

- Gilt auch für Tabellen (cdb_tables) dort gibt es aber standardmäßig keine

```
SQL> SELECT table_name FROM cdb_tables WHERE container_data='YES';
```

```
no rows selected
```

```
SQL(SYS)> CREATE USER c##developer IDENTIFIED BY manager;
SQL> GRANT create Session TO c##developer;
SQL> connect c##developer/manager
SQL> SELECT * from v$pdb;
SELECT * from v$pdb
      *
ERROR at line 1:
ORA-00942: table or view does not exist

SQL(SYS)> GRANT SELECT ON v_$pdb TO c##developer;

SQL> SELECT name from v$pdb;
no rows selected

SQL (SYS)> ALTER USER c##developer SET CONTAINER_DATA=ALL CONTAINER=CURRENT;

SQL> SELECT name from v$pdb;

NAME
-----
PDB$SEED
CELLO
DORIA
ANDREA
```

- Alle CDB-Views

```
SQL> ALTER USER c##dbaadmin SET CONTAINER_DATA=ALL CONTAINER=CURRENT;
```

- Bestimmte Views

```
SQL> ALTER USER c##developer SET CONTAINER_DATA=ALL  
FOR sys.v_$pdbs CONTAINER=CURRENT;
```

- CONTAINER=CURRENT oder CONTAINER=<CDB>,<PDB>

```
SQL> ALTER USER c##developer SET CONTAINER_DATA=CDB$ROOT  
FOR sys.v_$pdbs CONTAINER=CURRENT;
```

- Keine CDB-Views / Zurücknahme

```
SQL> ALTER USER c##dbaadmin SET CONTAINER_DATA=DEFAULT CONTAINER=CURRENT;
```

- Container „wegnehmen“

```
SQL> ALTER USER c##developer SET CONTAINER_DATA=ALL  
FOR v_$pdbs CONTAINER=CURRENT;
```

- CONTAINER=CURRENT oder CONTAINER=<CDB>,<PDB>

```
SQL> ALTER USER c##developer SET CONTAINER_DATA=CDB$ROOT  
FOR v_$pdbs CONTAINER=CURRENT;
```

- SYS
 - Ist in allen Pluggable Databases vorhanden und darf alles
- SYSTEM
 - Ebenfalls in allen Pluggable Databases vorhanden
 - Darf Pluggable Databases erstellen
 - Kann keine Pluggable Database öffnen oder schließen

- Syntax

```
SELECT ...  
  FROM CONTAINERS (<tables_name>)  
WHERE CON_ID = ...  
  AND ...
```


- Objekt muss für Common User existieren
 - Ev. als View (schemaname.tablename)
- Objekt muss auch in der CDB existieren

- CDB:

```
SQL> CREATE ROLE c##resource CONTAINER=ALL;
```

```
SQL> GRANT create session, alter session, create table, create view TO c##resource;
```

```
SQL> CREATE user c##demo IDENTIFIED BY demo CONTAINER=ALL;
```

```
SQL> GRANT c##resource TO c##demo CONTAINER=ALL;
```

```
SQL> ALTER USER c##demo QUOTA UNLIMITED ON users;
```

```
SQL> CREATE TABLE c##demo.personen(  
    persid      NUMBER(10),  
    anrede      VARCHAR2(5 CHAR),  
    vorname     VARCHAR2(50 CHAR),  
    nachname    VARCHAR2(50 CHAR));
```

```
SQL> INSERT ...
```

- PDB cello (SYS):

```
SQL> CREATE user demo IDENTIFIED BY demo;
```

```
SQL> GRANT c##resource TO demo;
```

```
SQL> ALTER USER demo QUOTA UNLIMITED ON users;
```

```
SQL> CREATE TABLE demo.personen (  
    persid      NUMBER(10),  
    anrede      VARCHAR2(5 CHAR),  
    vorname     VARCHAR2(50 CHAR),  
    nachname    VARCHAR2(50 CHAR));
```

```
SQL> INSERT ...
```

```
SQL> GRANT SELECT ON demo.personen TO c##demo;
```

- Synonyme:

```
SQL> CONNECT c##demo/demo@localhost:1521/CELLO

SQL> CREATE SYNONYM personen FOR demo.personen;
```

- Abfrage aus CDB\$ROOT:

```
SQL> CONNECT c##demo/demo

SQL> SELECT vorname, nachname, con_id FROM containers(personen);
```

VORNAME	NACHNAME	CON_ID
-----	-----	-----
Gustav	Meier	1
Grete	Schmitz	1
Johannes	Müller	3
Susanne	Peters	3

- Abfrage der Container ID

```
SQL> CONNECT c##demo/demo
```

```
SQL> SELECT vorname, nachname, con_id FROM containers(personen)  
        WHERE con_id = 3;
```

VORNAME	NACHNAME	CON_ID
-----	-----	-----
Johannes	Müller	3
Susanne	Peters	3

- Abfrage der Container ID, wenn die Tabelle dort nicht existiert:

```
SQL> connect demo/demo@localhost:1521/DORIA;
```

```
SQL> select * from personen;
```

```
select * from personen
          *
```

```
ERROR at line 1:
```

```
ORA-00942: table or view does not exist
```

```
SQL> connect c##demo/manager
```

```
SQL> SELECT vorname, nachname, con_id FROM containers(personen) where con_id=4;
```

```
no rows selected
```

- No Rows Selected statt ORA-00942

Wofür braucht man das?

- Auswertung / Vergleich von Performancedaten
- z.B. Statspack installiert in jeder PDB
- Neuer User `c##perfstat` mit Zugriff auf alle Statspack Schemata?

Service Erstellung für die PDB (1)

- Jede PDB hat automatisch einen eigenen Service, der aber nicht geändert werden kann
- Konfigurieren von Services mit `srvctl` (GI Installation):
 - Spezieller Parameter `-pdb` zum Anlegen bzw. Ändern von Services
 - Geänderte Optionen (z.B. `-db` statt `-d`, `-service` statt `-s`)

```
$ srvctl add service -db udo_s1 -pdb cello -service cello_p  
  
$ srvctl start service -db udo_s1 -service cello_p
```

- Die PDB wird automatisch geöffnet, wenn der Service gestartet wird

- Ohne srvctl (d.h. keine GI Installation):
 - dbms_service Package
 - Muss in der entsprechenden PDB ausgeführt werden!

```
SQL> ALTER SESSION SET CONTAINER = cello;
```

Session wurde geändert.

```
SQL> EXECUTE dbms_service.create_service('CELLO_P', 'CELLO_P.DB.DE');
```

PL/SQL-Prozedur erfolgreich abgeschlossen.

```
SQL> EXECUTE dbms_service.start_service('CELLO_P');
```

PL/SQL-Prozedur erfolgreich abgeschlossen.

```
SQL> ALTER PLUGGABLE DATABASE SAVE STATE;
```

- Services „gehören“ zur PDB
- Werden automatisch gestartet
- Werden beim Cloning mitgenommen
- Services können dadurch doppelt existieren
 - gleiche CDB → kein Problem, zweiter Service kann nicht gestartet werden
 - andere CDB, gleicher Server → Services werden mehrfach gestartet, Gefahr des Split Brain!

- `v$pdb`s → Liste alle PDBs ohne CDB aus dem Controlfile (enthält PDB_ID)
- `v$containers` → Liste aller Container inklusive CDB
- `cdb_pdb`s → Liste aller PDBs aus dem Data Dictionary (mounted PDBs werden nicht angezeigt)
 - Es gibt (scheinbar) keine View aus dem DD, dass alle PDBs und die CDB anzeigt
- `sys_context('USERENV', 'CON_NAME')`
- `sys_context('USERENV', 'CON_ID')`

sys_context → DB_NAME

```
SQL> select sys_context('USERENV','DB_NAME') from dual;
```

```
SYS_CONTEXT('USERENV','DB_NAME')
```

```
-----  
UDO
```

```
SQL> ALTER session set container=cello;
```

```
SQL> SELECT sys_context('USERENV','DB_NAME') from dual;
```

```
SYS_CONTEXT('USERENV','DB_NAME')
```

```
-----  
CELLO
```

```
SQL> SELECT sys_context('USERENV','DB_UNIQUE_NAME') from dual;
```

```
SYS_CONTEXT('USERENV','DB_UNIQUE_NAME')
```

```
-----  
UDO_S1
```

PDB Parameter

- Oracle 12.1 nur sehr eingeschränkt
- Oracle 12.2 für SGA Parameter und Lockdown Profiles

- Einige ALTER SYSTEM Befehle beziehen sich auf die aktuelle PDB:

```
SQL> ALTER SYSTEM FLUSH SHARED_POOL  
SQL> ALTER SYSTEM FLUSH BUFFER_CACHE  
SQL> ALTER SYSTEM CHECKPOINT
```

- z.B. Buffer Cache

```
SQL> select con_id, count(*) from v$bh  
       group by con_id;
```

CON_ID	COUNT(*)
1	12978
2	1000
4	1792
3	12873

v\$parameter mit ispdb_modifiable

```
SQL> SELECT name, value FROM v$parameter WHERE ispdb_modifiable = 'TRUE' ORDER BY name;
```

NAME	VALUE
-----	-----
_exclude_seed_cdb_view	FALSE
adg_account_info_tracking	LOCAL
allow_rowid_column_type	FALSE
approx_for_aggregation	FALSE
approx_for_count_distinct	FALSE
approx_for_percentile	none
aq_tm_processes	1
asm_diskstring	
awr_pdb_autoflush_enabled	FALSE
bitmap_merge_area_size	1048576
blank_trimming	FALSE
cell_offload_compaction	ADAPTIVE
...	
cpu_count	1
cpu_min_count	1
db_create_file_dest	/u02/oradata
db_create_online_log_dest_1	
...	

- In der CDB für alle Container:

```
SQL> ALTER SYSTEM SET optimizer_index_cost_adj=10 CONTAINER=ALL;
```

- Für Root Container (CDB\$ROOT):

```
SQL> ALTER SYSTEM SET optimizer_index_cost_adj=20 CONTAINER=CURRENT;
```

- In der PDB:

```
SQL> ALTER SESSION SET CONTAINER=cello;
```

```
SQL> ALTER SYSTEM SET optimizer_index_cost_adj=50 CONTAINER=CURRENT;
```

- Parameter der PDB werden NICHT im SPFILE gespeichert
- Anzeige in der PDB mit:

```
SHOW PARAMETER optimizer_index_cost_adj
```

NAME	TYPE	VALUE
optimizer_index_cost_adj	integer	10

- Abfrage in der CDB mit:

```
SELECT name, display_value, con_id FROM v$system_parameter  
WHERE name = 'optimizer_index_cost_adj';
```

NAME	DISPLAY_VALUE	CON_ID
optimizer_index_cost_adj	20	0
optimizer_index_cost_adj	50	3

- Globaler Parameter auf CDB Level
- Jede PDB kann davon eine „Scheibe“ bekommen
- Über-Provisionierung ist möglich (First comes first served)

```
# CDB-Level
SQL> ALTER SYSTEM SET inmemory_size=1000M SCOPE=spfile;
SQL> SHUTDOWN IMMEDIATE
SQL> STARTUP

SQL> ALTER SESSION SET CONTAINER = <pdb1>;
SQL> ALTER SYSTEM SET inmemory_size=500M;
SQL> ALTER SESSION SET CONTAINER = <pdb2>;
SQL> ALTER SYSTEM SET inmemory_size=500M;
SQL> ALTER SESSION SET CONTAINER = <pdb3>;
SQL> ALTER SYSTEM SET inmemory_size=500M;
```

- Nur eine SGA
- Nur ein Set an Redologs
- Keine Einschränkung bezüglich I/O
- Ressourceneinschränkungen über den Resource Manager (z.B: CPU, Sessions, execution Time) und Parameter (nur Enterprise Edition)
- Einschränkung der Gesamtgröße einer PDB, z.B:
 - `ALTER PLUGGABLE DATABASE
STORAGE (MAXSIZE xG) ;`
- Eigener TEMP-TABLESPACE

- Jede PDB eigene SGA
 - `sga_max_size` → CDB / Instanz-Ebene
 - `sga_target` → PDB
 - `sga_min_size` → PDB
- Jede PDB eigene PGA
 - `pga_aggregate_target` → PDB
 - `pga_aggregate_limit` → PDB
- Die Summe aller SGA Parameter (`SGA_MIN_SIZE`, `DB_CACHE_SIZE`, etc.) aller PDBs darf nicht größer sein als 50% der SGA Size der CDB

- Einschränkung von Datenbank Optionen
 - Partitioning*
 - Database Queuing (warum auch immer)
- Datenbank Features
 - AWR*
 - OS_ACCESS
 - usw.
- SQL Befehle
 - ALTER DATABASE
 - ALTER PLUGGABLE DATABASE
 - ALTER SYSTEM
 - ALTER SESSION

* Nur Enterprise Edition

- Erstellen eines Profiles

```
SQL> CREATE LOCKDOWN PROFILE darfdasnicht;
```

- Entziehen von Features

```
SQL> ALTER LOCKDOWN PROFILE darfdasnicht  
      DISABLE FEATURE = ('AWR_ACCESS', 'JAVA', 'DROP_TABLESPACE_KEEP_DATAFILES');
```

- Entziehen von Statements

```
SQL> ALTER LOCKDOWN PROFILE darfdasnicht  
      DISABLE STATEMENT = ('ALTER SYSTEM') CLAUSE = ('SET');
```

- Entziehen von Optionen

```
SQL> ALTER LOCKDOWN PROFILE darfdasnicht  
      DISABLE OPTION = ('PARTITIONING');
```

- Zuweisen von Profiles

```
SQL> ALTER SESSION SET container=cello;  
  
SQL> ALTER SYSTEM SET PDB_LOCKDOWN=darfdasnicht SCOPE=SPFILE;  
  
SQL> ALTER PLUGGABLE DATABASE CLOSE;  
  
SQL> ALTER PLUGGABLE DATABASE OPEN;
```


- Fehlermeldung

```
SQL> SQL> ALTER SYSTEM SET optimizer_index_cost_adj=20;  
ALTER SYSTEM SET optimizer_index_cost_adj=20  
*  
ERROR at line 1:  
ORA-01031: insufficient privileges
```

- Überprüfung

```
SQL> show parameter lockdown
```

NAME	TYPE	VALUE
-----	-----	-----
pdb_lockdown	string	DARFDASNICHT

- Views

```
SQL> col rules FORMAT a30
SQL> col profile_name FORMAT a20
SQL> col rule_type FORMAT a20
SQL> SELECT profile_name, rule_type, rule, clause, con_id
        FROM cdb_lockdown_profiles;
```

PROFILE_NAME	RULE_TYPE	RULE	CLAUSE	CON_ID
DARFDASNICHT	FEATURE	AWR_ACCESS		1
DARFDASNICHT	FEATURE	DROP_TABLESPACE_KEEP_DATAFILES		1
DARFDASNICHT	FEATURE	JAVA		1
DARFDASNICHT	OPTION	PARTITIONING		1
DARFDASNICHT	STATEMENT	ALTER SYSTEM	SET	1
PRIVATE_DBAAS				1
PUBLIC_DBAAS				1
SAAS				1

PDB Cloning

- tnsnames.ora

```
XE_COHEN =  
  (DESCRIPTION =  
    (ADDRESS = (PROTOCOL = TCP) (HOST = cohen) (PORT = 1521))  
    (CONNECT_DATA =  
      (SERVER = DEDICATED)  
      (SERVICE_NAME = XE)  
    )  
  )  
  
XE_clapton =  
  (DESCRIPTION =  
    (ADDRESS = (PROTOCOL = TCP) (HOST = clapton) (PORT = 1521))  
    (CONNECT_DATA =  
      (SERVER = DEDICATED)  
      (SERVICE_NAME = XE)  
    )  
  )
```

- Quelle (cohen): Common User

```
SQL> CREATE USER c##johannes IDENTIFIED BY manager CONTAINER=ALL;  
SQL> GRANT create session, sysoper, sysdba TO c##johannes CONTAINER=ALL;
```

- Ziel (clapton): Database Link

```
SQL> CREATE DATABASE LINK xe_cohen  
      CONNECT TO c##johannes IDENTIFIED BY manager  
      USING 'XE_COHEN';
```

- PDB Cloning
 - NON-CDB (klassische Architektur)
 - PDB einer anderen Datenbank
- Über Datenbank Link

```
SQL> CREATE PLUGGABLE DATABASE leonard  
      FROM leonard@xe_cohen;
```

- Fehlermeldung OMF → NON-OMF

```
*  
ERROR at line 2:  
ORA-65016: FILE_NAME_CONVERT must be specified
```

- OMF definieren

```
SQL> ALTER SYSTEM SET db_create_file_dest='/u02/oradata';
```

```
SQL> CREATE PLUGGABLE DATABASE leonard  
      FROM leonard@xe_cohen;
```

```
SQL> ALTER PLUGGABLE DATABASE leonard OPEN;
```

```
SQL> ALTER PLUGGABLE DATABASE leonard SAVE STATE;
```

```
SQL> show pdbs
```

CON_ID	CON_NAME	OPEN MODE	RESTRICTED
2	PDB\$SEED	READ ONLY	NO
3	XEPDB1	READ WRITE	NO
4	LEONARD	READ WRITE	NO


```
oracle@clapton[XE]% sqlplus demo/demo@clapton/leonard
```

```
SQL*Plus: Release 21.0.0.0.0 - Production on Wed Nov 14 16:52:57 2021  
Version 21.3.0.0.0
```

```
Copyright (c) 1982, 2018, Oracle. All rights reserved.
```

```
Last Successful login time: Wed Nov 14 2021 16:52:48 +01:00
```

```
Connected to:
```

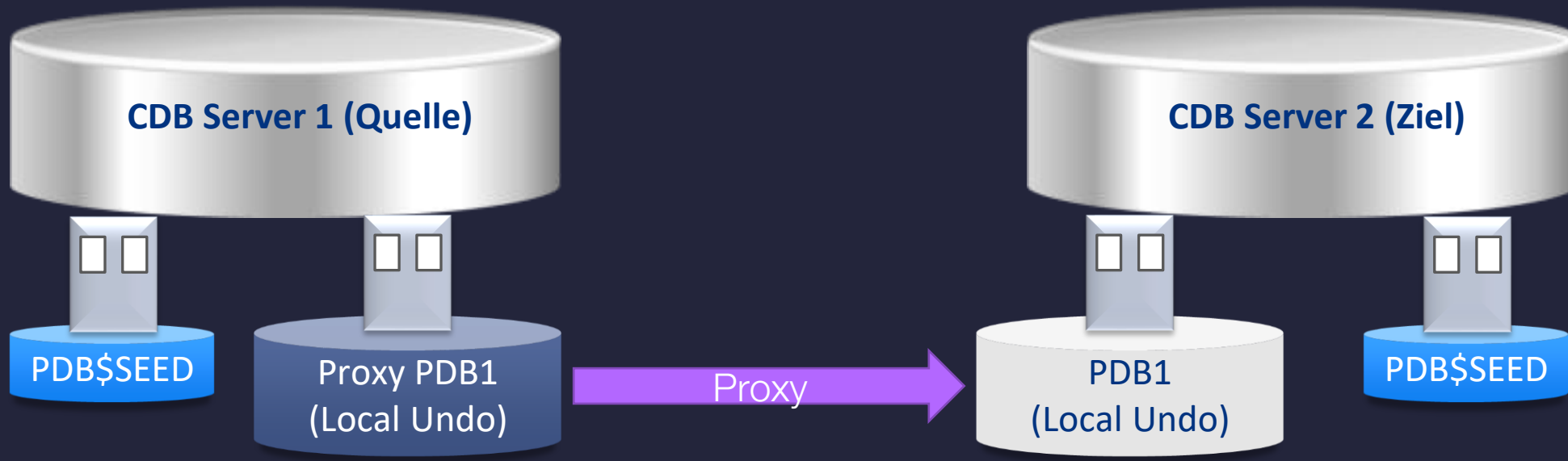
```
Oracle Database 21c Express Edition Release 21.0.0.0.0 - Production  
Version 21.3.0.0.0
```

```
SQL> SELECT count(*) FROM personen;
```

```
      COUNT (*)  
-----  
      8010000
```

Proxy PDB

Proxy PDB



- Quelle (cohen): Common User

```
SQL> CREATE USER c##johannes IDENTIFIED BY manager CONTAINER=ALL;  
SQL> GRANT create session, sysoper, sysdba TO c##johannes CONTAINER=ALL;
```

- Ziel (claption): Database Link

```
SQL> CREATE DATABASE LINK xe_cohen  
      CONNECT TO c##johannes IDENTIFIED BY manager  
      USING 'XE_COHEN';
```

```
SQL> CREATE PLUGGABLE DATABASE suzanne AS PROXY FROM leonard@xe_cohen;
```

```
SQL> ALTER PLUGGABLE DATABASE suzanne OPEN;
```

```
SQL> ALTER PLUGGABLE DATABASE suzanne SAVE STATE;
```

```
SQL> show pdbs
```

CON_ID	CON_NAME	OPEN MODE	RESTRICTED
2	PDB\$SEED	READ ONLY	NO
3	ANABELLE	READ WRITE	NO

```
SQL> ALTER SESSION SET CONTAINER=suzanne;
```

```
Session altered.
```

```
SQL> show pdbs
```

```
ERROR:
```

```
ORA-01017: invalid username/password; logon denied
```

```
SP2-1545: This feature requires Database availability.
```

```
oracle@mey[REINHARD]% sqlplus demo/demo@clapton/SUZANNE
```

```
...
```

```
SQL> select count(*) from personen;
```

```
      COUNT (*)
```

```
-----
```

```
      10000
```

```
SQL> SELECT d.name AS datafile, round(d.bytes/1024/1024) MByte
       FROM v$datafile WHERE con_id=3;
```

- Original (UDO)

DATAFILE	MBYTE
-----	-----
/u02/oradata/UDO/7A155AE705960D3FE0530100007F7B1E/datafile/o1_mf_system_fy5zf72y_.dbf	210
/u02/oradata/UDO/7A155AE705960D3FE0530100007F7B1E/datafile/o1_mf_sysaux_fy5zf73d_.dbf	165
/u02/oradata/UDO/7A155AE705960D3FE0530100007F7B1E/datafile/o1_mf_undotbs1_fy5zf73f_.dbf	155
/u02/oradata/UDO/7A155AE705960D3FE0530100007F7B1E/datafile/o1_mf_users_fy5zfgsq_.dbf	100

- Proxy (REINHARD)

DATAFILE	MBYTE
-----	-----
/u02/oradata/REINHARD/7A161E860F5D2075E053CB6410ACEE09/datafile/o1_mf_system_fy62p3rn_.dbf	210
/u02/oradata/REINHARD/7A161E860F5D2075E053CB6410ACEE09/datafile/o1_mf_sysaux_fy62p3rp_.dbf	165
/u02/oradata/REINHARD/7A161E860F5D2075E053CB6410ACEE09/datafile/o1_mf_undotbs1_fy62p3rq_.dbf	155


```
SQL> SELECT d.file_name AS datafile, round(d.bytes/1024/1024) MByte
       FROM cdb_data_files d WHERE con_id=3;
```

- Original (UDO)

DATAFILE	MBYTE
/u02/oradata/UDO/7A155AE705960D3FE0530100007F7B1E/datafile/o1_mf_system_fy5zf72y_.dbf	210
/u02/oradata/UDO/7A155AE705960D3FE0530100007F7B1E/datafile/o1_mf_sysaux_fy5zf73d_.dbf	165
/u02/oradata/UDO/7A155AE705960D3FE0530100007F7B1E/datafile/o1_mf_undotbs1_fy5zf73f_.dbf	155
/u02/oradata/UDO/7A155AE705960D3FE0530100007F7B1E/datafile/o1_mf_users_fy5zfgsq_.dbf	100

- Proxy (REINHARD)

```
no rows selected
```

Proxy Datafiles

- SYSTEM, SYSAUX und UNDO werden lokal gepflegt
- Anwendungstabespaces werden in der Originalen Datenbank gepflegt

- Create User

```
sqlplus system/manager@mey/ANABELLE
```

```
SQL> GRANT create session TO doag IDENTIFIED BY konferenz;
```

```
SQL> GRANT select ON demo.personen TO doag;
```

```
SQL> CONNECT doag/konferenz@mey/ANABELLE
```

```
SQL> SELECT count(*) FROM demo.personen;
```

```
  COUNT (*)  
-----  
      10000
```

```
sqlplus system/manager@lindenberg/CELLO
```

```
SQL> SELECT username FROM dba_users WHERE oracle_maintained='N';
```

```
USERNAME
```

```
-----
```

```
DEMO
```

```
PDB_ADMIN
```

```
DOAG
```

```
C##JOHANNES
```

```
SQL> CONNECT doag/konferenz@lindenberg/CELLO
```

```
SQL> SELECT count(*) FROM demo.personen;
```

```
COUNT (*)
```

```
-----
```

```
10000
```

- Warum?
 - Umzug der Anwendung
 - Andere Oracle Version
 - Umzug in die Cloud
- Warum nicht?
 - Netzwerkbandbreite!

```
SQL> CREATE PLUGGABLE DATABASE suzanne AS PROXY FROM leonard@xe_cohen;
```

```
SQL> ALTER PLUGGABLE DATABASE suzanne OPEN;
```

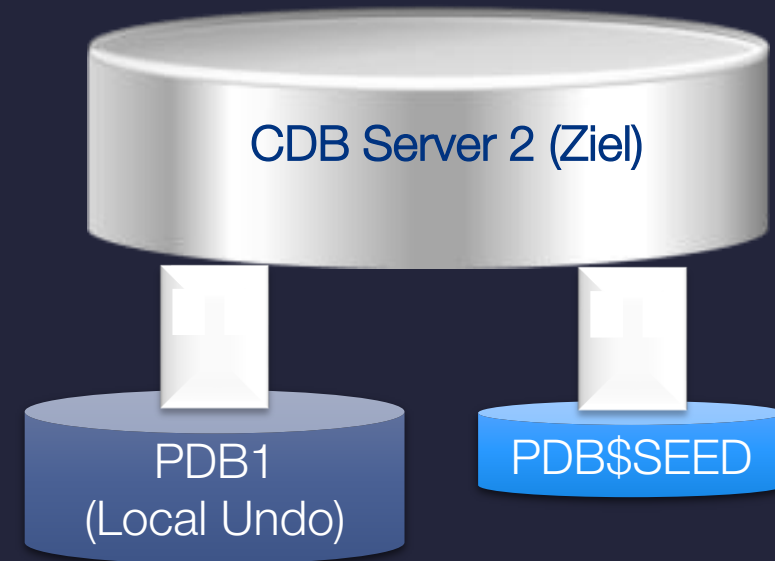
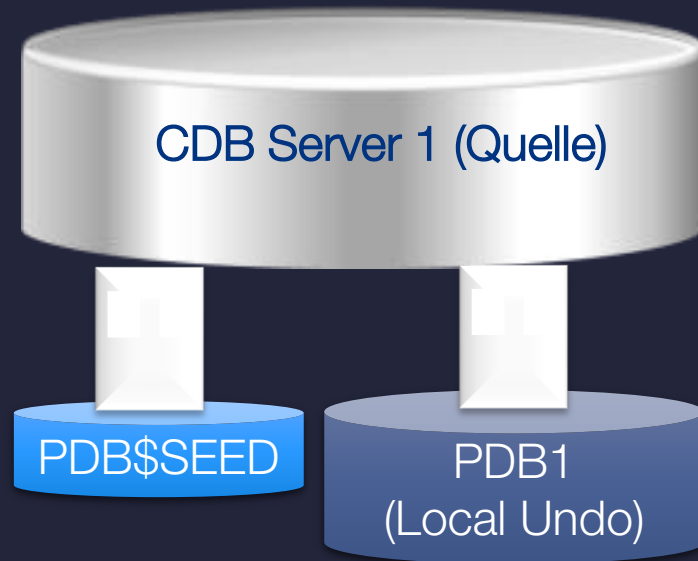
```
SQL> ALTER PLUGGABLE DATABASE suzanne SAVE STATE;
```

```
show pdba
```

CON_ID	CON_NAME	OPEN MODE	RESTRICTED
2	PDB\$SEED	READ ONLY	NO
3	ANABELLE	READ WRITE	NO

Refreshable PDB

- Automatische Synchronisation zwischen Quell- und Ziel-PDB
- Gedacht für Test und Entwicklung
- Refreshmodes:
 - None
 - Manual
 - Every n Minutes
- Ziel-PDB bleibt geschlossen



Erstellen der PDB

```
SQL> CREATE PLUGGABLE DATABASE friends  
      ADMIN USER pdbadmin IDENTIFIED BY manager  
      DEFAULT TABLESPACE users  
      DATAFILE size 100M AUTOEXTEND ON NEXT 10M MAXSIZE 1000M;  
  
SQL> ALTER PLUGGABLE DATABASE friends OPEN;  
  
SQL> ALTER PLUGGABLE DATABASE friends SAVE STATE;
```

Erstellen eines Services

```
SQL> ALTER SESSION SET CONTAINER=friends;
```

```
SQL> execute dbms_service.create_service('help_of_my','help_of_my');
```

```
SQL> execute dbms_service.start_service('help_of_my');
```

- Wird für den Datenbank Link benötigt (Quelldatenbank)

```
SQL> CREATE USER c##johannes IDENTIFIED BY manager CONTAINER=ALL;
```

```
User created.
```

```
SQL> GRANT create session, sysoper, sysdba TO c##johannes CONTAINER=ALL;
```

```
Grant succeeded.
```

- Derzeit muss explizit sysoper Privileg vergeben werden, sonst funktioniert der Refresh nicht!

Anlegen eines Datenbank Links

- Datenbank Link auf die CDB (nicht auf die PDB!)

```
SQL> CREATE DATABASE LINK XE_COCKER
      2  CONNECT TO c##johannes IDENTIFIED BY manager
      3  USING 'XE_COCKER';
```

Database link created.

- Beispiel: Refresh jede Minute

```
SQL> CREATE PLUGGABLE DATABASE friends
      2 FROM friends@cocker
      3 REFRESH MODE EVERY 1 MINUTES;
```

Pluggable database created.

```
SQL> show pdbs
```

CON_ID	CON_NAME	OPEN MODE	RESTRICTED
2	PDB\$SEED	READ ONLY	NO
3	FRIENDS	MOUNTED	

- VORSICHT: die Ziel-PDB darf nur Read-Only geöffnet werden!

```
SQL> ALTER PLUGGABLE DATABASE friends OPEN READ ONLY;
```

```
Pluggable database altered.
```

```
SQL> ALTER SESSION SET CONTAINER=friends;
```

```
Session altered.
```

```
SQL> SELECT count(*) FROM demo.personen;
```

```
  COUNT (*)  
-----  
        1000
```

```
SQL> SELECT pdb_name, con_id ,refresh_mode, refresh_interval  
2      FROM cdb_pdbs
```

PDB_NAME	CON_ID	REFRES	REFRESH_INTERVAL
-----	-----	-----	-----
PDB\$SEED	2	NONE	
FRIENDS	3	AUTO	1

- Alert-File auf der Zieldatenbank

```
Applying media recovery for pdb-4099 from SCN 637092 to SCN 637401
Remote log information: count-1 thr-1, seq-11, logfile-
/u03/orabackup/XE/foreign_archivelog/FRIENDS/2021_11_18/o1_mf_1_11_1911113739_.arc, los-
590304, nxs-18446744073709551615
CELLO(3):Media Recovery Start
2021-11-18T13:16:09.454723+01:00
CELLO(3):Serial Media Recovery started
2021-11-18T13:16:09.501618+01:00
CELLO(3):Media Recovery Log
/u03/orabackup/UDO/foreign_archivelog/FRIENDS/2021_11_18/o1_mf_1_11_1911113739_.arc
2021-11-18T13:16:10.656324+01:00
CELLO(3):Incomplete Recovery applied until change 637401 time 11/18/2021 13:16:07
2021-11-18T13:16:10.662316+01:00
CELLO(3):Media Recovery Complete (XE)
CELLO(3):Completed: alter pluggable database refresh
```

```
Begin  
execute immediate 'alter session set container = FRIENDS';  
execute immediate 'alter pluggable database refresh';  
end;
```

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

```
ALTER PLUGGABLE DATABASE OPEN
```

```
*
```

```
ERROR at line 1:
```

```
ORA-65341: cannot open pluggable database in read/write mode
```

```
SQL> ALTER SESSION SET CONTAINER=friends;
```

```
Session altered.
```

```
SQL> ALTER PLUGGABLE DATABASE REFRESH MODE NONE;
```

```
Pluggable database altered.
```

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

```
Pluggable database altered.
```



PDB Snapshot Copy

- Voraussetzung:
 - geeigneter Storage (z.B. NFS)
 - Parameter clonedb = TRUE (bedingt Instance Restart)
 - Vorsicht: Default für das Bitmap-File ist \$ORACLE_HOME/dbs
 - Quell-PDB Read-Only

```
SQL> ALTER SYSTEM SET clonedb = TRUE SCOPE=SPFILE;  
SQL> ALTER SYSTEM SET clonedb_dir='/u02/oradata/HANNES_S1/bitmap' scope=spfile;
```

- Create PDB

```
SQL> CREATE PLUGGABLE DATABASE karl FROM konstantin SNAPSHOT COPY;
```

- Dateien im Verzeichnis des Snapshot Clones

```
ls -l /u02/oradata/HANNES_S1/754AA536C9C006B7E053261E10AC2048/datafile
total 2148
-rw-r-----. 1 oracle oinstall 173023232 Sep  7 17:33 o1_mf_sysaux_fs568wxv_.dbf
-rw-r-----. 1 oracle oinstall 220209152 Sep  7 17:33 o1_mf_system_fs568wx7_.dbf
-rw-r-----. 1 oracle oinstall  20979712 Sep  7 17:28 o1_mf_temp_fs568wy9_.dbf
-rw-r-----. 1 oracle oinstall 230694912 Sep  7 17:33 o1_mf_undotbs1_fs568wy2_.dbf
-rw-r-----. 1 oracle oinstall  10493952 Sep  7 17:33 o1_mf_users_fs568wyw_.dbf
```

- Tatsächliche Größe

```
du -sh /u02/oradata/HANNES_S1/754AA536C9C006B7E053261E10AC2048/datafile
148K /u02/oradata/HANNES_S1/754AA536C9C006B7E053261E10AC2048/datafile
```

Backup und Restore

- CDB Backup immer erforderlich
 - CDB enthält die Redolog Dateien
 - CDB enthält die Parameter (spfile)
- PDB Backup

- Kein Unterschied zwischen NON-CDB und CDB

```
RMAN> backup database plus archivelog;
```

```
Starting backup at 29-FEB-20
```

```
current log archived
```

```
using target database control file instead of recovery catalog
```

```
allocated channel: ORA_DISK_1
```

```
channel ORA_DISK_1: SID=144 device type=DISK
```

```
channel ORA_DISK_1: starting archived log backup set
```

```
channel ORA_DISK_1: specifying archived log(s) in backup set
```

```
input archived log thread=1 sequence=20 RECID=1 STAMP=1033407034
```

```
input archived log thread=1 sequence=21 RECID=2 STAMP=1033407082
```

```
...
```

```
channel ORA_DISK_1: starting piece 1 at 29-FEB-20
```

```
channel ORA_DISK_1: finished piece 1 at 29-FEB-20
```

```
piece
```

```
handle=/u03/fast_recovery_area/UDO_S1/backupset/2020_02_29/o1_mf_annnn_TAG20200229T133156_h5npfxht_.
```

```
bkp tag=TAG20200229T133156 comment=NONE
```

```
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:07
```

```
Finished backup at 29-FEB-20
```

- 1. Teil Backup CDB

```
Starting backup at 29-FEB-20
using channel ORA_DISK_1
channel ORA_DISK_1: starting full datafile backup set
channel ORA_DISK_1: specifying datafile(s) in backup set
input datafile file number=00001 name=/u02/oradata/UDO_S1/datafile/o1_mf_system_h522gqmq_.dbf
input datafile file number=00003 name=/u02/oradata/UDO_S1/datafile/o1_mf_sysaux_h522h356_.dbf
input datafile file number=00005 name=/u02/oradata/UDO_S1/datafile/o1_mf_undotbs1_h522h7yv_.dbf
input datafile file number=00007 name=/u02/oradata/UDO_S1/datafile/o1_mf_users_h522h9fb_.dbf
channel ORA_DISK_1: starting piece 1 at 29-FEB-20
channel ORA_DISK_1: finished piece 1 at 29-FEB-20
piece
handle=/u03/fast_recovery_area/UDO_S1/backupset/2020_02_29/o1_mf_nnndf_TAG20200229T133205_h5npg7p1_.
bkp tag=TAG20200229T133205 comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:15
```

- 2. Teil Backup PDB\$SEED

```
SQL> SELECT con_id, name, guid FROM v$pdb$seeds WHERE con_id = 2;
→ 2      PDB$SEED      9F290C149A1D0795E055FB909B80DF97

channel ORA_DISK_1: starting full datafile backup set
channel ORA_DISK_1: specifying datafile(s) in backup set
input datafile file number=00009
name=/u02/oradata/UDO_S1/9F2C4A9C6B704912E055FB909B80DF97/datafile/o1_mf_system_h52j25dc_.dbf
input datafile file number=00011
name=/u02/oradata/UDO_S1/9F2C4A9C6B704912E055FB909B80DF97/datafile/o1_mf_undotbs1_h52j25f4_.dbf
input datafile file number=00010
name=/u02/oradata/UDO_S1/9F2C4A9C6B704912E055FB909B80DF97/datafile/o1_mf_sysaux_h52j25f3_.dbf
input datafile file number=00012
name=/u02/oradata/UDO_S1/9F2C4A9C6B704912E055FB909B80DF97/datafile/o1_mf_users_h52j25f5_.dbf
input datafile file number=00029
name=/u02/oradata/UDO_S1/9F2C4A9C6B704912E055FB909B80DF97/datafile/o1_mf_demo_h5f6x7c6_.dbf
channel ORA_DISK_1: starting piece 1 at 29-FEB-20
channel ORA_DISK_1: finished piece 1 at 29-FEB-20
piece
handle=/u03/fast_recovery_area/UDO_S1/9F2C4A9C6B704912E055FB909B80DF97/backupset/2020_02_29/o1_mf_nn
ndf_TAG20200229T133205_h5npgpwh_.bkp tag=TAG20200229T133205 comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:07
```

- 3. und folgende Teile Backup PDBs

```
SQL> SELECT con_id, name, guid FROM v$pdb$ WHERE con_id = 4;  
→ 4      DORIA      9F3AC1DE46E50A89E0554D86E6126571
```

```
channel ORA_DISK_1: starting full datafile backup set  
channel ORA_DISK_1: specifying datafile(s) in backup set  
input datafile file number=00013  
name=/u03/oradata/UDO_S1/9F3AC1DE46E50A89E0554D86E6126571/datafile/o1_mf_system_h54n42c4_.dbf  
input datafile file number=00015  
name=/u03/oradata/UDO_S1/9F3AC1DE46E50A89E0554D86E6126571/datafile/o1_mf_undotbs1_h54n42cj_.dbf  
input datafile file number=00014  
name=/u03/oradata/UDO_S1/9F3AC1DE46E50A89E0554D86E6126571/datafile/o1_mf_sysaux_h54n42cg_.dbf  
input datafile file number=00016  
name=/u03/oradata/UDO_S1/9F3AC1DE46E50A89E0554D86E6126571/datafile/o1_mf_users_h54n42ck_.dbf  
channel ORA_DISK_1: starting piece 1 at 29-FEB-20  
channel ORA_DISK_1: finished piece 1 at 29-FEB-20  
piece  
handle=/u03/fast_recovery_area/UDO_S1/9F3AC1DE46E50A89E0554D86E6126571/backupset/2020_02_29/o1_mf_nn  
ndf_TAG20200229T133205_h5npgz4_.bkp tag=TAG20200229T133205 comment=NONE  
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:07  
channel ORA_DISK_1: starting full datafile backup set
```

```
SQL> select count(*) from personen;
```

```
  COUNT(*)  
-----  
      10010
```

```
rman target /
```

```
RMAN> ALTER PLUGGABLE DATABASE cello CLOSE;
```

```
RMAN> RUN {
```

```
  SET UNTIL TIME "TO_DATE('2020-02-29 13:45:00','YYYY-MM-DD HH24:MI:SS')";
```

```
  RESTORE PLUGGABLE DATABASE cello;
```

```
  RECOVER PLUGGABLE DATABASE cello;
```

```
}
```

```
RMAN> ALTER PLUGGABLE DATABASE cello OPEN RESETLOGS;
```

```
SQL> select count(*) from personen;
```

```
  COUNT(*)  
-----  
      10000
```

Patches und Upgrades



New Paradigms for Rapid Patching and Upgrades

The investment of time and effort to patch one multitenant container database results in patching all of its many pluggable databases. To patch a single pluggable database, you simply unplug/plug to a multitenant container database at a different Oracle Database software version.

To upgrade all hosted pluggable databases in a container database, simply upgrade the container database and all hosted pluggable databases are upgraded 'in-place'. To upgrade a single PDB, you simply unplug/plug the pluggable database in to a container database at a higher version and upgrade the pluggable database as described in the [Database Upgrade Guide](#).

<http://www.oracle.com/technetwork/database/multitenant/overview/index.html>

ALTERNATIVE FAKTEN

- Die Investition von Zeit und Aufwand für das Patchen einer Multitenant-Containerdatenbank führt dazu, dass all die vielen zugehörigen plug-fähigen Datenbanken gepatcht werden. Um eine einzelne plug-fähige Datenbank zu patchen, trennen Sie einfach mit einer anderen Oracle Database-Softwareversion die Verbindung zu einer Multitenant-Containerdatenbank oder stellen sie her.

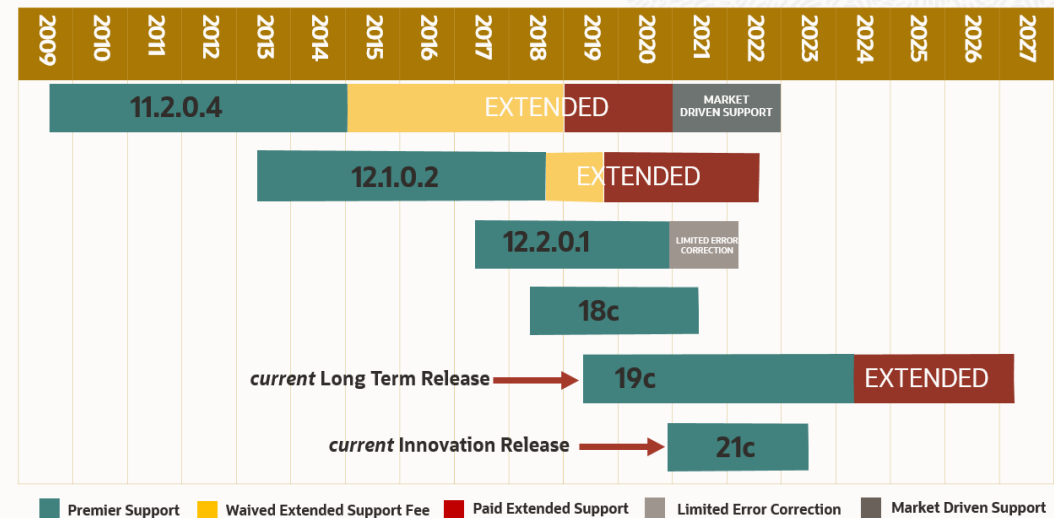
Quelle: <https://www.oracle.com/de/database/technologies/multitenant.html>

Warum 19c?

Long Term Support Release

- Premier Support (PS) ends April 30, 2024
- Extended Support (ES) fees will be required beginning May 01, 2024 through April 30, 2027
- Error Correction / Patching is available through April 30, 2027 with paid ES.
Without paid ES, patching is only available until April 30, 2024

Database Releases and Support Timelines



Derzeitige Releases

- 11.2.0.4 → Premier (Standard) Support bis 31. Januar 2015
Free Extended Support bis 31. Dezember 2018
Extended Support bis 31. Dezember 2020
- 12.1.0.1 → Premier Support bis 31. August 2016 (kein Extended Support)
- 12.1.0.2 → Premier Support bis 31. Juli 2018
Free Extended Support bis 31. Juli 2019
Extended Support bis 31. Juli 2022
- 12.2.0.1 → Premier Support bis 30. November 2020
Limited Error Correction Support bis 31. März 2022
- 18c → Premier Support bis 30. Juni 2021 (kein Extended Support)
- 19c → Premier Support bis 30. April 2024 ...
- 20c → gibt es nicht!
- 21c → Premier Support bis 30. Juni 2023 (kein Extended Support)

- Innovation Release:
 - Neue Features
 - Kein Extended Support
 - Zuerst benutzt bei 18c
 - 21c und wohl auch 22c
- Terminal Release
 - Letztes Release eines Zyklus
 - 11.2.0.4 und 12.1.0.2
- Long Term Support Release
 - Neue Namensgebung für Terminal Release
 - 19c und wahrscheinlich 23c

Long Term oder Innovation Release

- Long Term Release:
 - *Oracle Database Long Term Releases are ideal for use cases that benefit from less frequent upgrades to newer releases. Long Term Releases offer the highest level of stability and the longest length of error correction support. These releases have 5 years of Premier Support followed by 3 years of Extended Support. When combined with Extended Support, customers typically have almost 4 years to upgrade from one Long Term Release to the next Long Term Release.*
- Innovation Release:
 - *In between Oracle Database Long Term Releases, Oracle delivers Oracle Database Innovation Releases that include many enhancements and new capabilities which will also be included in the next Long Term Release. Innovation Releases are designed to enable customers to continuously use leading-edge technologies to rapidly develop or deploy new applications or augment existing applications. Support for Innovation Releases includes 2 years of Premier Support, but there is no Extended Support. Production workloads can be deployed on Innovation Releases if upgrading within 2 years to a newer release is factored into the deployment plan.*

+ Security

- [2 New Features in 19c Release Updates](#)

- Release Update 19.7 Features

SQL Macros (SQM)

- Release Update 19.8 Features

Database In-Memory Base Level

CellMemory Level

- Release Update 19.9 Features

Oracle Grid Infrastructure SwitchHome

Support for DBMS_CCRYPTO Asymmetric
Key Operations

- Release Update 19.10 Features

DBMS_CLOUD Package

New Database Initialization Parameters for
Database Resident Connection Pooling
(DRCP)

Oracle Blockchain Table

Oracle Instant Client Support for Linux for
ARM

Support Per-PDB Capture for Oracle
Autonomous Database

- Release Update 19.11 Features

Application Continuity Protection Check

Immutable Tables

New Database Initialization Parameter and
View for Database Resident Connection

2 New Features in 19c Release Updates



This chapter describes the features that are new in Oracle Database 19c Release Updates (RUs).

- Neu in Release 19.8.0.0.200714

Database In-Memory Base Level



Database In-Memory is an option to Enterprise Edition. Database In-Memory now has a new Base Level feature. This allows the use of Database In-Memory with up to a 16GB column store without triggering any license tracking.

This feature allows you to use Database In-Memory without having to license the option. The column store is limited to 16GB when using the Base Level. This helps to show the value of Database In-Memory without having to worry about licensing issues.

Related Topics

- [Oracle® Database In-Memory Guide](#)

Quelle: Oracle® Database New Features Guide 19c 13 May 2021

- Neu in 19.9.0.0.201020

Oracle Grid Infrastructure SwitchHome



You can use the `-switchGridHome` option with `gridSetup.sh` to switch from one Oracle Grid Infrastructure home to another.

You can use the `-switchGridHome` option for patching and upgrading Oracle Grid Infrastructure. Use the `-switchGridHome` option to switch from the source Oracle Grid Infrastructure home to the patched Oracle Grid Infrastructure home. All Oracle Clusterware and Oracle Restart services start from the patched Oracle Grid Infrastructure home automatically.

Related Topics

- [Oracle® Grid Infrastructure Installation and Upgrade Guide for Linux](#)

- Neu in 19.9.0.0.201020

Oracle Blockchain Table



Blockchain tables are append-only tables in which only insert operations are allowed. Deleting rows is either prohibited or restricted based on time. Rows in a blockchain table are made tamper-resistant by special sequencing and chaining algorithms. Users can verify that rows have not been tampered. A hash value that is part of the row metadata is used to chain and validate rows.

Blockchain tables can be used to implement blockchain applications where the participants trust the Oracle Database provider, but want means to verify that their data hasn't been tampered with. The participants are different database users who trust the Oracle Database provider to maintain a verifiable, tamper-resistant blockchain of transactions. All participants must have privileges to insert data into the blockchain table. The contents of the blockchain table are defined and managed by the application, with a few added metadata fields maintained by Oracle Database. By leveraging a trusted provider with verifiable crypto-secure data management practices, such applications can avoid the distributed consensus requirements. This provides most of the protection of the distributed peer-to-peer blockchains, but with much higher throughput and lower transaction latency compared to peer-to-peer blockchains using distributed consensus.

Related Topics

- [Oracle® Database Administrator's Guide](#)

- Einfache Suche nach Patches / Bundle Patches, Release Updates, etc.

★ Assistant: Download Reference for Oracle Database/GI Update, Revision, PSU, SPU(CPU), Bundle Patches, Patchsets and Base Releases (Doc ID 2118136.2) [To Bottom](#)

Selection(s)

What would you like to download?

- ☐ Oracle Database Base Releases
- ☐ Oracle Database Patchsets
- ☒ Oracle Database Updates (Versions 12.2 & higher)
- ☐ Oracle Database Update Revisions (Versions 12.2 & higher)
- ☐ Oracle Database PSU, SPU(CPU), Bundle Patches (Versions 12.1 & lower)
- ☐ OJVM Update/PSU/Bundle Patches
- ☐ Latest Available Microsoft Windows Patches

Additional Clarification

Please select an option above to begin clarifying the issue you are experiencing.

Solution(s)

Possible Solutions will appear once you make your selection.

REFERENCES

[NOTE:274523.1](#) - How To Download And Install The Latest OPatch(6000000) Version

[NOTE:1963039.1](#) - Latest Patch Set Update and Critical Patch Update Availability Document

[NOTE:293369.1](#) - Master Note For OPatch

[NOTE:802103.1](#) - Oracle Recommended Patches -- Previous Oracle Database Recommended Patches

[NOTE:148054.1](#) - Oracle Database 8.1.7 (8i) & 8.1.7.x (8i) Notice 2 of 2

[NOTE:2584628.1](#) - JDK and PERL Patches for Oracle Database Home and Grid Home

[NOTE:1929745.1](#) - Oracle Recommended Patches -- "Oracle JavaVM Component Database PSU and Update" (OJVM PSU and OJVM Update) Patches



Upgrade

Upgrade 12.1 → 19

- Quelle: 12.1.0.2 NON-CDB (DB_NAME → JOTESTK)
- Ziel: 19.6.0 PDB
- Voraussetzung: 19c CDB existiert schon (CDB-NAME → CDB01)
- Quelle: Mike Dietrich
 - <https://mikedietrichde.com/2019/07/25/database-migration-from-non-cdb-to-pdb-clone-via-noncdb-upgrade-convert>

- Erstellen eines Benutzers für die Migration (joupgrade)

```
sqlplus / as sysdba  
SQL> CREATE USER joupgrade IDENTIFIED BY manager;  
SQL> GRANT connect, resource, create pluggable database TO joupgrade;
```

- Erstellen eines PDB manifest

```
SQL> SHUTDOWN IMMEDIATE  
SQL> STARTUP OPEN READ ONLY  
  
SQL> EXEC dbms_pdb.describe('/home/oracle/upgrade/JOTESTK.xml');  
SQL> EXIT;
```

- Kopieren des Manifests auf das Zielsystem

```
scp /home/oracle/upgrade/JOTESTK.xml lxora19:/home/oracle/upgrade/JOTESTK.xml
```

- Erstellen eines TNS-Aliases für die Quelldatenbank (JOTESTK)

```
JOTESTK =  
  (DESCRIPTION =  
    (ADDRESS = (PROTOCOL = TCP) (HOST = lxora12) (PORT = 151))  
    (CONNECT_DATA =  
      (SERVER = DEDICATED)  
      (SERVICE_NAME = JOTESTK)  
    )  
  )  
)
```

- Überprüfen der Kompatibilität

```
DECLARE
    compatible CONSTANT VARCHAR2(3) :=
        CASE DBMS_PDB.CHECK_PLUG_COMPATIBILITY(
            pdb_descr_file => '/home/oracle/upgrade/JOTESTK.xml',
            pdb_name        => 'JOTESTK')
            WHEN TRUE THEN 'YES' ELSE 'NO'
        END;
BEGIN
    DBMS_OUTPUT.PUT_LINE('Is the future PDB compatible? ==> ' || compatible);
END;
/

==> Jede Menge Fehler und Warnings.
Wichtig: CTX muss in der CDB installiert sein
1 JOTESTK  ERROR      Database option CONTEXT mismatch: PDB installed version
12.1.0.2.0. CDB installed version NULL. PENDING
```


- Anlegen eines Datenbank-Links

```
SQL> CREATE DATABASE LINK jotestk  
      CONNECT TO joupgrade  
      IDENTIFIED BY manager USING 'JOTESTK';
```

- Local Undo Ausschalten (wegen 12.1 Kompatibilität)

```
SQL> SHUTDOWN IMMEDIATE  
SQL> STARTUP UPGRADE  
SQL> ALTER DATABASE LOCAL UNDO OFF;  
SQL> SHUTDOWN IMMEDIATE  
SQL> STARTUP
```

- Ansonsten beim Versuch, die PDB anzulegen:

```
CREATE PLUGGABLE DATABASE jotestk FROM non$cdb@jotestk  
*  
ERROR at line 1:  
ORA-65353: The undo tablespace is missing from the XML metadata file.
```

- Kopieren der Datenbank über Datenbank Link

```
SQL> CREATE PLUGGABLE DATABASE jotestk FROM non$cdb@jotestk;  
...
```

- Öffnen der Datenbank

```
SQL> ALTER PLUGGABLE DATABASE jtestk OPEN;
```

```
Warning: PDB altered with errors.
```

```
SQL> show pdbs
```

CON_ID	CON_NAME	OPEN MODE	RESTRICTED
2	PDB\$SEED	READ ONLY	NO
3	JTEST1	READ WRITE	NO
4	JTESTK	MIGRATE	YES

- Upgrade der Datenbank nach Version 19

```
mkdir -p /home/oracle/jo/logs
dbupgrade -l /home/oracle/jo/logs -c "JOTESTK"
```

```
...
```

```
Grand Total Upgrade Time:      [0d:0h:24m:56s]
```

```
sqlplus / as sysdba
```

```
SQL> show pdbs
```

CON_ID	CON_NAME	OPEN MODE	RESTRICTED
2	PDB\$SEED	READ ONLY	NO
3	JOTEST1	READ WRITE	NO
4	JOTESTK	MOUNTED	

- Konvertieren der Datenbank in eine PDB

```
SQL> ALTER SESSION SET container=jotestk;

SQL> SET timing ON
SQL> SET serveroutput ON
SQL> SET echo ON
SQL> SET termout ON
SQL> spool /home/oracle/jo/logs/jotestk_noncdb2pdb.log
SQL> @?/rdbms/admin/noncdb_to_pdb.sql
...
Dauer: 1 Stunde !!!
```

- Öffnen der neuen PDB

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

```
SQL> show pdbs
```

CON_ID	CON_NAME	OPEN MODE	RESTRICTED
2	PDB\$SEED	READ ONLY	NO
3	JOTEST1	READ WRITE	NO
4	JOTESTK	READ WRITE	NO

```
SQL> ALTER PLUGGABLE DATABASE SAVE STATE;
```

- Local Undo einschalten

```
sqlplus / as sysdba
```

```
SQL> SHUTDOWN IMMEDIATE
```

```
SQL> STARTUP UPGRADE
```

```
SQL> ALTER DATABASE LOCAL UNDO ON;
```

```
SQL> SHUTDOWN IMMEDIATE
```

```
SQL> STARTUP
```


- Rollen ändern
 - MYDBA → C##MYDBA
- User löschen
 - johannes → c##johannes
- Profiles anpassen
 - mydefault → c##mydefault
- Default Tablespaces löschen
 - Es ist sinnvoll, nur einen Default Tablespace zu haben

Migration ohne Upgrade

Migration NON-CDB → PDB

- Quelle: 19.6 NON-CDB (DB_NAME → JOTESTK)
- Ziel: 19.6.0 PDB
- Voraussetzung: 19c CDB existiert schon (CDB-NAME → CDB01)
- Quelle: Mike Dietrich
 - <https://mikedietrichde.com/2019/07/25/database-migration-from-non-cdb-to-pdb-clone-via-noncdb-upgrade-convert>

- Erstellen eines Benutzers für die Migration (joupgrade)

```
sqlplus / as sysdba  
SQL> CREATE USER joupgrade IDENTIFIED BY manager;  
SQL> GRANT connect, resource, create pluggable database TO joupgrade;
```

- Erstellen eines PDB manifest

```
SQL> SHUTDOWN IMMEDIATE  
SQL> STARTUP OPEN READ ONLY  
  
SQL> EXEC dbms_pdb.describe('/home/oracle/upgrade/JOTESTK.xml');  
SQL> EXIT;
```

- Kopieren des Manifests auf das Zielsystem

```
scp /home/oracle/upgrade/JOTESTK.xml lxora19:/home/oracle/upgrade/JOTESTK.xml
```

- Erstellen eines TNS-Aliases für die Quelldatenbank (JOTESTK)

```
JOTESTK =  
  (DESCRIPTION =  
    (ADDRESS = (PROTOCOL = TCP) (HOST = lxora12) (PORT = 151))  
    (CONNECT_DATA =  
      (SERVER = DEDICATED)  
      (SERVICE_NAME = JOTESTK)  
    )  
  )  
)
```

- Überprüfen der Kompatibilität

```
DECLARE
    compatible CONSTANT VARCHAR2(3) :=
        CASE DBMS_PDB.CHECK_PLUG_COMPATIBILITY(
            pdb_descr_file => '/home/oracle/upgrade/JOTESTK.xml',
            pdb_name        => 'JOTESTK')
            WHEN TRUE THEN 'YES' ELSE 'NO'
        END;
BEGIN
    DBMS_OUTPUT.PUT_LINE('Is the future PDB compatible? ==> ' || compatible);
END;
/

==> Jede Menge Fehler und Warnings.
Wichtig: CTX muss in der CDB installiert sein
1 JOTESTK  ERROR      Database option CONTEXT mismatch: PDB installed version
12.1.0.2.0. CDB installed version NULL. PENDING
```

- Anlegen eines Datenbank-Links

```
SQL> CREATE DATABASE LINK jotestk  
      CONNECT TO joupgrade  
      IDENTIFIED BY manager USING 'JOTESTK';
```


- Kopieren der Datenbank über Datenbank Link

```
SQL> CREATE PLUGGABLE DATABASE jotestk FROM non$cdb@jotestk;  
...
```

- Öffnen der Datenbank

```
SQL> ALTER PLUGGABLE DATABASE jtestk OPEN;
```

```
Warning: PDB altered with errors.
```

```
SQL> show pdbs
```

CON_ID	CON_NAME	OPEN MODE	RESTRICTED
2	PDB\$SEED	READ ONLY	NO
3	JOTEST1	READ WRITE	NO
4	JOTESTK	MIGRATE	YES

- Konvertieren der Datenbank in eine PDB

```
SQL> ALTER SESSION SET container=jotestk;

SQL> SET timing ON
SQL> SET serveroutput ON
SQL> SET echo ON
SQL> SET termout ON
SQL> spool /home/oracle/jo/logs/jotestk_noncdb2pdb.log
SQL> @?/rdbms/admin/noncdb_to_pdb.sql
...
Dauer: 1 Stunde !!!
```

- Öffnen der neuen PDB

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

```
SQL> show pdbs
```

CON_ID	CON_NAME	OPEN MODE	RESTRICTED
2	PDB\$SEED	READ ONLY	NO
3	JOTEST1	READ WRITE	NO
4	JOTESTK	READ WRITE	NO

```
SQL> ALTER PLUGGABLE DATABASE SAVE STATE;
```

Autoupgrade

- Java-basierendes Tool für Upgrade
 - Aktuelle Version über MOS Note: 2485457.1 – AutoUpgrade Tool
 - Minimum Version 11.2.0.4
 - Zielversion 12.2, 18c, 19c, 21c
- Infos: <https://mikedietrichde.com/2019/04/29/the-new-autoupgrade-utility-in-oracle-19c/>
- Details: Oracle Dokumentation: Database Upgrade Guide

1. Sample Konfigurationsdatei

```
$ORACLE_HOME/jdk/bin/java -jar ./autoupgrade.jar -create_sample_file config config.cfg
```

- Erstellt die Datei `sample_config.cfg` mit mehreren Beispieleinträgen
- Einschränkungen sind möglich (z.B. full für NON-CDB Upgrades)

2. Konfigurationsdatei

- Dedizierte Version

```
cp sample_config.cfg configJOSEPH.cfg
```

```
cat configJOSEPH.cfg
```

```
upg1.log_dir=/home/oracle/autoupgrade
upg1.sid=JOSEPH
upg1.source_home=/u01/app/oracle/product/12.2.0/dbhome_1
upg1.target_home=/u01/app/oracle/product/19.0/dbhome_1
upg1.start_time=NOW
upg1.upgrade_node=haydn
upg1.run_utlrp=yes
#upg1.timezone_upg=[yes|no]
#upg1.target_version=[12.2|18|19|21]
```


3. Analyse

```
java -jar ./autoupgrade.jar -config configJOSEPH.cfg -mode analyze
```

```
No parameter 'global.autoupg_log_dir' found in config file, using
/u01/app/oracle/cfgtoollogs/autoupgrade
AutoUpgrade tool launched with default options
Processing config file ...
+-----+
| Starting AutoUpgrade execution |
+-----+
1 databases will be analyzed
Type 'help' to list console commands
upg> Job 100 completed
----- Final Summary -----
Number of databases          [ 1 ]

Jobs finished                [1]
Jobs failed                  [0]
Jobs pending                 [0]

Please check the summary report at:
/u01/app/oracle/cfgtoollogs/autoupgrade/cfgtoollogs/upgrade/auto/status/status.html
/u01/app/oracle/cfgtoollogs/autoupgrade/cfgtoollogs/upgrade/auto/status/status.log
```

- Parameter `global.autoupgrade_log_dir` auf das Log Verzeichnis setzen
 - Ansonsten zwei Verzeichnisse mit Log-Dateien

```
cat configJOSEPH.cfg
```

```
global.autoupgrade_log_dir=/home/oracle/autoupgrade  
upg1.log_dir=/home/oracle/autoupgrade  
upg1.sid=JOSEPH  
upg1.source_home=/u01/app/oracle/product/12.2.0/dbhome_1  
upg1.target_home=/u01/app/oracle/product/19.0/dbhome_1  
upg1.start_time=NOW  
upg1.upgrade_node=haydn  
upg1.run_utlrp=yes  
#upg1.timezone_upg=[yes|no]  
#upg1.target_version=[12.2|18|19|21]
```

4. Upgrade

```
java -jar ./autoupgrade.jar -config configJOSEPH.cfg -mode deploy
```

```
AutoUpgrade tool launched with default options
```

```
Processing config file ...
```

```
+-----+
```

```
| Starting AutoUpgrade execution |
```

```
+-----+
```

```
1 databases will be processed
```

```
Type 'help' to list console commands
```

```
upg>
```

- Jeder „run“ hat seine eigenen Log-Verzeichnisse:
- Im Beispiel:
 - 100 → Analyze Lauf
 - 101 → Deployment
- Zwei wichtige Logdateien
 - Übersicht:

```
/home/oracle/autoupgrade/JOSEPH/101/autoupgrade_20210426_user.log
```

- Details:

```
/home/oracle/autoupgrade/JOSEPH/101/dbupgrade/autoupgrade20210426111829joseph.log
```

- Bei Aufruf aus einem Scheduler (z.B. Control-M) Parameter `-noconsole` setzen

```
java -jar ./autoupgrade.jar -config configJOSEPH.cfg -mode deploy -noconsole
```

- Autoupgrade ist Restartable!

```
java -jar ./autoupgrade.jar -config configJOSEPH.cfg -mode deploy
```

```
Previous execution found loading latest data
Total jobs recovered: 1
+-----+
| Starting AutoUpgrade execution |
+-----+
Type 'help' to list console commands
upg>
```

- Außerdem gibt es einen Guaranteed Restore Point (nicht für SE2!)

```
SQL> select name from v$restore_point;
```

```
NAME
```

```
-----
AUTOUPGRADE_9212_JOSEPH122010
```

- Bei Fehlern
 - Backup einspielen (Standard Edition 2)
 - Flashback auf Guaranteed Restore Point

```
SQL> select name from v$restore_point;
```

```
NAME
```

```
-----  
AUTOUPGRADE_9212_JOSEPH122010
```

- Log-Dateien löschen!
 - Deployment merkt sich den Status der Datenbank

- Autoupgrade 12.1 → 19c NON-CDB
- Insgesamt ca. 200 Datenbanken (Vmware)
- Derzeitiges Stand ca. 120 Datenbanken erfolgreich migriert
- Kein einziger Fehler!!!

- Pending Transactions
 - Einträge in 2pc_pending führen zum Abbruch der Migration
 - Logfile-Eintrag: TWO_PC_TXN_EXIST
- v\$recover_file
 - Einträge führen zum Abbruch der Migration
 - Oft bei Data Guard und der PDB\$SEED nach Switchover (laut Oracle kein Bug!)
- Shared ORACLE_HOME/dbs
 - Verschieben der init<SID>.ora bzw. spfile<SID>.ora schlägt fehl
- Zentrales TNS_ADMIN Verzeichnis
 - Probleme mit Listener, wird nicht neu gestartet

- v\$recover_file

```
SQL> SELECT * FROM v$recover_file;
```

FILE#	ONLINE	ONLINE_	ERROR	CHANGE#	TIME	CON_ID
2	ONLINE	ONLINE		1229668	30-APR-21	2
4	ONLINE	ONLINE		1229668	30-APR-21	2
6	ONLINE	ONLINE		1229668	30-APR-21	2
8	ONLINE	ONLINE		1229668	30-APR-21	2

- Lösung: PDB\$SEED read-write öffnen und wieder schließen

- OLTP Datenbanken
 - NON-CDB auf Vmware
 - Dauer ca. 30 Minuten
- OLTP Datenbank
 - Multitenant mit 2 PDBs (ca. 1 TB)
 - Dauer ca. 1,5 Stunden
- Data Warehouse Datenbanken
 - Multitenant mit 6 PDBs (ins. 6 TByte)
 - Dauer ca. 1,5 Stunden

Patching

- CMD> cd
%ORACLE_HOME%\OPatch
- CMD> opatch lsinventory

```
Administrator: Command Prompt
C:\Users\Administrator>cd ..\..
C:\>cd C:\app\svc-oracle\product\12.1.0.2\dbhome_1\OPatch
C:\app\svc-oracle\product\12.1.0.2\dbhome_1\OPatch>opatch lsinventory

Oracle Interim Patch-Installationsprogramm Version 12.1.0.1.3
Copyright (c) 2015, Oracle Corporation. All rights reserved. Alle Rechte vorbehalten.

Oracle Home      : C:\app\SVC-OR~1\product\1210~1.2\dbhome_1
Central Inventory : C:\Program Files\Oracle\Inventory
                  from      : n/a
OPatch version   : 12.1.0.1.3
OUI version      : 12.1.0.2.0
Log file location : C:\app\SVC-OR~1\product\1210~1.2\dbhome_1\cfgtoollogs\opatch\
\opatch2015-11-06_13-16-06PM_1.log

Lsinventory Output file location : C:\app\SVC-OR~1\product\1210~1.2\dbhome_1\cfg
toollogs\opatch\lsinv\lsinventory2015-11-06_13-16-06PM.txt

-----

Installierte Produkte der obersten Ebene (1):

Oracle Database 12c                               12.1.0.2.0
In diesem Oracle-Standardverzeichnis sind 1 Produkte installiert.

In diesem Oracle-Standardverzeichnis sind keine Interim-Patches installiert.

-----

OPatch succeeded.
C:\app\svc-oracle\product\12.1.0.2\dbhome_1\OPatch>
```

- Neueste OPatch Version über My Oracle Support beziehen:

- Patch 6880880
- aktuell 21.0.0.0.28

Patch 6880880

[Simple Search](#) [Advanced Search](#) [Quick Links](#) [Saved Searches](#)

Description	OPatch 12.2.0.1.25 for DB 12.x, 18.x, 19.x, 20.x and 21.x releases (May 2021)
Product	Oracle Global Lifecycle Management OPatch
Select a Release	OPatch 19.0.0.0.0
Platform or Language	Linux x86-64
Last Updated	18-MAY-2021
Size	116M (121981878 bytes)
Entitlement Class	Software
Classification	General

[Download](#) [View Readme](#) [View Digest](#)

Patchsets known to include or supersede Patch 6880880

No information available from the patch repository

Download Notes

Copyright (c) 2015, Oracle. All Rights Reserved.
[Legal Notices and Terms of Use](#)
[Privacy Statement](#)

- Opatch FAQ (Frequently Asked Questions) (Doc ID 1486109.1)

- OPatch patchen
 - Software in ein temporäres Verzeichnis laden (z.B. /tmp)
 - Wechseln in ORACLE_HOME (`cd $ORACLE_HOME`)
 - Löschen der alten Version (`rm -rf OPatch/*`)
 - Auspacken der neuen Version (`unzip /tmp/p6880880`)
 - Überprüfen der Version (`./OPatch/opatch version`)

- Wichtige Seite:

The screenshot shows the Oracle My Oracle Support (MOS) interface. At the top, there's a navigation bar with tabs: Dashboard, Knowledge, Service Requests, Patches & Updates, Community, Certifications, Systems, Collector, Advanced Customer Services, and Settings. Below the navigation bar, a copyright notice states "Copyright (c) 2018, Oracle. All rights reserved." The main content area features a star icon and the title "Assistant: Download Reference for Oracle Database/GI Update, Revision, PSU, SPU(CPU), Bundle Patches, Patchsets and Base Releases (Doc ID 2118136.2)". Under the "Selection(s)" heading, a section titled "What would you like to download?" contains a list of radio button options: Oracle Database Base Releases, Oracle Database Patchsets, Oracle Database Updates (Versions 12.2 & higher), Oracle Database Update Revisions (Versions 12.2 & higher), Oracle Database PSU, SPU(CPU), Bundle Patches (Versions 12.1 & lower), OJVM Update/PSU/Bundle Patches, and Latest Available Microsoft Windows Patches. Below this is an "Additional Clarification" section with the text "Please select an option above to begin clarifying the issue you are experiencing." The "REFERENCES" section lists several Oracle support notes, including NOTE:148054.1, NOTE:293369.1, NOTE:783141.1, NOTE:161549.1, NOTE:274526.1, NOTE:756671.1, NOTE:1430923.1, NOTE:1963039.1, NOTE:1581950.1, NOTE:1071023.1, NOTE:802103.1, NOTE:1351428.1, and NOTE:1929745.1. At the bottom, there's a search bar with the text "Didn't find what you are looking for?" and a button "Ask in Community...". A "Related" section is partially visible at the very bottom.

ORACLE MY ORACLE SUPPORT PowerView is Off

Dashboard Knowledge Service Requests Patches & Updates Community Certifications Systems Collector Advanced Customer Services Settings

Copyright (c) 2018, Oracle. All rights reserved.

☆ Assistant: Download Reference for Oracle Database/GI Update, Revision, PSU, SPU(CPU), Bundle Patches, Patchsets and Base Releases (Doc ID 2118136.2)

Selection(s)

What would you like to download?

- ☐ Oracle Database Base Releases
- ☐ Oracle Database Patchsets
- ☐ Oracle Database Updates (Versions 12.2 & higher)
- ☐ Oracle Database Update Revisions (Versions 12.2 & higher)
- ☐ Oracle Database PSU, SPU(CPU), Bundle Patches (Versions 12.1 & lower)
- ☐ OJVM Update/PSU/Bundle Patches
- ☐ Latest Available Microsoft Windows Patches

Additional Clarification

Please select an option above to begin clarifying the issue you are experiencing.

REFERENCES

[NOTE:148054.1](#) - Oracle Database 8.1.7 (8i) & 8.1.7.x (8i) Notice 2 of 2
[NOTE:293369.1](#) - Master Note For OPatch
[NOTE:783141.1](#) - Reference List of Critical Patch Update Availability(CPU) and Patch Set Update (PSU) Documents For Oracle Database and Fusion Middleware Product
[NOTE:161549.1](#) - Oracle Database, CRS, ASM, Networking and EM Agent Patches for Microsoft Platforms
[NOTE:274526.1](#) - How To Download And Install The Latest OPatch(6880880) Version
[NOTE:756671.1](#) - Master Note for Database Proactive Patch Program
[NOTE:1430923.1](#) - New Patch Nomenclature for Oracle Products
[NOTE:1963039.1](#) - Latest Patch Set Update and Critical Patch Update Availability Document
[NOTE:1581950.1](#) - Database Security Patching from 12.1.0.1 onwards
[NOTE:1071023.1](#) - How to Request Software Media when it is Unavailable on the Oracle Software Delivery Cloud Site
[NOTE:802103.1](#) - Oracle Recommended Patches -- Previous Oracle Database Recommended Patches
[NOTE:1351428.1](#) - Information Center: Patching and Maintaining Oracle Database Server/Client Installations
[NOTE:1929745.1](#) - Oracle Recommended Patches -- "Oracle JavaVM Component Database PSU and Update" (OJVM PSU and OJVM Update) Patches

< Didn't find what you are looking for? [Ask in Community...](#)

Related

- MOS > Patches & Updates > Patch ...

ORACLE MY ORACLE SUPPORT PowerView is Off Sebastian (Available) (0) Contact Us Help

Knowledge Service Requests **Patches & Updates** Community Certifications More...

Patch Details

Patch 21821214: WINDOWS DB BUNDLE PATCH 12.1.0.2.10

Last Updated: 20-Oct-2015 17:53 (16 days ago) Size: 471.4 MB

Product: Oracle Database - Enterprise Edition (More...) Download Access: Software Classification: Security Patch Tag: All Database

Release: Oracle 12.1.0.2.0 Platform: Microsoft Windows x64 (64-bit)

Recommendations / Certifications
Recommended for Oracle Database 12.1.0.2.0

Bugs Resolved by This Patch

Bug ID	Description
14103428	46719: XML REDO NOT GENERATED FOR APPLY SIDE RPI
14643995	ONLY 1 SQLPATCH INVOCATION AT A TIME
14851828	REQUEST SELF-EXPLANATORY REBOOT ADVISORY MESSAGE
15894842	INSERT IN TO STAT TABLE FAILS WITH ORA-00932: INCONSISTENT DATATYPES
15996428	SESSIONS SID BLOCKING 0 SESSIONS
16286734	NLS:12.1_WING4_130128:THERE ARE MANY QUESTION MASK IN GI INSTALLATION LOG
16293223	HEALTH FACTOR IMPROVEMENTS.
16359751	ORA-600[QCSGPVC:TB2ZEVCOLNO:COLUMNS] DURING INSERT
16619249	VPD : ORA-7445 IN [KZRTVEV] AND KZFTVEV
16756406	SR12.1.0.2RC - HNG - KPP_CONCATQ

[Open Readme to View all Bugs](#)

[View Related Knowledge to this Patch](#)

Release Oracle 12.1.0.2.0
Platform Microsoft Windows x64 (64-bit)
Language American English

[Read Me](#) [Download](#) [Add to Plan](#) [Analyze with OPatch...](#)

All-time Downloads **No history available**

[Discussion in community.oracle.com](#) [Reply to](#)

2646280

error when applying post patch steps
Tuesday
Posted by system on behalf of 2646280

I followed the instructions for this patch and get the error when proceeding "loading ...SQL files.." (the post step prereq checks failed! [patch 21153530](#): The pluggable that need to be patched must be in upgrade mode Prereq check failed, exiting without installing any patches. Don't remember I had similar issues from the previous patch...OK. I am still in 11G world. So,...don't really care for now.

[Follow the discussion here](#) [Refresh](#)

[Read Me](#) [Download](#) [Add to Plan](#) [Analyze with OPatch...](#)

All-time Downloads **No history available** [View Trends](#)

Copyright (c) 2015, Oracle. All rights reserved. Legal Notices and Terms of Use 02.02.2022

- MOS > Patches & Updates > Readme

Oracle® Database Patch 33509923 - GI Release Update 19.14.0.0.220118

Oracle® Database

Patch 33509923 - GI Release Update 19.14.0.0.220118

In this document, Oracle Database home refers to Oracle Database Enterprise Edition and Oracle Database Standard Edition.

The GI Release Update 19.14.0.0.220118 includes updates for both the Oracle Grid Infrastructure home and Oracle Database home that can be applied in a rolling fashion.

This patch is Oracle Data Guard Standby-First Installable. See [Installing Patch in Oracle Data Guard Standby-First Mode](#) for more information.

The database subpatch includes the JDK fixes released in the prior cycle and will update the JDK in the Oracle home. For the most recent JDK fixes, a separate patch is available and needs to be installed in addition to this patch. Refer to My Oracle Support document [888.1 - Primary Note for Database Proactive Patch Program](#) for the JDK patch number.

For the latest update with security fixes that should be used on client-only installations, refer to the Critical Patch Update (CPU) Program Patch Availability Document (PAD) section on the Oracle Database for the cycle in which you are interested.

This document is accurate at the time of release. For any changes and additional information regarding GI Release Update 19.14.0.0.220118, see this related document that is available at My Oracle Support (<http://support.oracle.com/>):

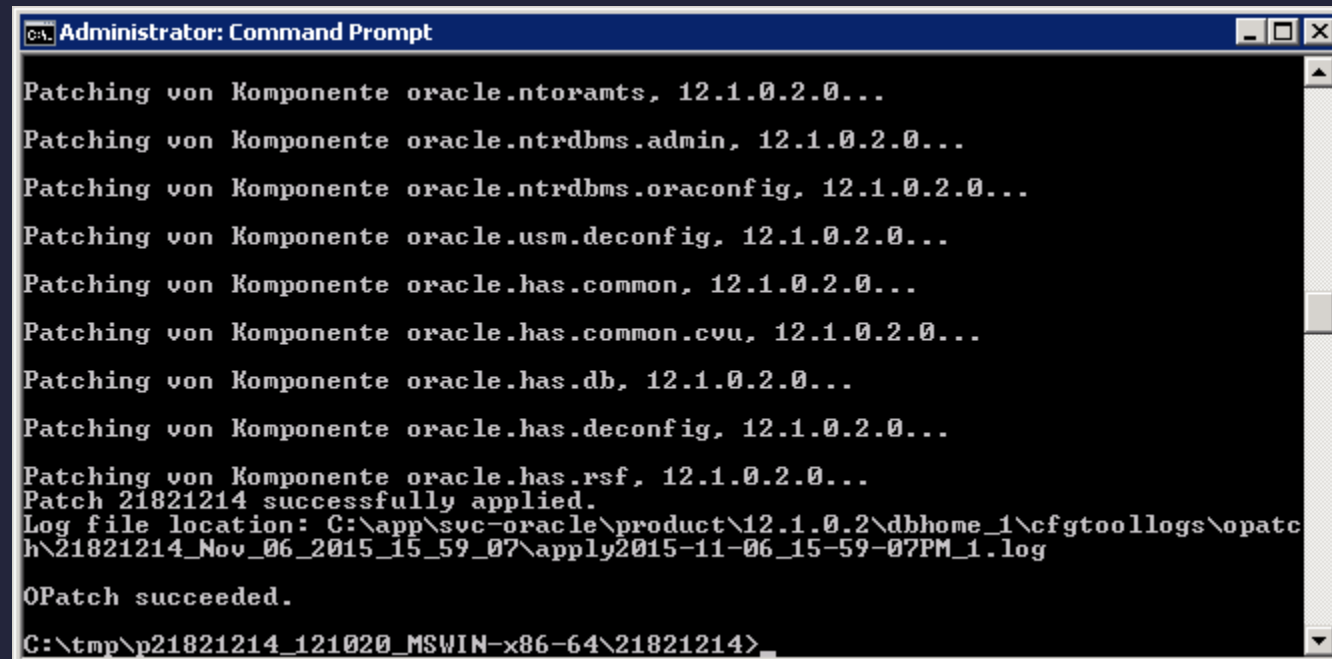
- Note [19202201.9](#) Oracle Database 19c RU/RUR Jan 2022 Known Issues

This document includes the following sections:

- [1.1 Patch Information](#)
- [2.1 Patch Installation and Deinstallation](#)

Oracle Patchen (Windows)

- p21821214...zip Download entpacken
- SQL> shutdown immediate
- Stop aller Oracle Services
- CMD> set
PATH=C:\app\oracle\product\12.1.0.2\dbhome_1\perl\bin;%PATH%
- CMD> cd C:\tmp\p21821214_121020_MSWIN-x86-64\21821214
- CMD> %ORACLE_HOME%\OPatch\opatch apply
- CMD> opatch lsinventory / lspatches



```
Administrator: Command Prompt

Patching von Komponente oracle.ntonants, 12.1.0.2.0...
Patching von Komponente oracle.ntrdbms.admin, 12.1.0.2.0...
Patching von Komponente oracle.ntrdbms.oraconfig, 12.1.0.2.0...
Patching von Komponente oracle.usm.deconfig, 12.1.0.2.0...
Patching von Komponente oracle.has.common, 12.1.0.2.0...
Patching von Komponente oracle.has.common.cvu, 12.1.0.2.0...
Patching von Komponente oracle.has.db, 12.1.0.2.0...
Patching von Komponente oracle.has.deconfig, 12.1.0.2.0...
Patching von Komponente oracle.has.rsfs, 12.1.0.2.0...
Patch 21821214 successfully applied.
Log file location: C:\app\svc-oracle\product\12.1.0.2\dbhome_1\cfgtoollogs\opatch\21821214_Nov_06_2015_15_59_07\apply2015-11-06_15-59-07PM_1.log

OPatch succeeded.

C:\tmp\p21821214_121020_MSWIN-x86-64\21821214>
```

- Patch auspacken

```
cd /home/oracle  
unzip p32904851_190000_Linux-x86-64.zip
```

- Alle Datenbanken, Listener, Befehle (z.B. SQL*Plus) beenden
- Installation des Patches

```
cd 32904851  
$ORACLE_HOME/Opatch/opatch apply
```

- Starten der Datenbanken und Listener
- Patchen der Datenbank

```
$ORACLE_HOME/Opatch/datapatch apply
```

- Überprüfung

```
$ORACLE_HOME/Opatch/opatch lspatches
```

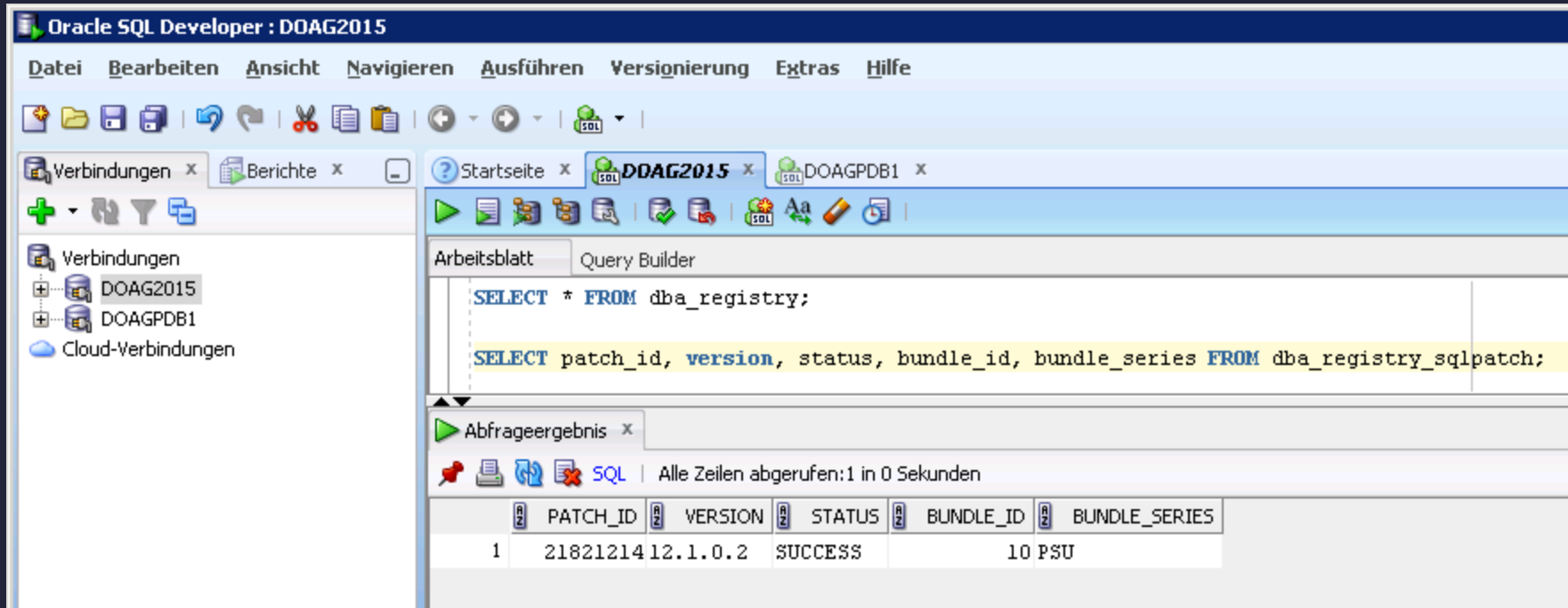
- Optimizerpatches werden installiert aber nicht aktiviert!
- Zum Aktivieren Package optim_bundle:

```
sqlplus / as sysdba
SQL> execute dbms_optim_bundle.enable_optim_fixes('ON','BOTH', 'YES');
```

- Resultat

```
sqlplus / as sysdba
SQL> show parameter fix_control
```

NAME	TYPE	VALUE
-----	-----	-----
_fix_control	string	29331066:1, 28965084:1, 287768 11:1, 28498976:1, 28567417:1, 28558645:1, 29132869:1, 294508 ...



```
SQL> SELECT patch_id, version, status, bundle_id, bundle_series FROM dba_registry_sqlpatch;
```

```
SQL> SELECT * FROM dba_registry;
```

```
SQL> SELECT * FROM dba_registry_history;
```

Patch Probleme

```
$ORACLE_HOME/OPatch/datapatch
```

- Standardmäßig CDB und alle PDBs
- Alternativ
 - -pdbs <pdb1,pdb2,...>


```
SQL> show pdbs
```

CON_ID	CON_NAME	OPEN	MODE	RESTRICTED
2	PDB\$SEED	READ	ONLY	NO
3	WE8ISO8859P15	READ	WRITE	NO
4	TEST1	READ	WRITE	NO
5	GOOFY	READ	WRITE	NO
6	MINIMAU	READ	WRITE	NO
7	GUSTAF	READ	WRITE	NO
8	DAGOBERT	READ	WRITE	YES
9	DONALD	READ	WRITE	NO
10	...			

- PDB DAGOBERT nicht gepatcht und nicht nutzbar

pdb_plug_in_violations

```
SQL> SELECT * FROM pdb_plug_in_violations WHERE name = 'DAGOBERT';
```

TIME	NAME	CAUSE	TYPE	ERROR_NUMBER	LINE	MESSAGE	STATUS	ACTION	CON_ID
25-JUL-18 03.23.05.380143 PM	DONALD	Sync Failure	WARNING	990	1	Sync PDB failed with ORA-00990 while performing 'REVOKE COMMIT ON "SYS"."MAP_OBJECT" FROM "C##DBA" CONTAINER=ALL'	PENDING		8
25-JUL-18 03.23.05.382327 PM	DONALD	Sync Failure	WARNING	990	2	Sync PDB failed with ORA-00990 while performing 'REVOKE REWRITE ON "SYS"."MAP_OBJECT" FROM "C##DBA" CONTAINER=ALL'	PENDING		8
25-JUL-18 03.23.05.389343 PM	DONALD	Sync Failure	WARNING	911	3	Sync PDB failed with ORA-00911 while performing 'REVOKE ON "SYS"."MAP_OBJECT" FROM "C##DBA" CONTAINER=ALL'	PENDING		8
25-JUL-18 03.23.14.056746 PM	DONALD	PDB not Unicode	WARNING	0	1	Character set mismatch: PDB character set WE8ISO8859P15. CDB character set AL32UTF8.	RESOLVED	Oracle recommends using Unicod er set for the database. Consi tabase to Unicode.	8
28-FEB-20 08.27.08.089965 AM	DONALD	SQL patch error	ERROR	0	1	DBRU bundle patch 200114 (DATABASE JAN 2020 RELEASE UPDATE 12.2.0.1.200114): APPLY with status BEGIN in the PDB.	PENDING	Call datapatch to reinstall	
28-FEB-20 08.27.08.091048 AM	DONALD	SQL Patch	ERROR	0	2	DBRU bundle patch 200114 (DATABASE JAN 2020 RELEASE UPDATE 12.2.0.1.200114): Installed in the CDB but not in the PDB.	PENDING	Call datapatch to install in t	8

- Ab 19c werden „*resolved*“ Einträge gelöscht

```
% cd /u01/app/oracle/product/12.2.0/dbhome_1/OPatch

% ./datapatch -pdbs DAGOBERT -force
SQL Patching tool version 12.2.0.1.0 Production on Fri Feb 28 10:20:39 2020
Copyright (c) 2012, 2019, Oracle. All rights reserved.

Connecting to database...OK
Note: Datapatch will only apply or rollback SQL fixes for PDBs
      that are in an open state, no patches will be applied to closed PDBs.
      Please refer to Note: Datapatch: Database 12c Post Patch SQL Automation
      (Doc ID 1585822.1)
Bootstrapping registry and package to current versions...done
Determining current state...done
Adding patches to installation queue and performing prereq checks...done
Installation queue:
  For the following PDBs: DAGOBERT
    Nothing to roll back
    Nothing to apply

SQL Patching tool complete on Fri Feb 28 10:20:52 2020
```

- Keine Infos aus der PDB

```
SQL> SELECT * FROM cdb_registry_sqlpatch WHERE con_id=8;
```

```
no rows selected
```

- PDB hat keine Verbindung zur CDB → no rows selected

- DROP PLUGGABLE DATABASE

```
SQL> ALTER PLUGGABLE DATABASE dagobert CLOSE IMMEDIATE INSTANCES=ALL;
```

```
Pluggable database altered.
```

```
SQL> ALTER PLUGGABLE DATABASE dagobert UNPLUG INTO '/home/oracle/dagobert.xml';
```

```
Pluggable database altered.
```

```
SQL> DROP PLUGGABLE DATABASE dagobert;
```

```
Pluggable database dropped.
```

- CREATE PLUGGABLE DATABASE

```
SQL> CREATE PLUGGABLE DATABASE dagobert USING '/home/oracle/dagobert.xml' NOCOPY;
```

Pluggable database created.

```
SQL> ALTER PLUGGABLE DATABASE dagobert OPEN INSTANCES=ALL;
```

Warning: PDB altered with errors.

Patchen neue PDB

```
./datapatch -pdba dagobert
SQL Patching tool version 12.2.0.1.0 Production on Fri Feb 28 10:50:42 2020
Copyright (c) 2012, 2019, Oracle. All rights reserved.

Connecting to database...OK
Note: Datapatch will only apply or rollback SQL fixes for PDBs
      that are in an open state, no patches will be applied to closed PDBs.
      Please refer to Note: Datapatch: Database 12c Post Patch SQL Automation
      (Doc ID 1585822.1)
Bootstrapping registry and package to current versions...done
Determining current state...done
Adding patches to installation queue and performing prereq checks...done
Installation queue:
  For the following PDBs: DAGOBERT
  Nothing to roll back
  The following patches will be applied:
    30593149 (DATABASE JAN 2020 RELEASE UPDATE 12.2.0.1.200114)

Installing patches...
Patch installation complete. Total patches installed: 1

Validating logfiles...done
SQL Patching tool complete on Fri Feb 28 10:54:10 2020
```

Öffnen der PDB

```
SQL> ALTER PLUGGABLE DATABASE dagobert CLOSE IMMEDIATE INSTANCES=ALL;
```

Pluggable database altered.

```
SQL> ALTER PLUGGABLE DATABASE dagobert OPEN INSTANCES=ALL;
```

Pluggable database altered.


```
SQL> show pdbs
```

CON_ID	CON_NAME	OPEN	MODE	RESTRICTED
2	PDB\$SEED	READ	ONLY	NO
3	WE8ISO8859P15	READ	WRITE	NO
4	TEST1	READ	WRITE	NO
5	GOOFY	READ	WRITE	NO
6	MINIMAU	READ	WRITE	NO
7	GUSTAF	READ	WRITE	NO
8	DAGOBERT	READ	WRITE	NO
9	DONALD	READ	WRITE	NO
10	...			

Nicht vergessen!

```
SQL> ALTER PLUGGABLE DATABASE dagobert SAVE STATE INSTANCES=ALL;
```



Data Guard

CDB Read Only

- Die CDB und PDB\$SEED dürfen ohne ADG Lizenz Read Only geöffnet sein

- Oracle Managed Files:
 - Alle Datafiles werden automatisch angelegt
 - Tempfiles werden nicht angelegt
- Non OMF
 - Verzeichnis muss existieren

```
SQL> CREATE PLUGGABLE DATABASE fellbach ADMIN USER pdbadmin IDENTIFIED BY manager;  
SQL> ALTER PLUGGABLE DATABASE fellbach OPEN;  
SQL> ALTER PLUGGABLE DATABASE fellbach SAVE STATE;
```

- Standby Datenbank

```
SQL> show pdbs
```

CON_ID	CON_NAME	OPEN MODE	RESTRICTED
2	PDB\$SEED	MOUNTED	
3	FELLBACH	MOUNTED	

```
SQL> SELECT name FROM v$datafile WHERE con_id=3;
```

NAME

```
-----  
/u02/oradata/DOAG_BER/D6DD4C5C156B5C4BE05500000000001/datafile/o1_mf_system_jzh69259_.dbf  
/u02/oradata/DOAG_BER/D6DD4C5C156B5C4BE05500000000001/datafile/o1_mf_sysaux_jzh6925h_.dbf  
/u02/oradata/DOAG_BER/D6DD4C5C156B5C4BE05500000000001/datafile/o1_mf_undotbs1_jzh6925j_.dbf  
/u02/oradata/DOAG_BER/D6DD4C5C156B5C4BE05500000000001/datafile/o1_mf_users_jzh6925j_.dbf
```

```
DGMGRL> validate database "DOAG_BER"
```

```
Database Role:      Physical standby database  
Primary Database:   DOAG_CGN
```

```
Ready for Switchover:  Yes  
Ready for Failover:   Yes (Primary Running)
```

```
Flashback Database Status:  
DOAG_CGN:  Off  
DOAG_BER:  Off
```

```
Managed by Clusterware:  
DOAG_CGN:  NO  
DOAG_BER:  NO  
Validating static connect identifier for the primary database DOAG_CGN...  
The static connect identifier allows for a connection to database "DOAG_CGN".
```

```
Temporary Tablespace File Information:  
DOAG_CGN TEMP Files: 3  
DOAG_BER TEMP Files: 2
```

- Switchover und dann Tempfile anlegen
- Alternative: Standby Datenbank Read-Only öffnen

```
DGMGRL> EDIT DATABASE "DOAG_BER" SET STATE=apply-off;
```

```
SQL> ALTER DATABASE OPEN READ ONLY;
```

```
SQL> ALTER PLUGGABLE DATABASE fellbach OPEN READ ONLY;
```

```
SQL> ALTER SESSION SET CONTAINER=fellbach;
```

```
SQL> ALTER TABLESPACE TEMP ADD TEMPFILE SIZE 100M AUTOEXTEND ON NEXT 100M MAXSIZE 1000M;
```

```
SQL> ALTER SESSION SET CONTAINER=CDB$ROOT;
```

```
SQL> SHUTDOWN IMMEDIATE
```

```
SQL> STARTUP MOUNT
```

```
DGMGRL> EDIT DATABASE "DOAG_BER" SET STATE=apply-on;
```


Cloning einer PDB

- Oracle Managed Files:
 - Datafiles werden NICHT (!) angelegt
 - Ausnahme: Active Data Guard
 - Tempfiles werden nicht angelegt

```
SQL> CREATE PLUGGABLE DATABASE mannheim FROM fellbach;  
SQL> ALTER PLUGGABLE DATABASE mannheim OPEN;  
SQL> ALTER PLUGGABLE DATABASE mannheim SAVE STATE;
```

- Standby Datenbank

```
SQL> show pdba
```

CON_ID	CON_NAME	OPEN MODE	RESTRICTED
2	PDB\$SEED	MOUNTED	
3	FELLBACH	MOUNTED	
4	MANNHEIM	MOUNTED	

```
SQL> SELECT name FROM v$datafile WHERE con_id=4;
```

NAME

```
/u01/app/oracle/product/19/dbhome_1/dbs/UNNAMED00013  
/u01/app/oracle/product/19/dbhome_1/dbs/UNNAMED00014  
/u01/app/oracle/product/19/dbhome_1/dbs/UNNAMED00015  
/u01/app/oracle/product/19/dbhome_1/dbs/UNNAMED00016
```

Erstellen der Standby PDB

```
DGMGRL> edit database "DOAG_BER" set state=apply-off;

RMAN> run {
    ALLOCATE CHANNEL prim1 TYPE DISK;
    SET NEWNAME FOR PLUGGABLE DATABASE mannheim TO NEW;
    RESTORE PLUGGABLE DATABASE mannheim FROM SERVICE DOAG_CGN;
    SWITCH DATAFILE ALL;
}
...
channel prim1: specifying datafile(s) to restore from backup set
channel prim1: restoring datafile 00016 to
/u02/oradata/DOAG_BER/D6DD862270775FD0E055000000000001/datafile/o1_mf_users_%u_.dbf
channel prim1: restore complete, elapsed time: 00:00:03
Finished restore at 31-JAN-22

datafile 13 switched to datafile copy
input datafile copy RECID=17 STAMP=1095413900 file
...
SQL> ALTER SESSION SET CONTAINER=mannheim;
SQL> ALTER PLUGGABLE DATABASE ENABLE RECOVERY;

DGMGRL> EDIT DATABASE "DOAG_BER" SET STATE=apply-on;
```

- Standby Datenbank

```
SQL> SELECT name FROM v$datafile WHERE con_id=4;
```

```
NAME
```

```
-----  
/u02/oradata/DOAG_BER/D6DD862270775FDFE055000000000001/datafile/o1_mf_system_jzh7zwc7_.dbf  
/u02/oradata/DOAG_BER/D6DD862270775FDFE055000000000001/datafile/o1_mf_sysaux_jzh803jv_.dbf  
/u02/oradata/DOAG_BER/D6DD862270775FDFE055000000000001/datafile/o1_mf_undotbs1_jzh806pc_.dbf  
/u02/oradata/DOAG_BER/D6DD862270775FDFE055000000000001/datafile/o1_mf_users_jzh809t9_.dbf
```

- Dann noch TEMPFILE anlegen!

Active Data Guard

- Data Files werden nicht angelegt
- Tempfiles werden nicht angelegt
- Ausnahme: wenn Quell-PDB Read-Only ist

- Standby Datenbank

```
SQL> ALTER DATABASE OPEN;
```

```
DGMGRL> SHOW DATABASE "DOAG_BER"
```

Database - DOAG_BER

```
Role:                PHYSICAL STANDBY
Intended State:       APPLY-ON
Transport Lag:        0 seconds (computed 1 second ago)
Apply Lag:            0 seconds (computed 1 second ago)
Average Apply Rate:   1.00 KByte/s
Real Time Query:    ON
Instance(s) :
    DOAG
```

```
Database Status:
SUCCESS
```

```
SQL> ALTER PLUGGABLE DATABASE mannheim OPEN READ ONLY FORCE;
SQL> CREATE PLUGGABLE DATABASE mainz FROM mannheim;
SQL> ALTER PLUGGABLE DATABASE mainz OPEN;
SQL> ALTER PLUGGABLE DATABASE mainz SAVE STATE;
```

- Standby Datenbank

```
SQL> SELECT name, status FROM v$datafile WHERE con_id=5;
```

NAME	STATUS
-----	-----
-	
/u02/oradata/DOAG_BER/D6DE52E642027118E055000000000001/datafile/o1_mf_system_jzhblpsk_.dbf	SYSTEM
/u02/oradata/DOAG_BER/D6DE52E642027118E055000000000001/datafile/o1_mf_sysaux_jzhblpsv_.dbf	ONLINE
/u02/oradata/DOAG_BER/D6DE52E642027118E055000000000001/datafile/o1_mf_undotbs1_jzhblpsw_.dbf	ONLINE
/u02/oradata/DOAG_BER/D6DE52E642027118E055000000000001/datafile/o1_mf_users_jzhblpsx_.dbf	ONLINE

```
SQL> SELECT name, status FROM v$tempfile WHERE con_id=5;
```

```
no rows selected
```

ADG PDB Snapshot Copy

- Führt derzeit zum Crash der Standby Umgebung (ORA-600)
- Einfaches Recovery nicht möglich!

- Standby nur für bestimmte PDBs
- `ALTER SYSTEM enable_pdbson_standbys = ...`
 - Kann pro Standby Datenbank gesetzt werden
- Variablen:
 - `"*"` : Alle PDBs
 - `"PDB1"` : bestimmte PDB
 - `"PDB*"` : Alle PDBs, die mit dem Namen „PDB“ beginnen
 - `"*", "-PDB1"` : alle PDBs außer PDB1
 - `"*", "-PDB*"` : PDBs, die mit diesem Namen beginnen, nicht
 - `"*", "PDBx"` : Alle PDBs, außer wenn Sie mit „PDB“ anfangen



Application Container

- „Vererbung“ einer Anwendung an weitere PDBs
- Bis 12.1 → Zugriff auf Daten einer anderen PDB nur über Datenbank Link
- Ab 12.2 → Application Container
- Besteht aus:
 - Application Container Root
 - Enthält die zu teilende Anwendung
 - Application Container
 - Anwendungs-PDBs mit eigenen Daten und den „Metadaten“ aus dem Root Container

```
SQL> CREATE PLUGGABLE DATABASE appweg  
2 AS APPLICATION CONTAINER  
3 ADMIN USER pdb_admin IDENTIFIED BY manager  
4 DEFAULT TABLESPACE users;
```

Pluggable database created.

```
SQL> ALTER PLUGGABLE DATABASE appweg OPEN INSTANCES=ALL;
```

Pluggable database altered.

```
SQL> SQL> ALTER PLUGGABLE DATABASE appweg SAVE STATE;
```

Pluggable database altered.

- Erstellen neuer PDBs immer aus dem Application Container Root heraus

```
SQL> ALTER SESSION SET CONTAINER=appweg;
```

Session altered.

```
SQL> CREATE PLUGGABLE DATABASE appweg_flugzeuge  
2  ADMIN USER pdb_admin IDENTIFIED BY manager  
3  DEFAULT TABLESPACE users;
```

Pluggable database created.

```
SQL> ALTER PLUGGABLE DATABASE appweg_flugzeuge OPEN INSTANCES=ALL;
```

Pluggable database altered.

```
SQL> ALTER PLUGGABLE DATABASE appweg_flugzeuge SAVE STATE;
```

Pluggable database altered.

- Wichtig ist, eine Application zu definieren

```
SQL> ALTER SESSION SET CONTAINER=appweg;
```

```
Session altered.
```

```
SQL> ALTER PLUGGABLE DATABASE APPLICATION stammdaten BEGIN INSTALL '0.1';
```

```
Pluggable database altered.
```

```
SQL> @SCRIPT
```

```
...
```

```
SQL> ALTER PLUGGABLE DATABASE APPLICATION stammdaten END INSTALL;
```

```
Pluggable database altered.
```

- Applikationsname und Version sind frei wählbar

```
SQL> SELECT app_name, app_version, app_status  
2      FROM dba_applications  
3      WHERE app_name = 'STAMMDATEN';
```

APP_NAME	APP_VERSION	APP_STATUS
STAMMDATEN	0.1	NORMAL

- Zunächst der Application Container Root

```
SQL> ALTER SESSION SET CONTAINER=appweg;  
  
Session altered.  
  
SQL> SELECT count(*) FROM basis.bundeslaender;  
  
COUNT (*)  
-----  
16
```

- Und der Application Constainer?

```
SQL> ALTER SESSION SET CONTAINER=appweg_flugzeuge;  
  
Session altered.  
  
SQL> SELECT count(*) FROM basis.bundeslaender;  
SELECT count(*) FROM basis.bundeslaender  
                        *  
ERROR at line 1:  
ORA-00942: table or view does not exist
```


- Erst nach der Synchronisierung sind die Daten sichtbar

```
SQL> ALTER SESSION SET CONTAINER=appweg_flugzeuge;
```

```
Session altered.
```

```
SQL> ALTER PLUGGABLE DATABASE APPLICATION stammdaten SYNC;
```

```
Pluggable database altered.
```

```
SQL> SELECT count(*) FROM basis.bundeslaender;
```

```
  COUNT (*)  
-----  
          16
```

- Vor jeder Änderung an den Basisdaten muss eine neue Version erstellt werden:

```
ALTER SESSION SET CONTAINER=appweg;
```

Session altered.

```
SQL> ALTER PLUGGABLE DATABASE APPLICATION stammdaten BEGIN UPGRADE '0.1' TO '0.2';
```

Pluggable database altered.

```
SQL> INSERT INTO basis.bundeslaender (BLID, BUNDESLAND) VALUES (17, 'Mallorca');
```

1 row created.

```
SQL> COMMIT;
```

Commit complete.

```
SQL> ALTER PLUGGABLE DATABASE APPLICATION stammdaten END UPGRADE;
```

Pluggable database altered.

- Und natürlich muss eine neue Synchronisierung erfolgen

```
SQL> ALTER SESSION SET CONTAINER=appweg_flugzeuge;
```

```
Session altered.
```

```
SQL> SELECT count(*) FROM basis.bundeslaender;
```

```
  COUNT (*)  
-----  
         16
```

```
SQL> ALTER PLUGGABLE DATABASE APPLICATION stammdaten SYNC;
```

```
Pluggable database altered.
```

```
SQL> SELECT count(*) FROM basis.bundeslaender;
```

```
  COUNT (*)  
-----  
         17
```

- Alle Objekte der Application müssen gelöscht werden

```
SQL> ALTER PLUGGABLE DATABASE APPLICATION stammdaten BEGIN UNINSTALL;
```

```
Pluggable database altered.
```

```
SQL> DROP USER basis CASCADE;
```

```
User dropped.
```

```
SQL> ALTER PLUGGABLE DATABASE APPLICATION stammdaten END UNINSTALL;
```

```
Pluggable database altered.
```

Entfernen einer Application (2)

- Eine Wiederbenutzung des Applicationnamens ist nicht möglich
→ ORA-65245: application <appname> has been uninstalled

```
SQL> SELECT app_name, app_version, app_status  
2     FROM dba_applications  
3     WHERE app_name = 'STAMMDATEN';
```

APP_NAME	APP_VERSION	APP_STATUS
-----	-----	-----
STAMMDATEN	0.2	UNINSTALLED

- Solange kein Resync erfolgt, sind die Daten weiterhin sichtbar

```
SQL> SELECT count(*) FROM basis.bundeslaender;
```

```
      COUNT (*)  
-----  
           17
```

```
SQL> ALTER PLUGGABLE DATABASE APPLICATION stammdaten SYNC;
```

```
Pluggable database altered.
```

```
SQL> SELECT count(*) FROM basis.bundeslaender;
```

```
SELECT count(*) FROM basis.bundeslaender  
                                *
```

```
ERROR at line 1:
```

```
ORA-00942: table or view does not exist
```

- Es gibt kein „resync“, d.h. Änderungen die außerhalb von „UPGRADE“ gemacht werden, sind nie für andere Container sichtbar

- Was kann mit den Common Objekten gemacht werden?
 - Metadata (Default)
 - Es gibt nur eine Wahrheit; die des Application Root
 - Daten sind erst dann sichtbar, wenn ein SYNC erfolgt
 - Application Container kann keine Daten ändern
 - Data
 - Application Root definiert und kontrolliert Daten
 - Application Container sieht Daten sofort nachdem sie geändert wurden
 - Verhält sich wie ein Synonym über Container hinweg
 - Application Container können keine Daten ändern
 - Extended Data
 - Application Root definiert Daten
 - Application Container können eigene Inhalte haben

- Option SHARING=DATA

```
CREATE TABLE BASIS.BUNDESLAENDER SHARING=DATA (  
    BLID          NUMBER,  
    BUNDESLAND    VARCHAR2(30 CHAR),  
    Constraint PK_BUNDESLAENDER primary key (BLID)  
);
```

- Daten sind sofort sichtbar

```
SQL> ALTER SESSION SET CONTAINER=appweg;

INSERT INTO basis.bundeslaender (BLID, BUNDESLAND) VALUES (17, 'Mallorca');

SQL> COMMIT;

SQL> ALTER SESSION SET CONTAINER=appweg_flugzeuge;

SQL> SELECT count(*) FROM basis.bundeslaender;

  COUNT (*)
-----
         17
```

- Gilt NICHT für DDL

- Was passiert, wenn der Application Container Daten ändern / eingeben will?

```
SQL> ALTER SESSION SET CONTAINER=appweg_flugzeuge;
```

```
Session altered.
```

```
SQL> SELECT count(*) FROM basis.bundeslaender;
```

```
  COUNT (*)  
-----  
         17
```

```
SQL> INSERT INTO basis.bundeslaender (BLID, BUNDESLAND)  
      2  VALUES (18, 'Neufünfland');
```

```
INSERT INTO basis.bundeslaender (BLID, BUNDESLAND)  
                                *
```

```
ERROR at line 1:
```

```
ORA-65097: DML into a data link table is outside an application action
```

- Option SHARING=EXTENDED DATA

```
CREATE TABLE BASIS.BUNDESLAENDER SHARING=EXTENDED DATA
(
    BLID          NUMBER,
    BUNDESLAND    VARCHAR2(30 CHAR),
    Constraint PK_BUNDESLAENDER primary key (BLID)
);
```

- Daten sind sofort sichtbar

```
SQL> ALTER SESSION SET CONTAINER=appweg;

INSERT INTO basis.bundeslaender (BLID, BUNDESLAND) VALUES (17, 'Mallorca');

SQL> COMMIT;

SQL> ALTER SESSION SET CONTAINER=appweg_flugzeuge;

SQL> SELECT count(*) FROM basis.bundeslaender;

  COUNT (*)
-----
         17
```

- PDB kann Daten ändern

```
ALTER SESSION SET CONTAINER=appweg_flugzeuge;
SQL> SELECT count(*) FROM basis.bundeslaender;
COUNT(*)
-----
17

SQL> INSERT INTO basis.bundeslaender (BLID, BUNDESLAND)
2 VALUES (18, 'Neufünfland');
1 row created.
SQL> COMMIT;
SQL> SELECT count(*) FROM basis.bundeslaender;
COUNT(*)
-----
18

SQL> ALTER SESSION SET CONTAINER=appweg;
SQL> SELECT count(*) FROM basis.bundeslaender;
COUNT(*)
-----
17
```

- Was passiert, nachdem die Application gelöscht wurde?

```
SQL> ALTER PLUGGABLE DATABASE APPLICATION stammdaten_extdata BEGIN UNINSTALL;  
SQL> DROP USER BASIS CASCADE;  
SQL> ALTER PLUGGABLE DATABASE APPLICATION stammdaten_extdata END UNINSTALL;
```

```
SQL> ALTER SESSION SET CONTAINER=appweg_flugzeuge;  
SQL> SELECT count(*) FROM basis.bundeslaender;  
SELECT count(*) FROM basis.bundeslaender  
*  
ERROR at line 1:  
ORA-12801: error signaled in parallel query server P000  
ORA-01918: user 'BASIS' does not exist
```

```
SQL> ALTER PLUGGABLE DATABASE APPLICATION stammdaten_extdata SYNC;
```

```
SQL> SELECT count(*) FROM basis.bundeslaender;  
SELECT count(*) FROM basis.bundeslaender  
*  
ERROR at line 1:  
ORA-00942: table or view does not exist
```

- Was passiert, wenn man auf ein Application Objekt eine Änderung durchführen will?

```
SQL> ALTER TABLE basis.bundeslaender  
      ADD bemerkung VARCHAR2 (50);  
ALTER TABLE basis.bundeslaender ADD bemerkung VARCHAR2 (50)  
*  
ERROR at line 1:  
ORA-65274: operation not allowed from outside an application action
```


Welche Methode ist sinnvoll

- Metadata
 - PL/SQL
 - Daten, die sich extrem selten ändern
- Data
 - Stammdatenverwaltung
- Extended Data
 - Mandantenfähigkeit

- Vorlage für Application Container Datenbanken
- Erstellt aus
 - Application Container Root
 - Enthält nur die Daten des Application Root (ohne zusätzlichen SYNC)
 - Application Container
 - Enthält alle Daten des Application Containers (inklusive der Application)
- Bekommt den Namen des Application Container Root + \$SEED
 - Alle zukünftigen Application Container beziehen sich auf die Seed nicht mehr auf den Root

Application Seed aus Root Container

```
SQL> ALTER SESSION SET CONTAINER=appweg;  
Session altered.  
SQL> CREATE PLUGGABLE DATABASE AS SEED  
    2 ADMIN USER pdb_admin IDENTIFIED BY manager DEFAULT TABLESPACE users;  
Pluggable database created.  
SQL> ALTER PLUGGABLE DATABASE appweg$seed OPEN INSTANCES=ALL;  
Pluggable database altered.  
SQL> ALTER SESSION SET CONTAINER=appweg$seed;  
Session altered.  
SQL> ALTER PLUGGABLE DATABASE APPLICATION ALL SYNC;  
Pluggable database altered.  
SQL> ALTER PLUGGABLE DATABASE appweg$seed CLOSE IMMEDIATE INSTANCES=ALL;  
Pluggable database altered.  
SQL> ALTER PLUGGABLE DATABASE appweg$seed OPEN READ ONLY INSTANCES=ALL;  
Pluggable database altered.  
SQL> ALTER PLUGGABLE DATABASE appweg$seed SAVE STATE INSTANCES=ALL;  
Pluggable database altered.
```

```
SQL> ALTER SESSION SET CONTAINER=appweg_flugzeuge;
```

Session altered.

```
SQL> SELECT table_name from dba_tables where owner='DEMO';
```

```
TABLE_NAME
```

```
-----
```

```
ADRESSEN
```

```
PERSONEN
```

```
SQL> ALTER SESSION SET CONTAINER=appweg;
```

Session altered.

```
SQL> CREATE PLUGGABLE DATABASE AS SEED FROM appweg_flugzeuge;
```

Pluggable database created.

```
SQL> ALTER PLUGGABLE DATABASE appweg$seed OPEN INSTANCES=ALL;
```

Pluggable database altered.

Application Seed löschen

```
SQL> ALTER SESSION SET CONTAINER=appweg;
```

Session altered.

```
SQL> ALTER PLUGGABLE DATABASE appweg$seed CLOSE INSTANCES=ALL;
```

Pluggable database altered.

```
SQL> DROP PLUGGABLE DATABASE appweg$seed INCLUDING DATAFILES;
```

Pluggable database dropped.

- Neue Application Container werden aus der Seed erstellt
 - Application Container Root kann geändert werden
 - Gleiche Basis für alle PDBs
 - Keine spezielle Syntax erforderlich
 - Keine Anmeldung an SEED erforderlich

```
SQL> ALTER SESSION SET CONTAINER=appweg;
```

```
Session altered.
```

```
SQL> CREATE PLUGGABLE DATABASE appweg_flugzeuge  
2   ADMIN USER pdb_admin IDENTIFIED BY manager  
3   DEFAULT TABLESPACE users;
```

Application Seed

- Eigenes Template für neue PDBs
- Erstellung nur aus einem Application Root Container heraus
- Mischung von „Application“ und „normalen“ Daten möglich
- Synchronisierung trotzdem mit Application Root

Sonstiges

- Connect nur gegen die CDB möglich
- Ggf. After Logon Trigger mit Redirect auf die PDB
 - ALTER SESSION SET CONTAINER

- → Alle Passwortkomplexität
 - Mind. 8 Zeichen
 - Mind. 1 Buchstabe
 - Mind. 1 Ziffer
 - Mind. 1 special Character (kein „““)
 - Kein Username oder Reverse Username

- Ausführung von SQL Skripten gegen mehrere PDBs und CDB\$ROOT gleichzeitig
 - Kann auch für PDB\$SEED verwendet werden (z.B. für Installation / Deinstallation einer Datenbankoption)
 - Benötigt Perl
 - -c für bestimmte PDB(s)
 - -C nicht für diese PDB(s)

```
$ORACLE_HOME/perl/bin/perl $ORACLE_HOME/rdbms/admin/catcon.pl -c '<PDBNAME1> <PDBNAME2>'  
-l <logverzeichnis> -b <logprefix> <SQL-Script>
```

- Experten mit über 30 Jahren Datenbank Erfahrung
- Spezialisten für
 - Datenbank Administration (Oracle und PostgreSQL)
 - Hochverfügbarkeit (RAC, Data Guard, Replication, etc.)
 - Migrationen (Unicode, PostgreSQL)
 - Performance Optimierung
 - Monitoring (OEM, Foglight, CheckMK, PEM)
- Fernwartung
- Schulung und Workshops
 - PostgreSQL
 - Oracle Multitenant
 - Toad



- E-Mail: johannes.ahrends@carajandb.com
- Homepage: www.carajandb.com
- Adresse:
 - CarajanDB GmbH
Siemensstraße 25
50374 Erftstadt
- Telefon:
 - +49 (22 35) 1 70 91 84
 - +49 (1 70) 4 05 69 36
- Twitter: carajandb
- Facebook: johannes.ahrends
- Blogs:
 - blog.carajandb.com
 - www.toadworld.com