

# ERFAHRUNGEN MIT DER MIGRATION NACH 19C MULTITENANT

Andre Lünsmann (Barmenia Krankenversicherung AG)

Johannes Ahrends (CarajanDB GmbH)

18.11.2020



**Barmenia**  
EINFACH. MENSCHLICH.

---

Vorstellung Barmenia, CarajanDB

Derzeitige Oracle Architektur

- VMware mit einer oder mehrerer DB pro Anwendung pro Guest
- Actifio für Backup und Hochverfügbarkeit
- Toolset der DB Administration
- Automatischer Prozesse (Install, Patching)
- Shell Skripting

# 1

**Vorstellung Barmenia, CarajanDB**

**Derzeitige Oracle Architektur**

**Multitanent Datenbank**

# Johannes Ahrends

**Barmenia**  
EINFACH. MENSCHLICH.

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)
- Singen (überall)



Vers.-Fachwirt und Teamleiter Produktionssteuerung und Datenbanken

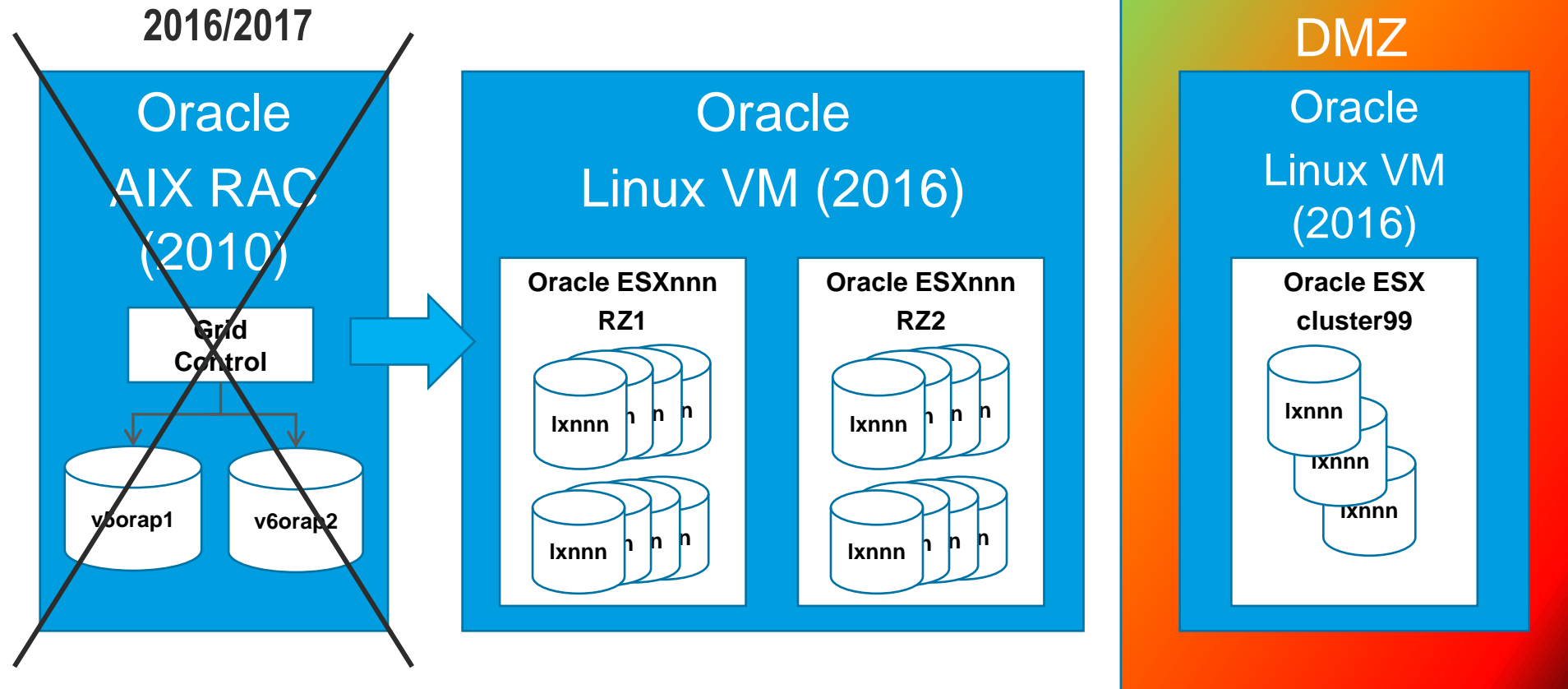
- 1980 Barmenia Versicherungen
- 1988/9: IT Ausbildung
- 1991: Projektleiter in diversen IT Projekten
- 2002: Verantwortlicher für Datenbanken
- 2013/4: Projektleiter Mainframe Ablösung
- 2015 Mitglied im DOAG Anwenderbeirat
- 2020 DOAG Vorstand Infrastruktur & Middleware

Hobbies:

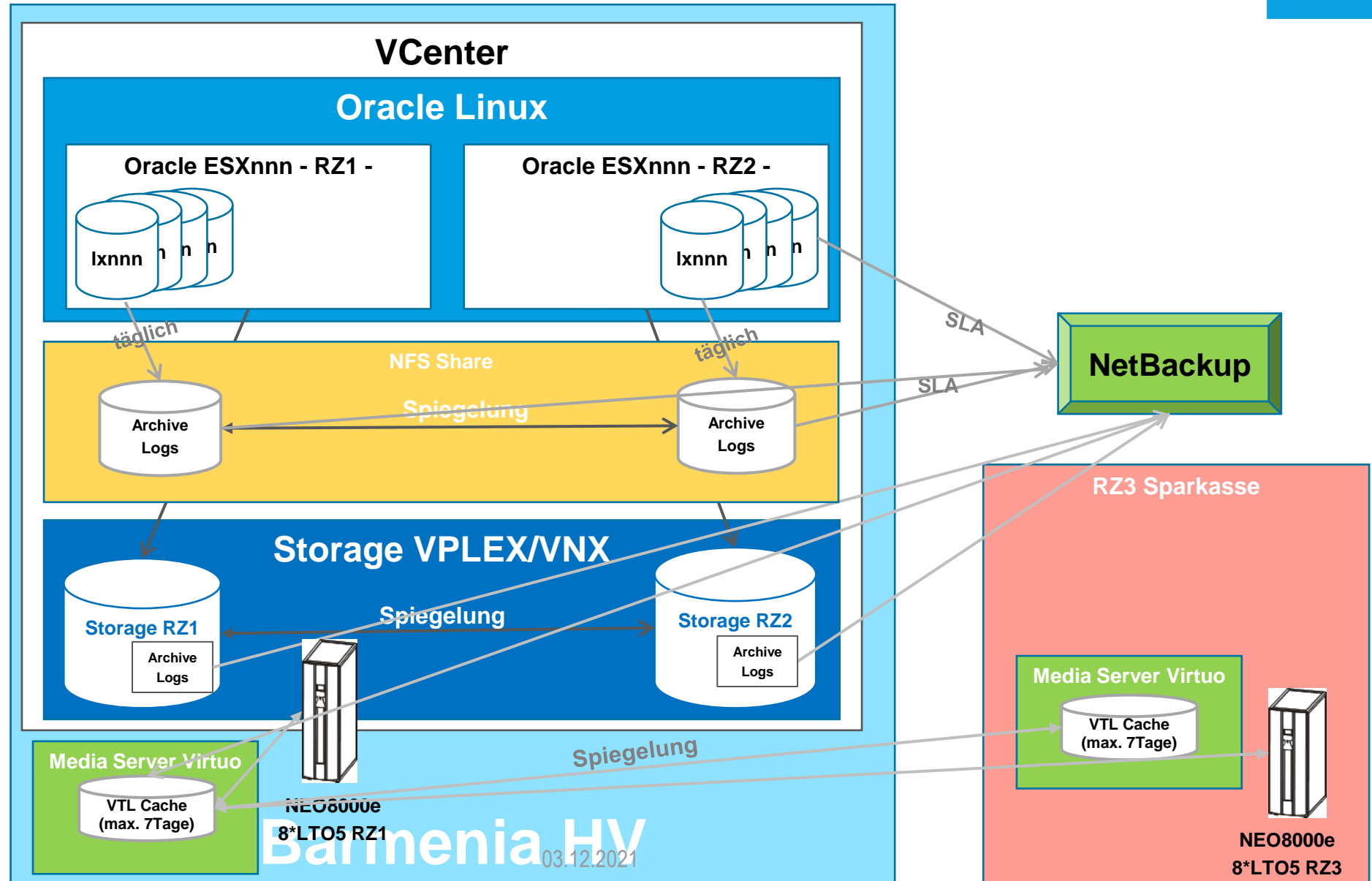
- Tischtennis (nur drinnen)
- Radfahren (nur draußen)
- DOAG (überall)



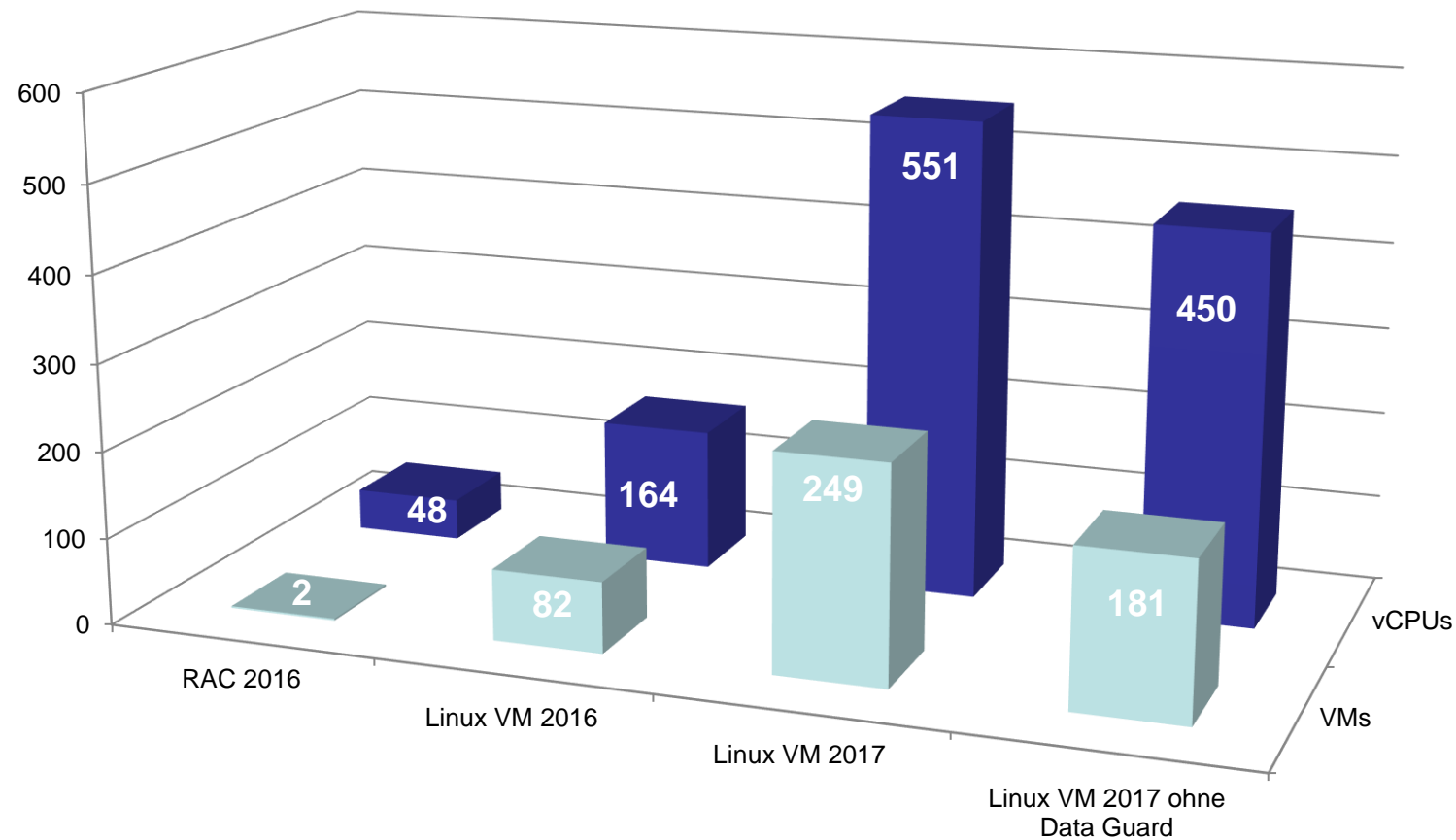
# Übersicht Oracle Datenbank Infrastruktur



# Linux Oracle VM Infrastruktur (2017)



# Beim Tausch NetBackup gegen actifio erfolgte der Verzicht auf Data Guard



## Lizenzvergleich

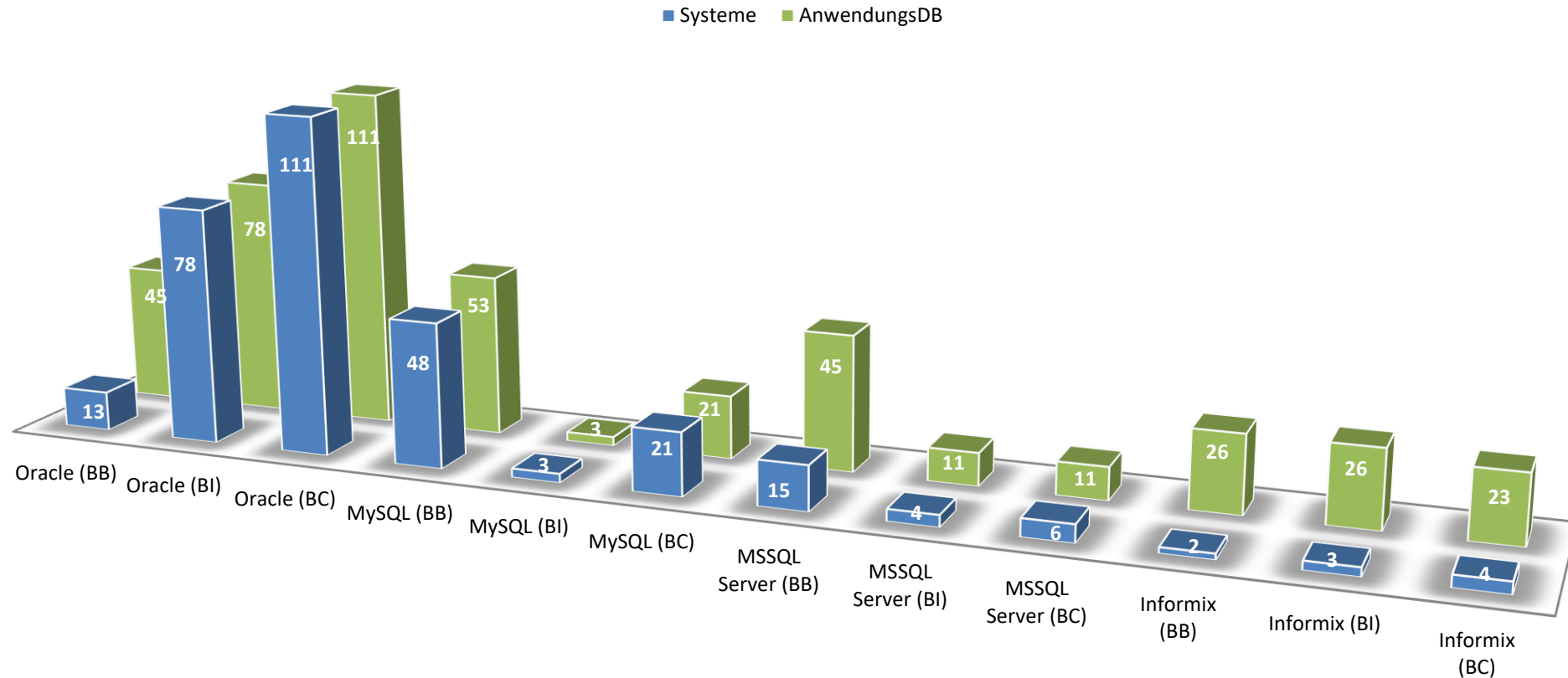
AIX 48 CPU = Linux 68,5 CPU

Durch die vCPU und sharen der Infrastruktur in Vmware genügen im Linux Umfeld 120 lizenzpflichtige CPUs, die für den HA Fall nur ca. 50 % ausgelastet werden.

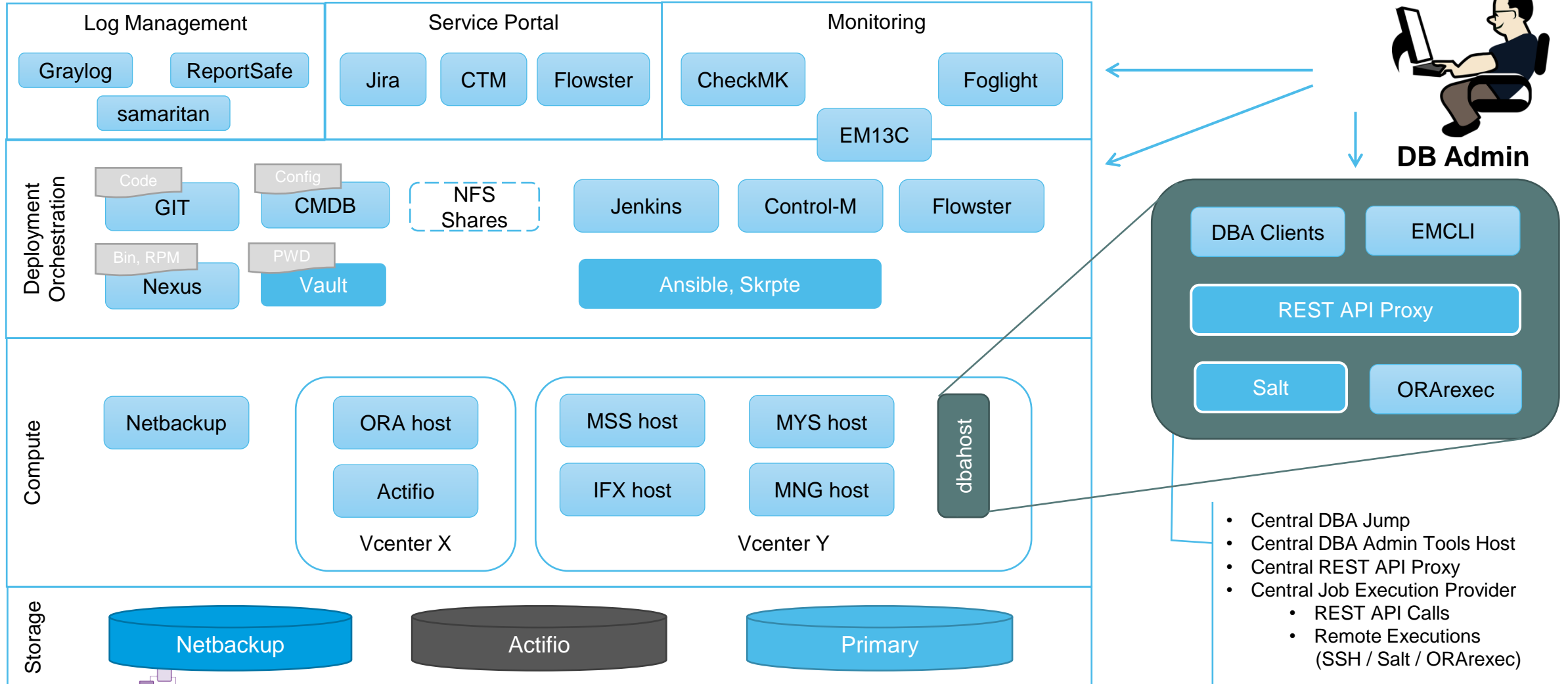


# Überblick der Datenbank Landschaft

## Datenbanken über alle RDBMS



# Toolset für DB Administration



- Central DBA Jump
- Central DBA Admin Tools Host
- Central REST API Proxy
- Central Job Execution Provider
  - REST API Calls
  - Remote Executions (SSH / Salt / ORArexec)

## Start Workflow Deploy Oracle

alles was mit ORADeploy bewerkstelligt werden kann (http://wiki.uniblen/wiki/root/du/wiki/dokumentationen/duku.php?id=databankensackradeployfmanuaLzu\_o radeploy)



### Nebenparameter

Nebenparameter von ORADeploy

Version	19000_190716 ▾	⊗
DB_Name	<input type="text"/>	⊗
DGSDIE	<input type="text"/>	⊗
LOGVOLSIZE	<input type="text"/>	⊗
ADDFRASE	<input type="text"/>	⊗
CHARSET	westeurop. Zeichensatz mit Eurozeichen ▾	⊗
NATCHARSET	AL16UTF16 ▾	⊗
Compatibility	<input type="text"/>	⊗
DBPORT	<input type="text"/>	⊗
Tem plate	BARM_122_cdb.cbt ▾	⊗
DG52HOST	<input type="text"/>	⊗
RCVFDSIZE	<input type="text"/>	⊗
NONDEFID	<input type="text"/>	⊗
ActioSLA	kein Actio- Backup ▾	⊗

Kommentar:

Einfach einplanen

[Wiederkehrend einplanen](#)

☐ Garantierte Ausführung Bitte wählen sie diese Option, wenn der Workflow nach einem Systemfehler sofort ausgeführt werden soll.

Ausführungszeit:

Schließen

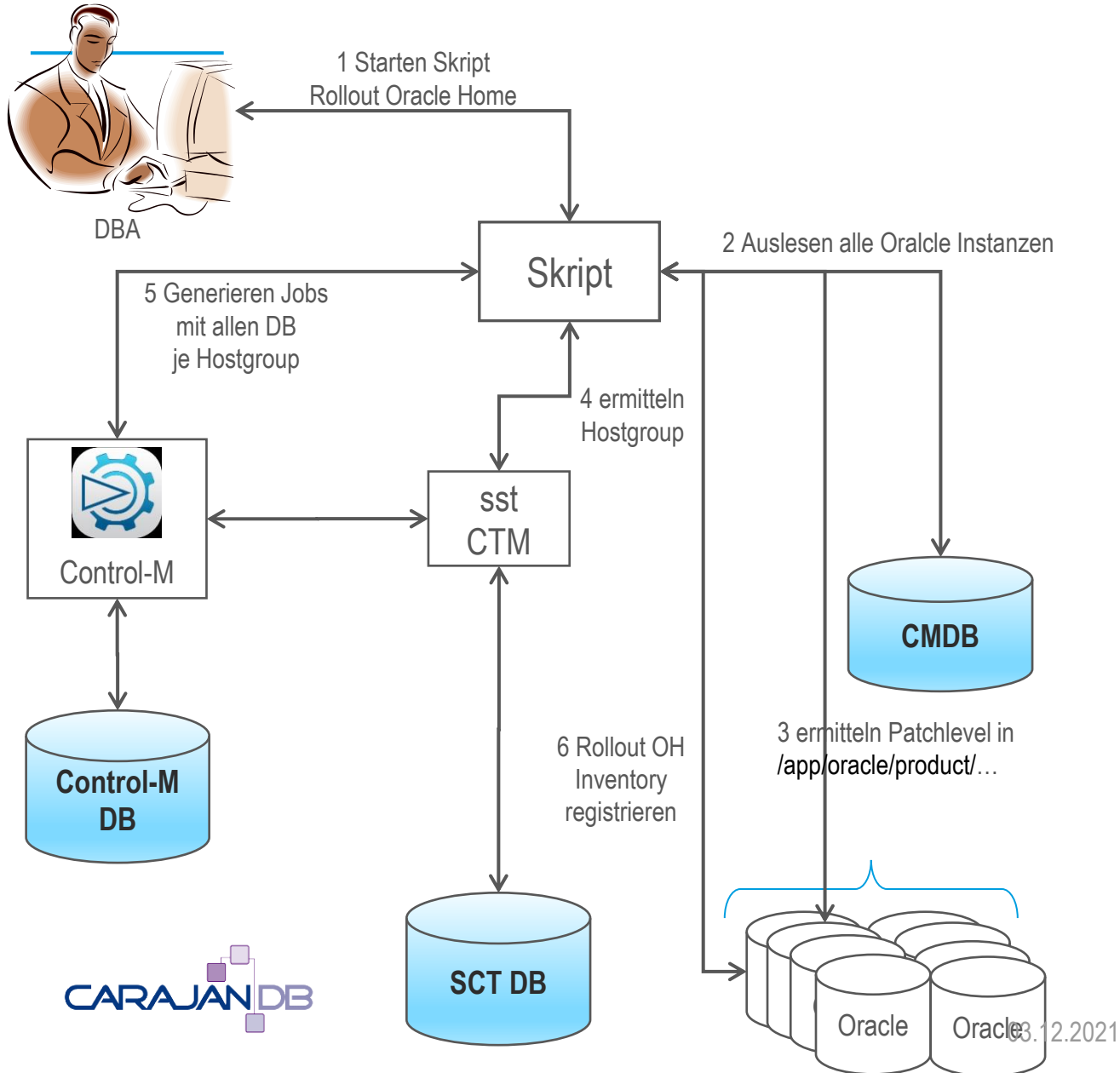
Zurück

Ausführen

CTM  
s Prüfung mit mind.

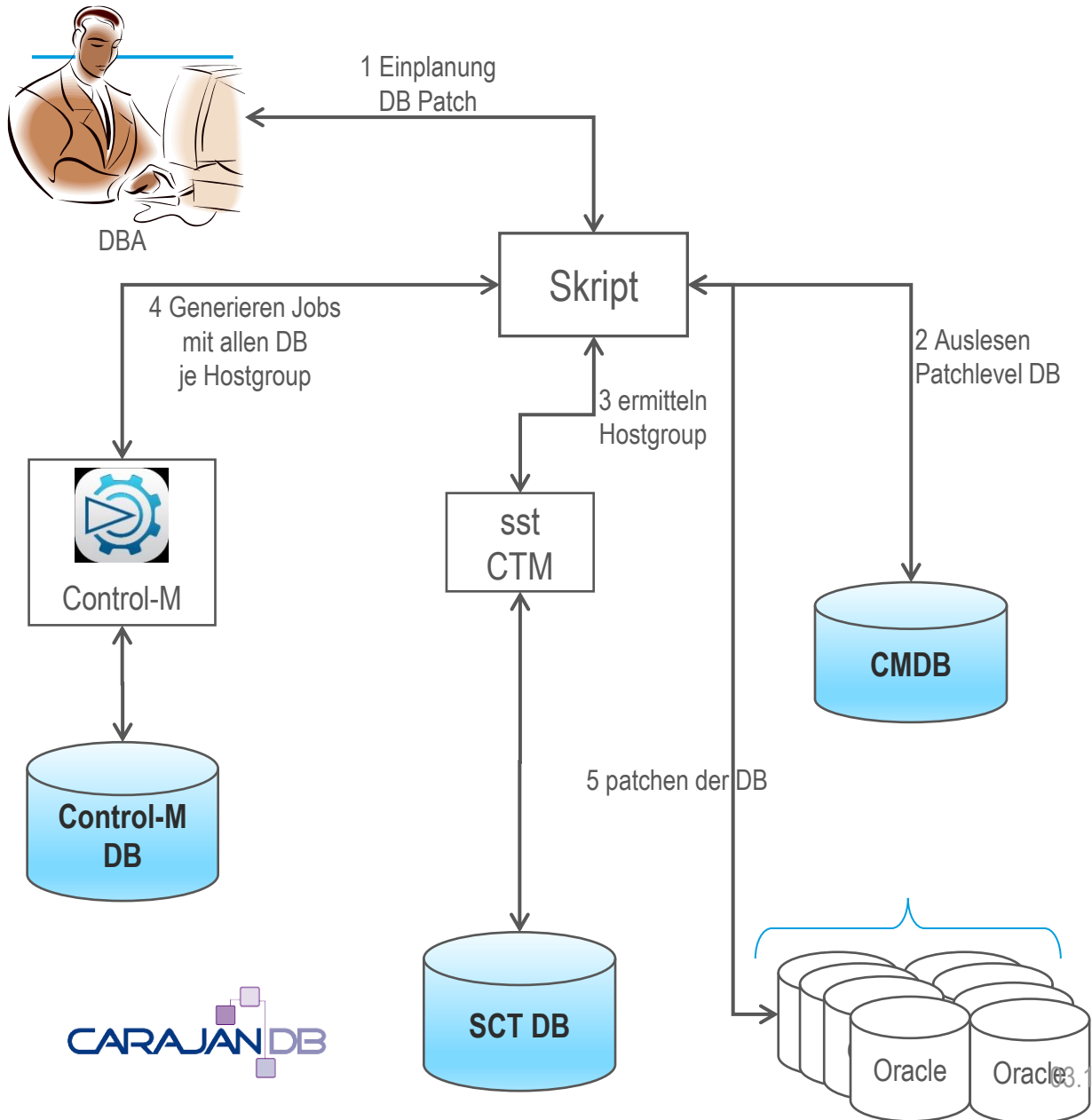
s  
Repository  
Backup der FRA

# Patch: Rollout neues Oracle Home



1. Start Rollout Skript Oracle Home
2. Auslesen aller Oracle Instanzen
3. Ermitteln Patchlevel auf Zielsystem
  - FS anlegen
    - Größe in Konfigurationsdatei vermerkt
  - TAR-Image aus „Repository“ holen und entpacken
  - neues ORACLE\_HOME in Inventory registrieren
4. Ermitteln Hostgroups
5. Generieren Jobs für alle DB je Hostgroup
6. Rollout neues Oracle Home

# Patch: Aktualisieren der Datenbanken



1. Einplanen des DB Patch
2. Auslesen der Patchlevel
3. Ermitteln der Hostgroup
4. Generieren der Jobs mit allen DB je Hostgroup
5. Patchen der DB Instanz
  - evtl. störende alte Patches deinstalliert
  - die Instanz herunterfährt
  - Den Listener herunterfährt
  - die Umgebung auf das neue OH setzt und daraus die Instanz und den Listener startet
    - listener.ora wird angepasst
  - datapatch aufruft
  - Erfolgreichen Patch testen

- ULA mit Oracle seit 2012
- Lizenzumfang
  - DB Enterprise Edition (unlimitiert)
  - Partitioning (unlimitiert)
  - Diagnostic und Tuning Pack (unlimitiert)
  - Advanced Compression (unlimitiert)
  - Advanced Security Option (unlimitiert)
  - In-Memory (unlimitiert)
  - Multitenant (28 – Processor Perputal)
  - Enterprise User Security für Oracle unified Directory Service
- Virtualisierungsvereinbarung seit 2018

# 2

## Multitanent Datenbank



# Ausgangslage aus Sicht der Technik und Lizenzsituation

## Zielarchitektur

- Multitenant mit max. 3 PDBs
- CDB Cloning (Templates)
- PDB kann nicht geklont werden (Details)
- Ansible für Software Rollout
- Shell Skripting für Datenbank Rollout

## Spezielle Herausforderungen

- Keine Multitenant Option lizenziert
- Namenskonventionen
- Einbindung EUS / AD mit Common User
- PDB Template umständlich





---

## 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

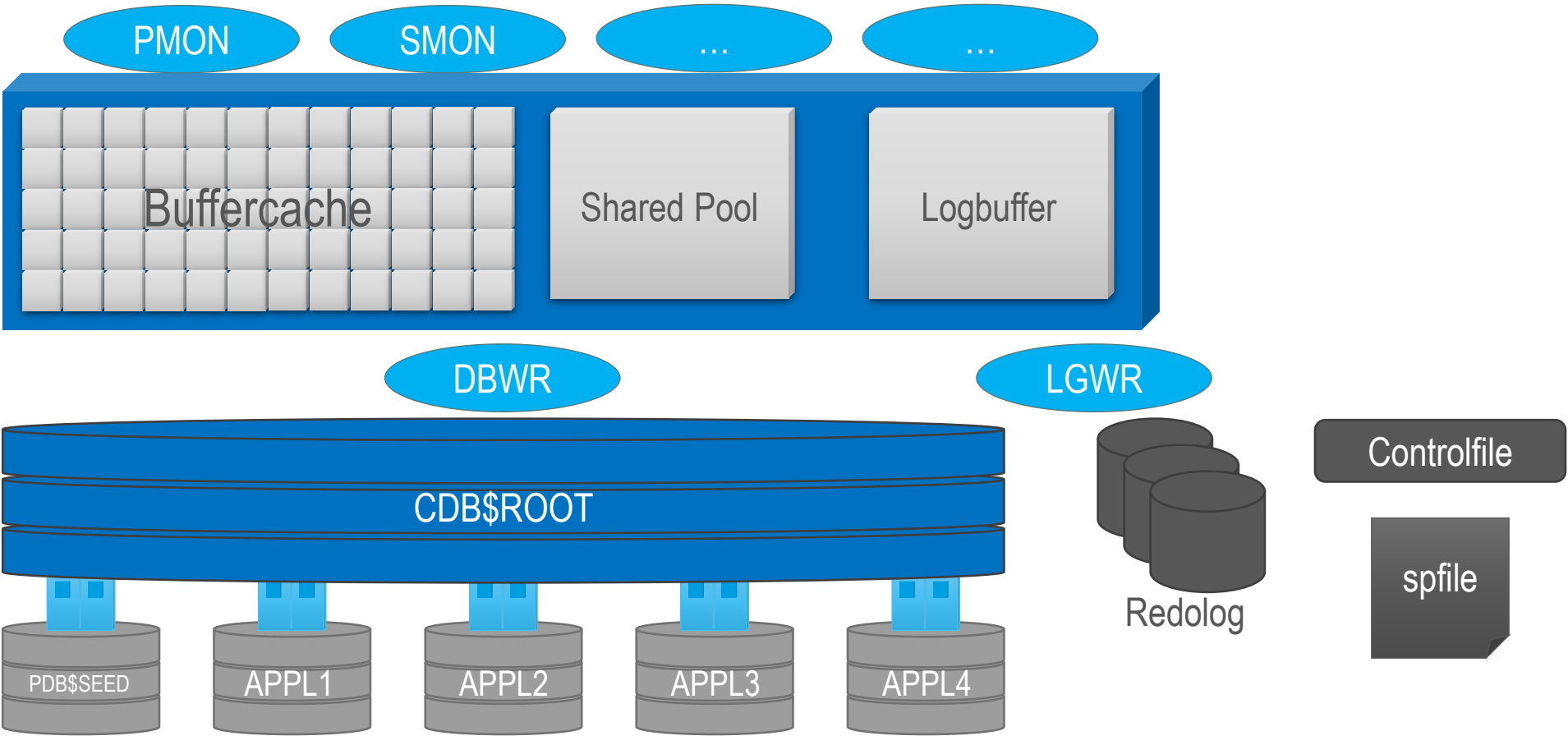
Oracle recommends use of the CDB architecture.

The non-CDB architecture will be desupported from Oracle 20c on  
(OOW 2019 Dominic Gilles)

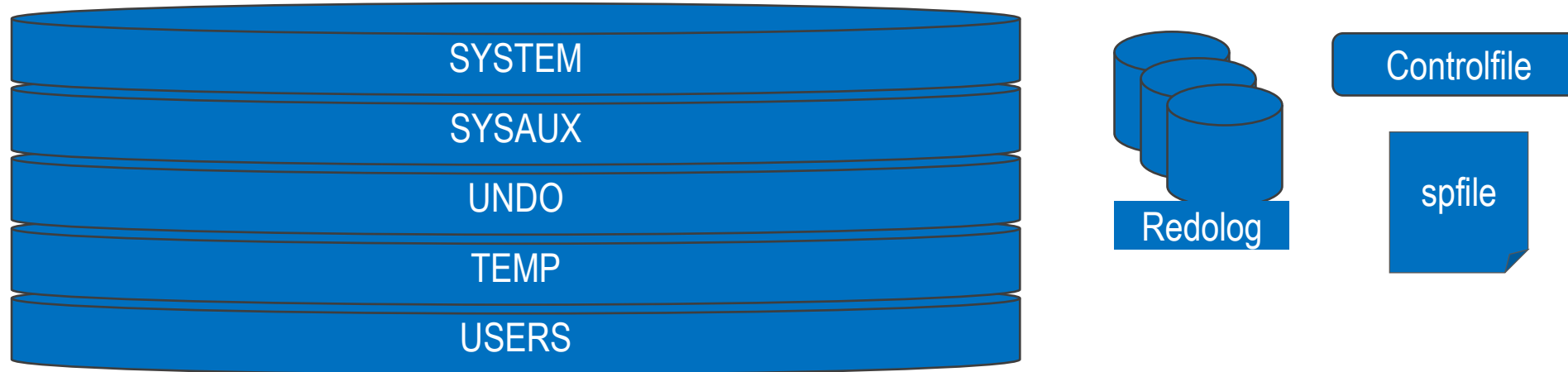
## 3 PDBs in jeder Edition

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	<p>The number value in each column indicates the maximum number of pluggable databases (PDBs) that can be created for each offering.</p> <ul style="list-style-type: none"> <li>• 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.</li> <li>• <b>EE:</b> Extra cost option; if you are licensed for Oracle Multitenant, then you can create up to 252 PDBs.</li> <li>• <b>EE-ES:</b> Extra cost option; if you are licensed for Oracle Multitenant, then you can create up to 4096 PDBs.</li> <li>• <b>DBCS EE-HP, DBCS EE-EP, and ExaCS:</b> Included option; you can create up to 4096 PDBs.</li> </ul>
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	

# Architektur



# Root Container (CDB)



Komponenten für den Aufbau der Instanz

- Controlfiles
- spfile
  - Connect an die Instanz nur über die CDB möglich

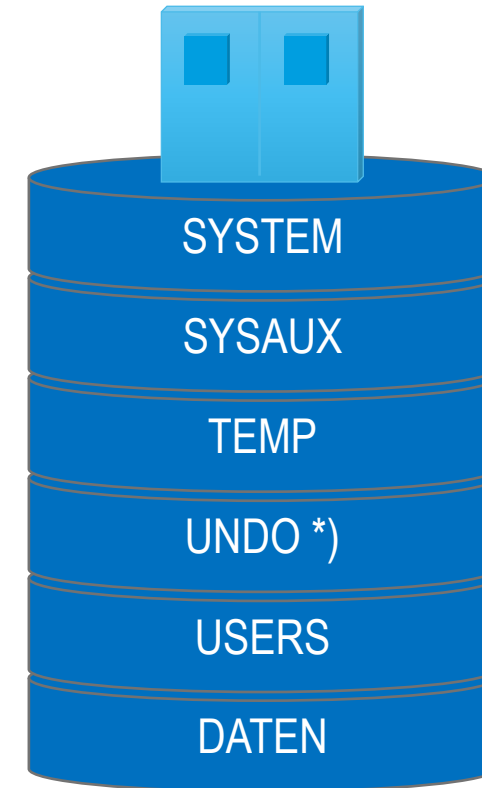
Verwaltung der Redolog-Dateien

- 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

# Common Roles

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##"

Common Privilegien durch Angabe von „CONTAINER=ALL“

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

```
Role created.
```

```
SQL> GRANT CREATE SESSION TO c##dba;
```

```
Grant succeeded.
```

## Voraussetzung wie bei der Role

```
SQL> CREATE USER c##dbauser IDENTIFIED BY "oracle" CONTAINER=ALL;  
User created.
```

```
SQL> GRANT c##dba TO c##dbauser;  
Grant succeeded.
```

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

```
SQL> ALTER SESSION SET CONTAINER=nachtmusik;  
ERROR:  
ORA-01031: insufficient privileges
```

## Warum?

- Weil das Privileg „CREATE SESSION“ nur für die CDB vergeben worden ist!



Definition eines Globalen (nicht Common) Users

- Zuordnung der DBAs, etc. zu den Global Usern über Active Directory

Problem?

- Derzeit ist der Global User ohne Prefix definiert
- Dadurch kann er aber nicht Common User werden
- Änderungen des Global Users in AD nicht möglich, weil noch NON-CDB Datenbank existieren

Lösung

- temporär Parameter `common_user_prefix = ''`
- Restart der Datenbank
- Benutzer anlegen
- Parameter zurücksetzen
- Restart der Datenbank

---

Derzeit:

- Name der Datenbank = Anwendungskürzel (meist dreistellig)

Multitenant:

- ????
- Name der CDB = Name der Anwendung plus „C“
- Name der PDB = Name der Anwendung
- Frage: was passiert, wenn wir mehrere PDBs in einer CDB haben?
- Alternative: CDBs durchnummerieren

## Andere Objekte

---

Es gibt keine „Common“ Packages, Procedures, DB-Links, etc.

- Außer, wenn Sie bei der Erstellung der Datenbank direkt mit installiert wurden

Daher müssen diese Objekte in jeder PDB neu installiert werden

---

Erstellen der CDB aus einem eigenen Template

- Erstellen der Datenbank dauert ca. 10 Minuten
- Alle Common Rollen, Profiles, User enthalten

Erstellen der PDB aus der PDB\$SEED

- PDB\$SEED kann nicht angepasst werden
- Es müssen diverse Skripte nachinstalliert werden

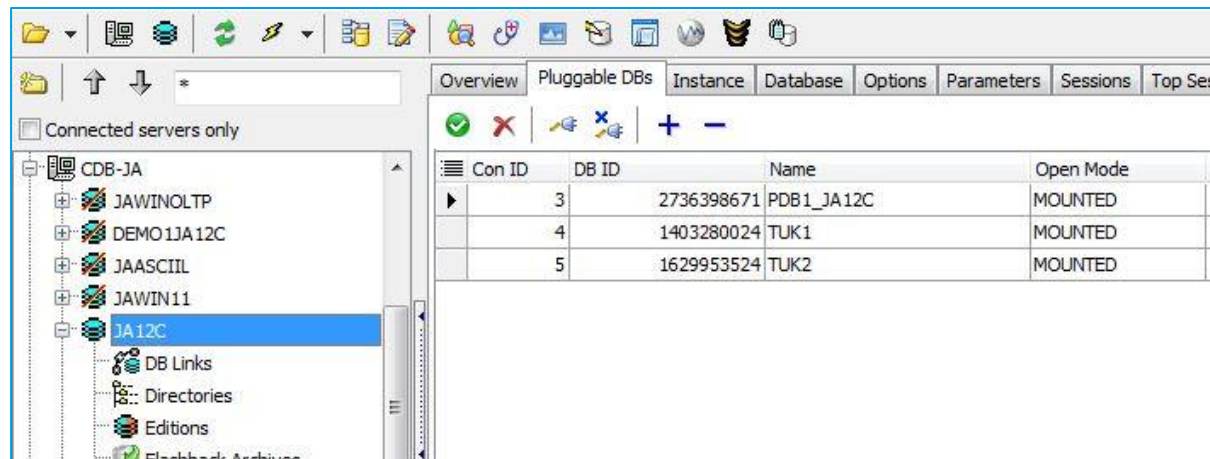
ca. 75 % aller Datenbanken benutzen noch WE8ISO8859P15

Ab 12.2: CDB mit Unicode kann PDB mit WE8 Zeichensatz enthalten (nicht aber umgekehrt!)

- aber: in einer Unicode CDB kann keine WE8 Datenbank erstellt werden
- Temporäre Lösung: CDB und PDB mit gleichem Zeichensatz (unflexibel)

z.B. Toad:

- Der DBA muss sich an jede PDB und jede CDB einzeln anmelden
- Für Anwendungen (z.B. DB-Visualizer) kein Problem, da nur eine Anmeldung an die PDB möglich ist



---

Actifio

Funktioniert problemlos (Version beachten)

Recovery bzw. Cloning nur von der gesamten CDB mit allen PDBs

- Muss noch intensiver getestet werden

# Neue Möglichkeiten

---

Cloning von Datenbank (DevOps)

Snapshot Cloning

Refresh2I (Neuaufbau der Vorproduktion) → Cloning über Datenbank –Link



... und was ist mit Patching?



## New Paradigms for Rapid Patching and Upgrades

The investment of time and effort to patch a multitenant container database results in patching all of its many pluggable databases. To patch a single pluggable database, you simply unplug/plug the 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>

---

Neues ORACLE\_HOME als Klon

CDB und alle PDBs werden gemeinsam gepatcht

# Ressourcenverbrauch

---

Derzeit kein nennenswerter Overhead erkennbar

Benchmark mit Swingbench zeigt keinen Unterschied von 12.1 (NON-CDB) zu 19c Multitenant

# 3

## Aktueller Status Fazit



# Aktueller Status Quo

- 
- Viele Skripte wurden überarbeitet
  - Self Service für die vollautomatisierte Bereitstellung ist noch in Arbeit
  - Erste produktive Instanz wird derzeit ausgerollt um Erfahrungen zu sammeln und etwaige todo's zu ermitteln
  - ....

# Ist 19c als Schritt in Richtung autonomes Database zu werten?

In der Stadt am Rhein wird jetzt die deutschlandweit erste Buslinie mit autonom fahrenden Bussen betrieben. Ganz ohne Mensch kommen die aber doch nicht vom Fleck.

<https://www.sueddeutsche.de/auto/automatisiert-fahrende-elektro-busse-in-monheim-nrw-1.4820833>

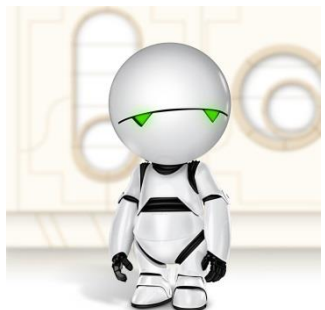


# Die 5 Stufen der Datenbank Automation

## Wie ist 19c hier einzustufen?



**DB Admin**



**Roboter**



- Eine durchgängige Trennung von CDB und PDB zu realisieren.
- Bereitstellung konfigurierbarer Self Services, mit denen die gängigen Funktionen (anlegen, sichern, patchen, löschen) von Datenbank Instanzen einfach bereitgestellt werden können.
- Integration einer Configuration Management Database (CMDB) für weitere Automationsmöglichkeiten.
- ...
- Wenn Multitenant der Standard ist, sollte Multitenant auch in einer Standardlizenz enthalten sein!



✓ **proaktiv**  
☒ **reaktiv**

**INNOVATION IST KEINE  
GARANTIE GEGEN DAS  
SCHEITERN, ABER OHNE  
INNOVATION IST DAS  
SCHEITERN  
GARANTIERT.**

STEFAN R. MUNZ

**PROBLEME KANN MAN  
NIEMALS MIT DERSELBEN  
DENKWEISE LÖSEN, DURCH  
DIE SIE ENTSTANDEN SIND.**

Albert Einstein



"Erfolg ist die Fähigkeit,  
von einer Niederlage  
zur nächsten zu schreiten,  
ohne dabei die Begeisterung  
zu verlieren."

Johannes Ahrends

**Winston Churchill**