



Oracle 12c Multitenant Database Braucht man das wirklich?

Johannes Ahrends
CarajanDB GmbH



DOAG
Deutsche ORACLE-Anwendergruppe e.V.

- Experten mit über 20 Jahren Oracle Erfahrung
- Firmensitz in Erftstadt bei Köln
- Spezialisten für
 - Oracle Datenbank Administration
 - Hochverfügbarkeit (RAC, Data Guard, Failsafe, etc)
 - Einsatz der Oracle Standard Edition
 - Oracle Migrationen (HW, Unicode, Konsolidierung, Standard Edition)
 - Replikation (Goldengate, SharePlex, Dbvisit)
 - Performance Tuning
- Schulung und Workshops (Oracle, Toad)



- **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
- **Leiter der DOAG SIG Database**
- **Hobbies:**
 - Drachen steigen lassen (Kiting) draußen wie drinnen (Indoorkiting)
 - Motorradfahren (nur draußen)



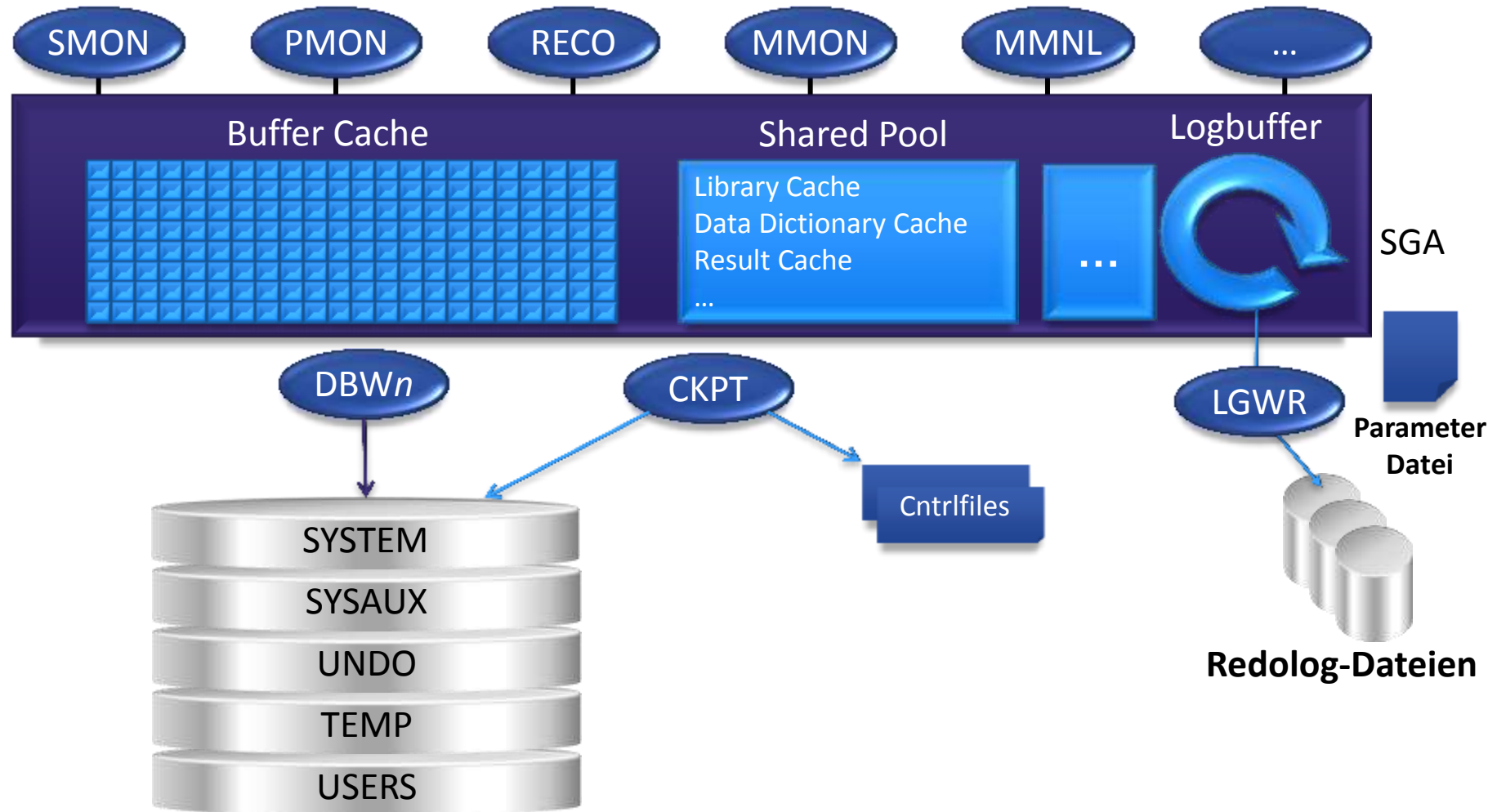
- E-Mail: johannes.ahrends@carajandb.com
- Homepage: www.carajandb.com
- Adresse:
 - CarajanDB GmbH
Siemensstraße 25
50374 Erftstadt
- Telefon:
 - +49 (22 35) 1 70 91 84
 - +49 (1 70) 4 05 69 36
- Twitter: streetkiter
- Facebook: johannes.ahrends
- Blogs:
 - www.carajandb.com/blogs
 - www.toadworld.com



Pluggable Database

- **Konsolidierung von Datenbanken (z.B: Exadata, Database Appliance)**
 - Viele Datenbanken auf einem Server
 - Hochverfügbarkeit durch RAC
 - Optional Data Guard für DR
- **Virtualisierung von Servern (OracleVM, VMware, Zones, ...)**
 - n virtuelle Server pro physikalischem Server
 - Je Datenbank ein eigener virtueller Server
 - Hochverfügbarkeit durch vMotion o.Ä.
 - Optional RAC und Data Guard

„Klassische“ Datenbank

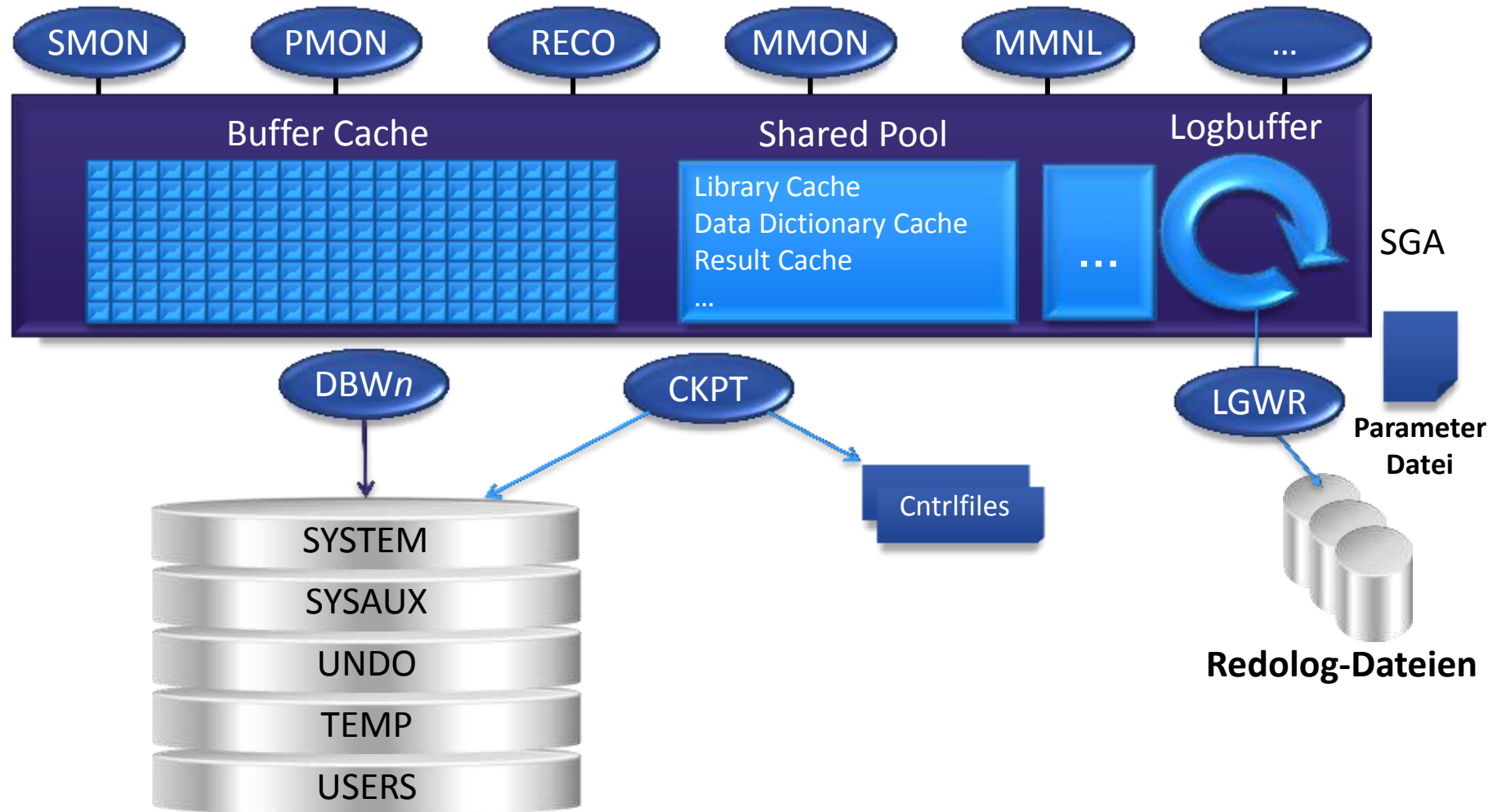


- **Datenbank:**
 - SYSTEM, SYSAUX, UNDO, TEMP Tablespace
 - Redolog Dateien
 - Archivierte Redolog Dateien
 - Controlfile
- **Instanz:**
 - SGA
 - PGA
 - Background Prozesse (einfache DB → 33 Prozesse)

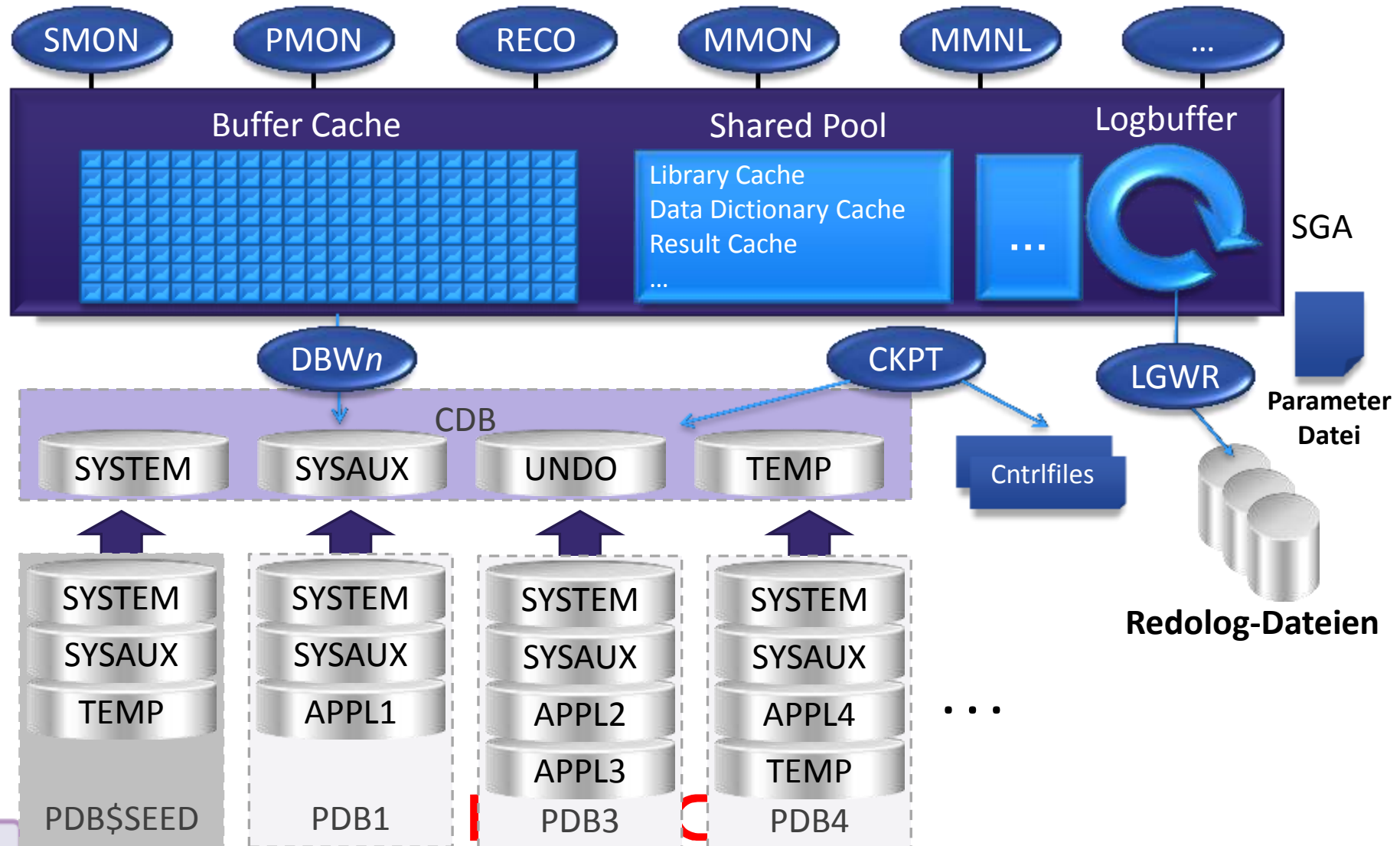
- Minimum 2 GB Memory pro Instanz
- Minimum 2 CPUs pro Instanz
- Tom Kyte:
„the only reasonable, correct number of instances on a production machine is one”

- Konsolidierung von Datenbanken
- Eine gemeinsame Instanz
- Ein gemeinsames Undo Management
- Ein gemeinsames Redo Management
- Separate Tablespace Verwaltung
- Separate Schema Verwaltung

„Klassische“ Datenbank



Multitenant Database



- **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 (1)

Database Configuration Assistant - Create Database - Step 2 of 5

ORACLE 12c DATABASE

Creation Mode

Database Operation
Creation Mode
Pre Requisite Checks
Summary
Progress Page

Create a database with default configuration

Global Database Name: CDBACFS

Storage Type: File System

Database Files Location: /u02/oradata Browse...

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

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

Administrative Password:

Confirm Password:

☒ Create As Container Database

Pluggable Database Name: PLUGACFS1

☐ Advanced Mode

Help < Back Next > Finish Cancel

Erstellen einer CDB (2)

- Alle Optionen werden installiert!

Database Configuration Assistant - Create Database - Step 4 of 5

Summary

Database Configuration Assistant: Summary

Memory Configuration Type: Automatic Memory Management
Template Name: General Purpose or Transaction Processing

Database Configuration Details

Database Components

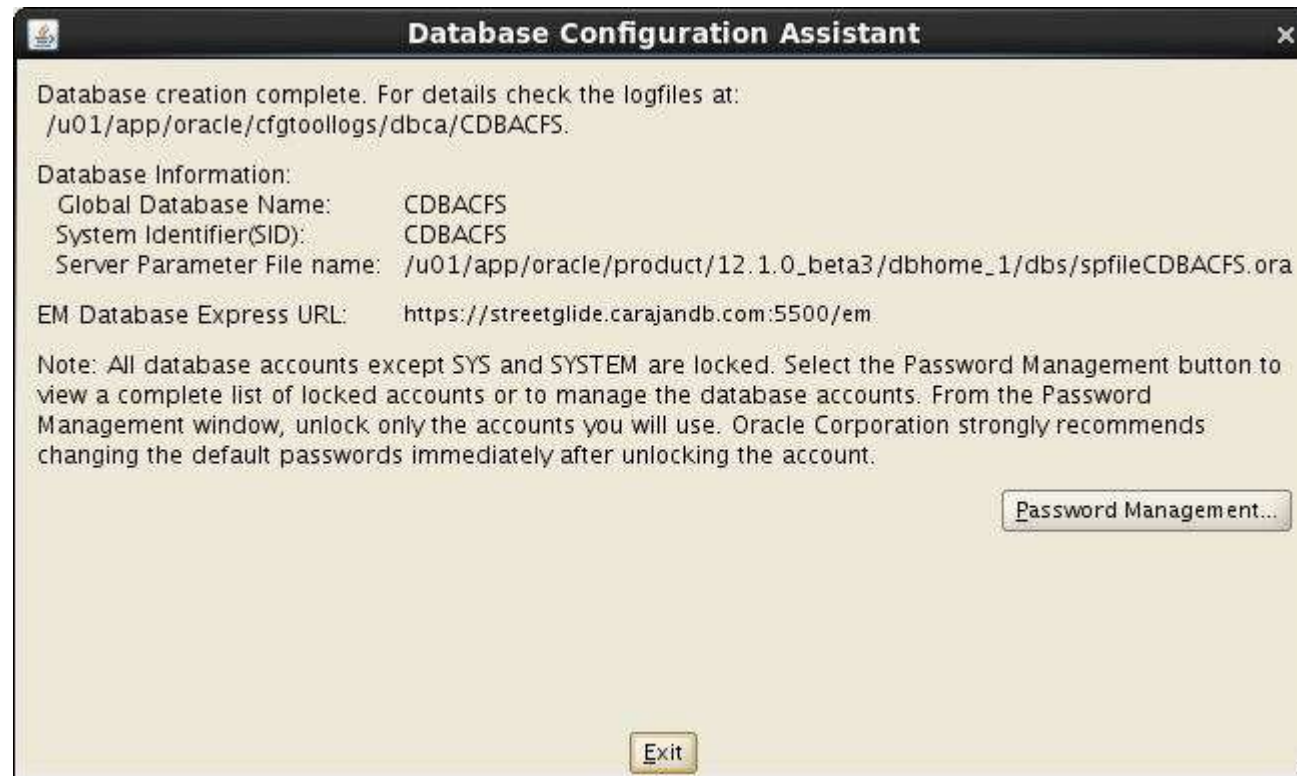
Component	Selected
Oracle JVM	true
Oracle Text	true
Oracle Multimedia	true
Oracle OLAP	true
Oracle Spatial	true
Oracle Label Security	true
Oracle Application Express	true
Oracle Database Vault	true

Initialization Parameters

Name	Value
audit_file_dest	{ORACLE_BASE}/admin/{DB_UNIQUE_NAME}/adump
audit_trail	db
compatible	12.0.0.0.0
db_block_size	8KB
db_create_file_dest	/u02/oradata
db_domain	
db_name	CDBACFS
db_recovery_file_dest	/u03/orabackup

Help < Back Next > Finish Cancel

Erstellen einer CDB (3)



- **V\$PDBS** → welche PDBs habe ich (in CDB)

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

CON_ID	NAME	OPEN_MODE
2	PDB\$SEED	READ ONLY
3	PLUGJA21	READ WRITE
4	PLUGJA22	MOUNTED
5	PJA12CD1	MOUNTED

- **CDB_PDBS → Infos über PDBs**

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

PDB_ID	PDB_NAME	GUID	STATUS	CON_ID
2	PDB\$SEED	D755CB386500614EE043CD9CA8C0E494	NORMAL	1
3	PLUGJA21	D7586649F163701DE043CD9CA8C0939E	NORMAL	1
4	PLUGJA22	D75871042DE67064E043CD9CA8C0FC7F	UNPLUGGED	1
5	PJA12CD1	D758AC64E992747FE043CD9CA8C0F054	UNPLUGGED	1

- **Keine einzelne View liefert beides (Status und Open_mode)**

Tablespaces

SQL Editor:

```
1 SELECT * FROM cdb_data_files
2
3 ORDER BY
```

Data Grid:

TABLESPACE_NAME	FILE_NAME	SIZE_MB	ONLINE_STATUS	CON_ID
SYSTEM	+DATA/CJA12C/DATAFILE/system.265.809443493	780	SYSTEM	1
SYSAUX	+DATA/CJA12C/DATAFILE/sysaux.267.809443577	850	ONLINE	1
UNDOTBS1	+DATA/CJA12C/DATAFILE/undotbs1.269.809443637	695	ONLINE	1
USERS	+DATA/CJA12C/DATAFILE/users.272.809443721	5	ONLINE	1
SYSAUX	+DATA/CJA12C/D755CB386500614EE043CD9CA8C0E494/DATAFILE/sysaux.268.809443621	705	OFFLINE	2
SYSTEM	+DATA/CJA12C/D755CB386500614EE043CD9CA8C0E494/DATAFILE/system.266.809443555	260	SYSOFF	2
SYSTEM	+DATA/CJA12C/D7586649F163701DE043CD9CA8C0939E/DATAFILE/system.274.809454689	260	SYSTEM	3
USERS	+DATA/CJA12C/D7586649F163701DE043CD9CA8C0939E/DATAFILE/users.277.809454861	5	ONLINE	3
SYSAUX	+DATA/CJA12C/D7586649F163701DE043CD9CA8C0939E/DATAFILE/sysaux.275.809454689	715	ONLINE	3
SYSTEM	+DATA/CJA12C/D75871042DE67064E043CD9CA8C0FC7F/DATAFILE/system.279.809454869	260	SYSTEM	4
SYSAUX	+DATA/CJA12C/D75871042DE67064E043CD9CA8C0FC7F/DATAFILE/sysaux.278.809454865	705	ONLINE	4

Anmelden an die PDB (1)

- **Indirektes Anmelden über die Container Datenbank:**

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

- **„Normale“ Anmeldung an die PDB nur über entsprechenden Servicennamen**

```
$ sqlplus username/password@SERVICENAME
```

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 ja12c -pdb pdb1_ja12c  
    -service pdb1ja12c  
$ srvctl start service -db ja12c -service pdb1ja12c
```

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

Service Erstellung für die PDB (2)

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

```
SQL> ALTER SESSION SET CONTAINER = pdb1_ja12c;
```

Session wurde geändert.

```
SQL> EXECUTE  
      dbms_service.create_service('PDB1JA12C','PDB1JA12C.DB.DE');
```

PL/SQL-Prozedur erfolgreich abgeschlossen.

```
SQL> EXECUTE dbms_service.start_service('PDB1JA12C');
```

PL/SQL-Prozedur erfolgreich abgeschlossen.

- **Zusätzlicher Befehl**
 - SQL> ALTER PLUGGABLE DATABASE ...
- **Pluggable Databases werden nicht automatisch gestartet:**
 - SQL> STARTUP PLUGGABLE DATABASE pdb1
 - SQL> STARTUP PLUGGABLE DATABASE ALL
- **Öffnen und Schließen von Pluggable Databases**
 - SQL> ALTER PLUGGABLE DATABASE ... OPEN
 - SQL> ALTER PLUGGABLE 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)

```
SQL> ALTER PLUGGABLE DATABASE plugja22  
      UNPLUG INTO '/tmp/plugja22.xml';
```


- **Kopieren einer Pluggable Database in der gleichen CDB**

```
SQL> CREATE PLUGGABLE DATABASE plugja23  
      FROM plugja21;
```

- **Einhängen einer Pluggable Database in eine CDB**

```
SQL> CREATE PLUGGABLE DATABASE plugja22  
      USING '/tmp/plugja22.xml';
```

- **Funktioniert nur, wenn die PDB noch nicht im DD existiert, ansonsten erst löschen (Default KEEP DATAFILES):**

```
SQL> DROP PLUGGABLE DATABASE plugja22;
```

- **Services werden nicht mit gelöscht!**

- **Backup Container Datenbank**

```
RMAN> BACKUP DATABASE ROOT;
```

- **Backup einer Pluggable Database;**

```
RMAN> BACKUP PLUGGABLE DATABASE plugja21;
```

- Nur eine SGA → Keine Einschränkung hinsichtlich Buffer-Cache, Shared-Pool, etc.
- Nur ein Set an Redologs und Undo-Tablespace
- Keine Einschränkung bezüglich I/O
- Ressourceneinschränkungen nur über den Resource Manager (z.B: CPU, Sessions, execution Time)
- Einschränkung der Gesamtgröße einer PDB, z.B.
 - `ALTER PLUGGABLE DATABASE
STORAGE (MAXSIZE xG) ;`
- **Eigener TEMP-TABLESPACE**

- 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

NAME	VALUE
-----	-----
cursor_sharing	EXACT
db_create_file_dest	/u02/oradata
db_create_online_log_dest_1	
db_file_multiblock_read_count	128
db_securefile	PREFERRED
deferred_segment_creation	TRUE
global_names	FALSE
log_archive_dest_1...	
log_archive_dest_state_1...	enable
log_archive_min_succeed_dest	1
open_cursors	300
optimizer_features_enable	12.1.0.1
optimizer_index_caching	0
optimizer_index_cost_adj	100
optimizer_mode	ALL_ROWS
optimizer_use_invisible_indexes	FALSE
optimizer_use_pending_statistics	FALSE
optimizer_use_sql_plan_baselines	TRUE
pdb_file_name_convert	
sessions	472
skip_unusable_indexes	TRUE
sort_area_size	65536
statistics_level	TYPICAL
timed_os_statistics	0
timed_statistics	TRUE
workarea_size_policy	AUTO

BUG!

Wo lohnt sich die Multitenant Option?



- **Aufbau von Test- bzw. Development-Datenbanken**
 - Einfache Administration
 - Schnelles Cloning (ev. Snapshot Clone)
 - Schnelles Patchen (alle auf einmal)
 - Lokale Administratoren möglich
 - Snapshots von bestimmten Ständen können einfach erstellt bzw. wieder genutzt werden

Wo lohnt sich die Multitenant Option?



- **Database As A Service**
 - Pluggable Databases sind unabhängig von RAC oder Data Guard
 - Wechsel des Servicelevels durch einfaches Unplug / Create
 - Lokale Administratoren
 - Keine Lizenzproblematiken (alles ist bereits lizenziert)

- **Vorteile:**

- Weniger Downtime bei Patch und Upgrades weil Data Dictionary unabhängig von den Daten
- Einfaches Clonen (bei ACFS und Direct NFS über Snapshots)
- Weniger Ressourcenverbrauch
- Einfache Migration von Single Instance zu RAC

- **Nachteile:**

- Neue Struktur für den DBA
- Keine echte Ressourcentrennung zwischen PDBs
- Hohe Lizenzkosten
- Nicht für Standard Edition (Single Tendance erlaubt)



Fragen?

Johannes Ahrends

www.carajandb.com

Johannes.ahrends@carajandb.com



DOAG
Deutsche ORACLE-Anwendergruppe e.V.

- 10. September Toad User Group in Köln
- 17. September Toad User Group in München
- 18. September DOAG SIG Database in Hamburg (Thema Replikation)
- 07. – 08. Oktober Berliner Expertenseminar „Oracle Multitenant Database“
- 18. – 20. November DOAG Konferenz und Ausstellung in Nürnberg
- 21. November DOAG Schulungstag – Toad für den DBA