



Multitenant in der Praxis

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
- Hobbies:
 - Drachen steigen lassen (Kiting) draußen wie drinnen (Indoorkiting)
 - Motorradfahren (nur draußen)



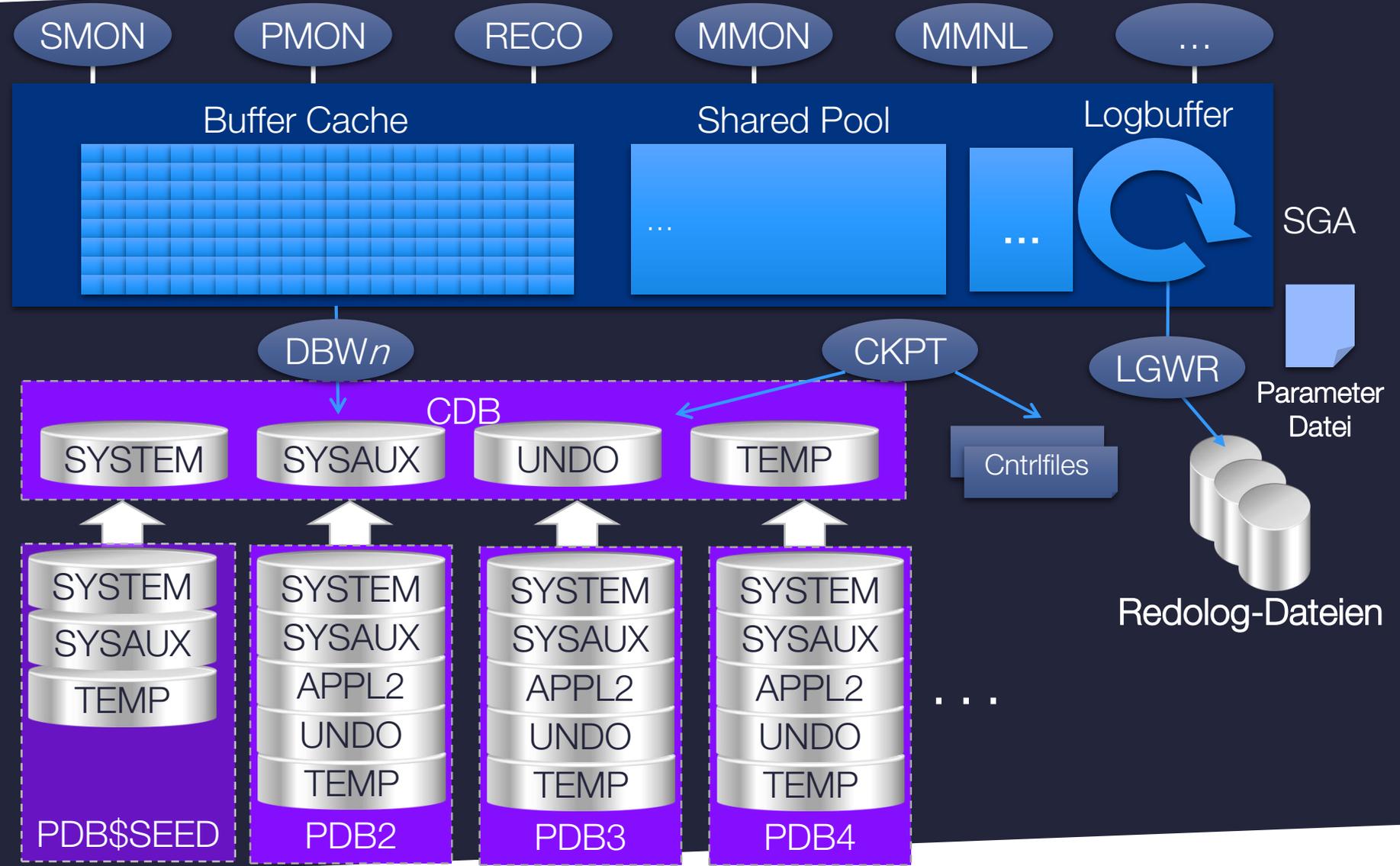


Multitenant Database

- 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 253 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 2. Oracle recommends use of the CDB architecture.

Multitenant Database 12.2



- Eigener Undo-Tablespace möglich
 - Flashback Pluggable Database
- Unterschiedliche Zeichensätze in einer CDB
 - CDB muss aber AL32UTF8 sein
- Ressourcenbegrenzung
- Clonen von PDBs im laufenden Betrieb
- Application Container und PDBs
- Relocate PDB
- Refreshable PDB
- Proxy PDB
- 4096 PDBs pro CDB (Nur Exadata!)
- MAX_PDBS Parameter

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 43-7 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 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	FILE_NAME
2	PDB\$SEED	6	SYSAUX	/disk1/oracle/dbs/pdbseed/cdb1_ax.f
2	PDB\$SEED	5	SYSTEM	/disk1/oracle/dbs/pdbseed/cdb1_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

- CDB_-Views ignorieren die SEED PDB
- Mike Dietrich
(https://blogs.oracle.com/UPGRADE/entry/new_undocumented_parameters_in_oracle):
 - EXCLUDE_SEED_CDB_VIEW
 - Undocumented
 - Per feedback by the Multitenant team:
 - Default: TRUE
 - Setting this parameter to FALSE would return results for the seed database when querying against the CDB views

Das hier sollte „richtig“ sein!

```
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;
```

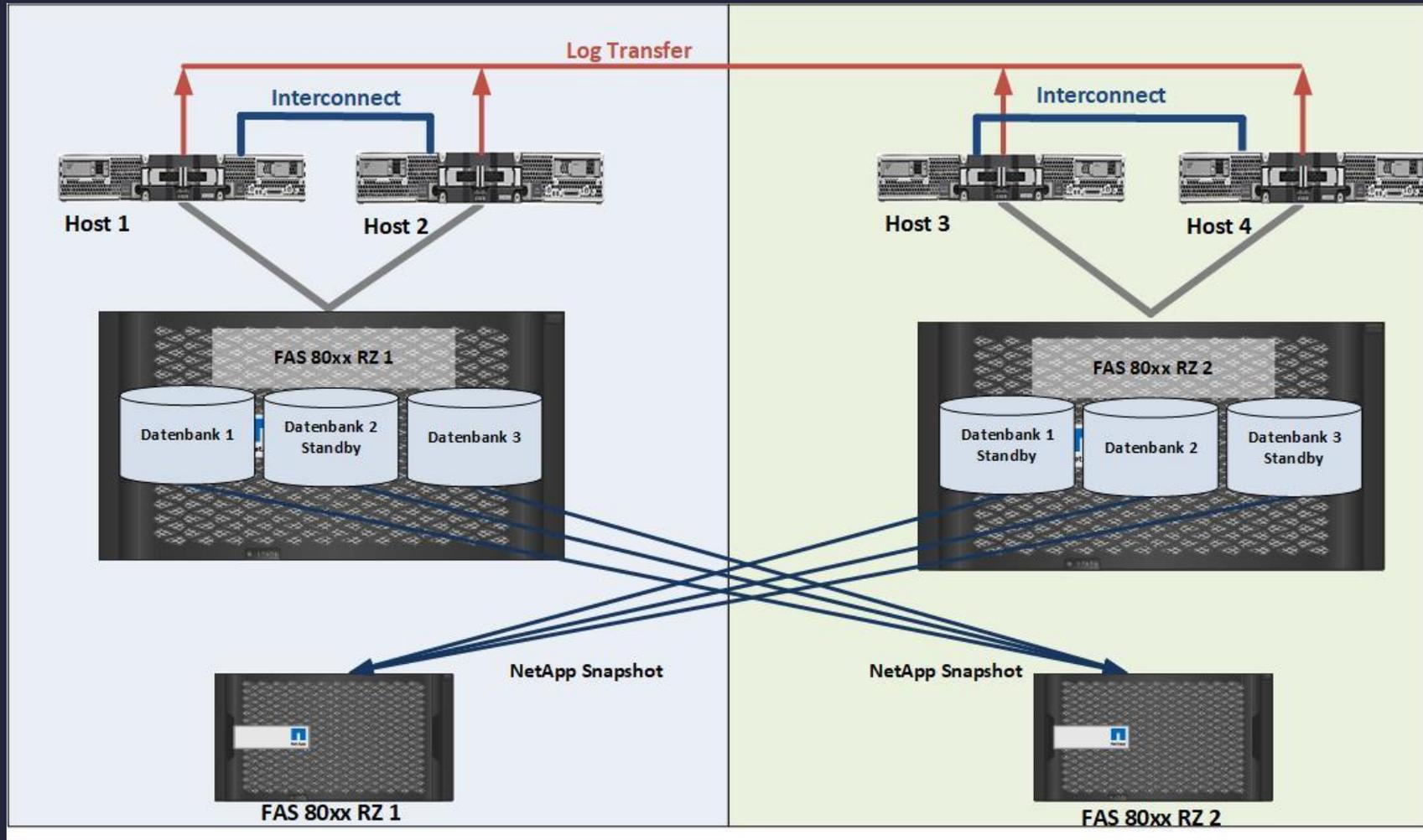
CON_ID	TABLESPACE_NAME	MBYTES	MAXMBYTES
1	USERS	5	32767
1	UNDOTBS1	920	32767
1	SYSTEM	800	32767
1	SYSAUX	870	32767
2	SYSTEM	250	32767
2	SYSAUX	550	32767
3	USERS	100	5120
3	SYSAUX	600	32767
3	SYSTEM	260	32767
4	USERS	300	1024
4	SYSTEM	260	32767
4	SYSAUX	600	32767



Das Projekt

- Migration von 8 Oracle 11g Datenbank 6 Knoten Stretched RAC nach
 - Oracle 12c (12.1.0.2)
 - 2 Knoten RAC
 - Data Guard
 - Multitenant
 - 4 Stages (Maintenance, Test, Vorproduktion, Produktion)
 - 2 Projekte
- Start des Projektes Migration August 2015
- Start des Projektes Multitenant Januar 2016

Schematische Darstellung



- Ursprünglich: Schemakonsolidierung auf 8 Datenbanken
- Minimierung der Downtime für Patching und Upgrades
- Flexible Patchzeiten
- Jede Anwendung kann eigene Publik Synonyme haben
- Mehrere Versionen einer Anwendung in Produktion (Mandanten)
- DBA Berechtigungen
- Zurücksetzen (Flashback) einzelner Anwendungen

- Oracle 12c (12.1.0.2)
- 2 Knoten RAC
- NetApp Storage mit Direct NFS
- Data Guard
- 2 Datenbanken
- Projektstart Oktober 2015

- Unkomplizierte Migration (Data Pump)
- RAC erweist sich als sehr stabil
 - Rolling Upgrades mit Oracle 12c möglich
- Data Guard erweist sich als sehr stabil
 - Jährliche Stromabschaltung im RZ ohne Probleme (... der Datenbank)
- Migration vorzeitig abgeschlossen!
- Projektende März 2017
- Hohe Akzeptanz im Unternehmen

- Oracle 12c (12.1.0.2)
- 2 Knoten RAC
- NetApp Storage mit Direct NFS
- Data Guard
- 6 Datenbanken (CDBs)
- Projektstart Januar 2016

- Warum 6 Datenbanken?
 - Je Competence Center (3) eine eigene CDB
 - Kritisches Dokumenten Management System alleine
 - Eine CDB für Java Anwendungen (Patching Problematik)
 - Eine CDB für Unicode (derzeit default noch WE8ISO)



New Paradigms for Rapid Patching and Upgrades

The investment of time and effort to patch one multi-tenant container database results in patching all of its many pluggable databases. To patch a single pluggable database, you simply unplug/plug to a multi-tenant 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](#).

ALTERNATIVE FAKTEN

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

- Prinzipiell: Out-Of-Place Patching
 - Datenbank kann im laufenden Betrieb gepatcht werden (kein Upgrade Modus)
 - Ausnahme: OJVM nur mit Einschränkung
- Konkretes Projekt
 - RAC Rolling Upgrade
 - Data Guard first

- Anwendungen greifen auf Daten in anderen Schemata zu
 - Durch Trennung der Anwendungen müssen jetzt Datenbank Links verwendet werden
 - Grants auf Objekte über Datenbank-Links sind kritisch
- Es dürfen nicht die Default Services verwendet werden
- Kein Zugriff über „/ as sysdba“ möglich

- Data Guard unterstützt Multitenant nur rudimentär!!
 - CREATE FROM SEED → Funktioniert
 - CREATE FROM PDB → Funktioniert nicht
 - Clone aus anderer CDB → Funktioniert nicht
 - Oracle Managed Files → db_unique_name der Primary für die Standby Datenbank
- Lösung: Erstellen einer PDB darf nur über ein Skript erfolgen!
- Keine Verwendung von Oracle Managed Files
- Mit Oracle 18 soll das Problem gelöst sein

- Common User
 - Alle DBAs als Common User angelegt
 - Teilweise in PDBs als expired angezeigt – Login trotzdem möglich
 - Wichtig: CONTAINER_DATA=ALL
- Common Roles
 - Funktionieren gut
 - Nicht vergessen: CONTAINER=ALL (!)
- Common Profiles
 - Funktionieren gut
 - DEFAULT Profile muss in jeder PDB angepasst werden!

- Eine Applikations-DBAs
- Mandantenfähigkeit
- Ressourcen besser genutzt (vor allem in den Testsystemen)

- Anzahl PDBs
 - Test: 73
 - Vorproduktion: 44
 - Produktion: 41
 - Insgesamt: 158

- Upgrade 12.2 bis Mai 2018
- Migration Dokumenten Management System August 2018

- Oracle 12.1: PDB muss read-only geöffnet sein
 - Alternative über Standby Datenbank
 - Skript funktioniert
- Bisher nicht eingesetzt
- Gründe:
 - Kopien zwischen Stages sind sehr aufwändig bzw. nicht erlaubt (Prod – Test)
 - Kopie der Datafiles von Standby → Primary → Standby sind sehr zeitintensiv
 - Bisher zu selten angefordert
- Anforderung kommt mit einer Anwendung, die im Juni migriert werden soll
 - Dann hoffentlich 12.2 → Clone bei geöffneter PDB
- Snapshot Cloning nur mit Active Data Guard empfohlen!

- In Oracle 12.1 nicht möglich
- Alternative
 - Flashback Database über Standby Database
 - Clone CDB über SnapCenter
 - Zusätzlicher Zeitaufwand für das Kopieren der PDB in die originale CDB
- Wichtiges Feature für die Umstellung auf 12.2

- In Multitenant muss man sich „eindenken“
- Multitenant ist eine wirkliche Datenbank-Konsolidierung
 - Verbraucht weniger Ressourcen
 - Ist einfacher zu verwalten (Patch / Upgrade)
- Multitenant ist einfach zu implementieren
- Leider weiß man bei Oracle nicht überall, was Multitenant ist!!!
 - Probleme bei Data Guard
 - Support oft nicht im Bilde
- „Der Appetit kommt mit dem Essen“

DOAG Events

DOAG 2018 Datenbank

- 14. & 15. Mai 2018
- Van der Valk Airport- Hotel, Düsseldorf



DOAG Events

HA Day Hochverfügbarkeit

- 13. September 2018
- Junges Hotel Hamburg

- Experten mit über 25 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, Standard Edition)
 - Replikation
 - Performance Tuning
 - Datenbank Cloning (Delphix, Actifio, CloneDB)
- Fernwartung
- Schulung und Workshops (Oracle, 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](https://twitter.com/carajandb)
- Facebook: [johannes.ahrends](https://www.facebook.com/johannes.ahrends)
- Blogs:
 - blog.carajandb.com
 - www.toadworld.com