

Hochverfügbarkeit für die Oracle Datenbank SE2

Markus Flechtner

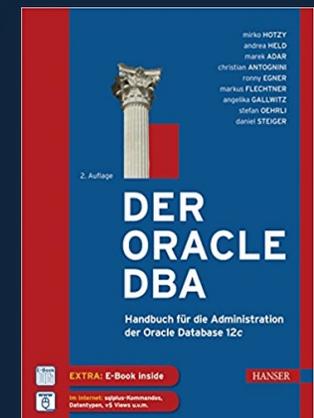
November 2024
DOAG Konferenz + Ausstellung
Nürnberg



Markus Flechtner

- Diplom-Mathematiker
- Oracle seit 1990
- Bisher: Entwickler, Oracle-Field-Support, DBA, Consultant
- Aktuell: Principal Consultant & Abteilungsleiter bei der ORDIX AG
- Schwerpunkte
 - Hochverfügbarkeit
 - Migrationen und Upgrades
 - Multitenant
- DOAG-Themenverantwortlicher "OpenSource-Datenbanken"
-  @markusdba  @markusdba.bsky.social
-  @markusdba@mastodon.social
-  markusdba.de|.net  mfl@ordix.de

ORDIX AG

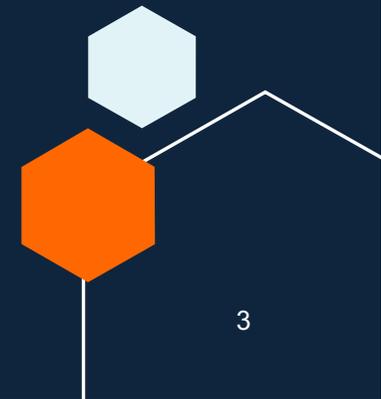


 Oracle ACE Pro

Agenda

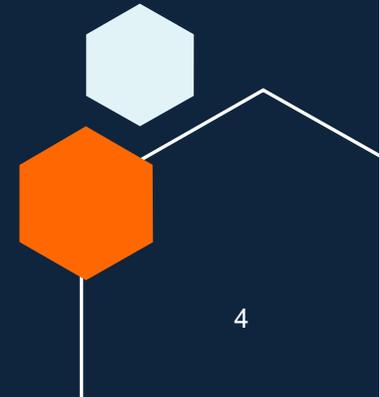
ORDIX AG

- **Einführung**
- Standard Edition High Availability („SEHA“)
- Hochverfügbarkeit mittels „Refreshable PDB“
- „SEMAA“
- Zusammenfassung & weitere Informationen



Woran denkt man, wenn es um Oracle HA geht?

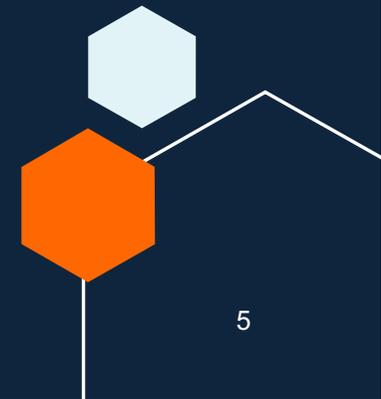
- Enterprise Edition
 - (Active Data Guard
 - Oracle Golden Gate
 - Real Application Cluster
-
- Und natürlich:



Hochverfügbarkeit für die Oracle Datenbank SE2 (bis 18c)

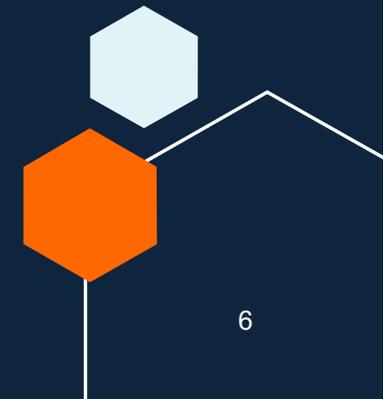
ORDIX AG

- Real Application Cluster
 - Bis Oracle Database 18c
 - 2 Knoten, je 1 Socket
 - Max. 8 Threads pro Instanz
- Failover Cluster
 - Mit eigenen Skripten
- Data Guard-ähnliche Tools von Drittherstellern



Was sind die aktuellen
"out-of-the-box" Hochverfügbarkeitslösungen
für die
Oracle Datenbank Standard Edition 2 - 19c und höher?

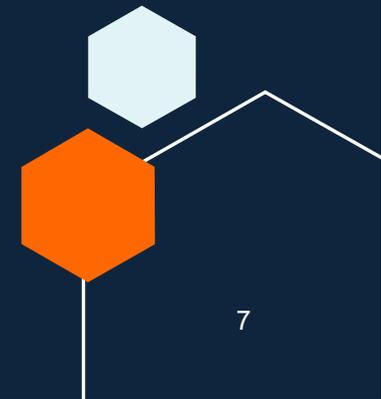
(keine Dritthersteller, keine zusätzlichen Lizenzkosten)



Agenda

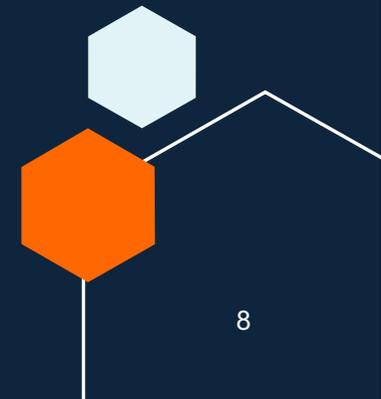
ORDIX AG

- Einführung
- **Standard Edition High Availability („SEHA“)**
- Hochverfügbarkeit mittels „Refreshable PDB“
- „SEMAA“
- Zusammenfassung & weitere Informationen



Standard Edition High Availability (SEHA)

- SEHA ist seit dem Release Update vom April 2020 (19.7) verfügbar
 - OCW muss im GI-Home und im RDBMS-Home die gleiche Version haben
- **SEHA ist eine automatisierte Failover-Datenbank-Lösung für die Oracle Datenbank SE2 basierend auf Oracle Grid Infrastructure**
 - ASM oder ACFS
 - Unterstützt auf Linux x86-64, Solaris SPARC, MS Windows (seit 19.7) sowie IBM AIX und HP-UX Itanium (seit 19.13)
- Unter Windows Ersatz/"Nachfolger" für Failsafe (wenn mit SE2 genutzt)
(Failsafe ist seit Oracle Database 12c desupported)



- 2 Sockets pro Server
- 16 Threads pro Instanz
- "10-days-rule" - 10 x 24 h

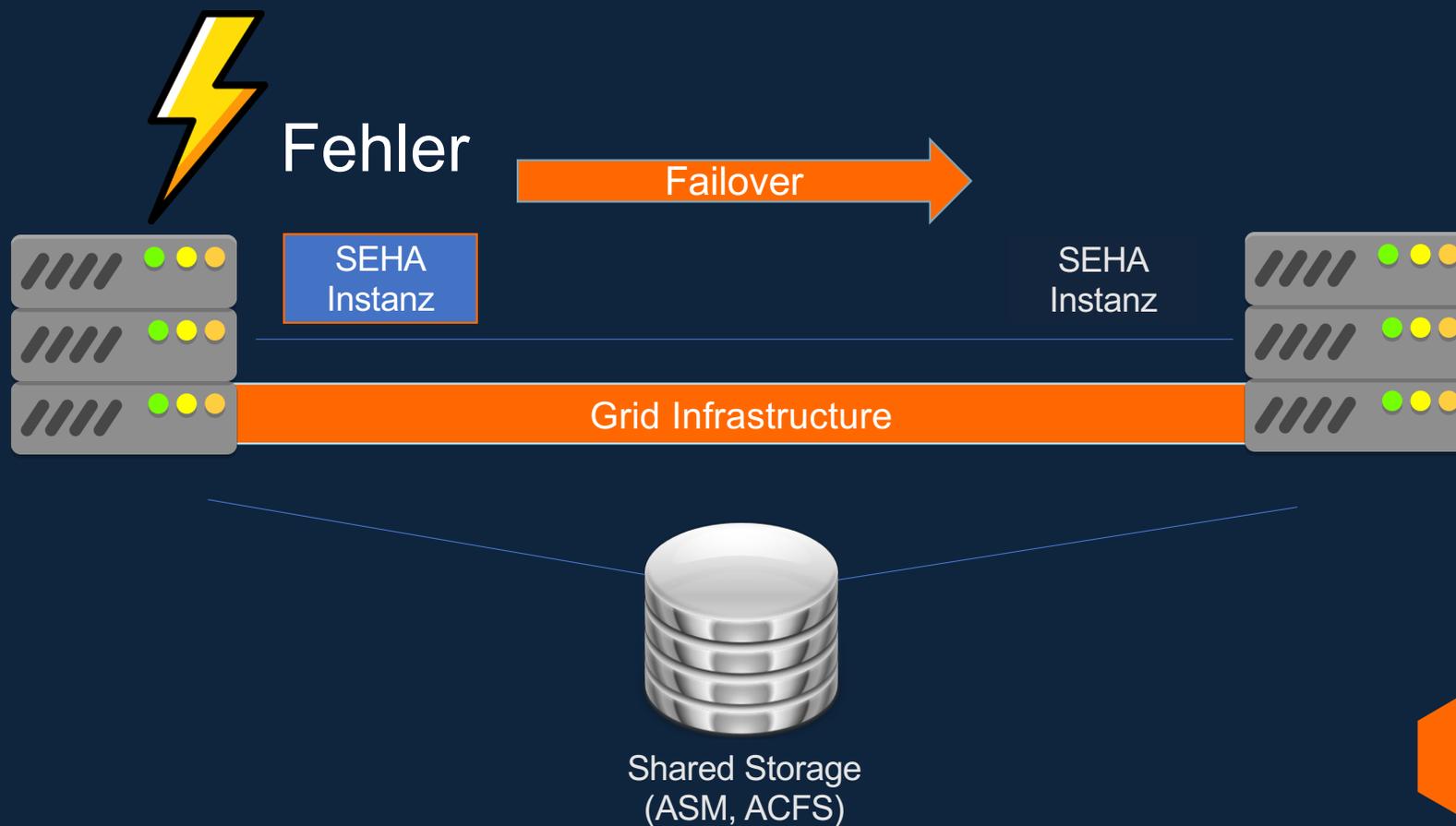
Data Recovery using Clustered Environments (Failover)

The failover data recovery method is an example of a clustered deployment, where multiple nodes/servers have access to one Single Storage/SAN. In such cases your license for the programs listed on the US Oracle Technology Price List (<http://www.oracle.com/corporate/pricing/pricelists.html>) is eligible for the 10-day rule, which includes the right to run the licensed program(s) on an unlicensed spare computer in a failover environment for up to a total of ten separate 24-hour periods in any given calendar year (for example, if a failover node is down for two hours on Tuesday and three hours on Friday, it counts as two 24-hour periods). The above right only applies when a number of physical or logical machines as defined in Oracle's Partitioning Policy (detailed in <https://www.oracle.com/assets/partitioning-070609.pdf>) are arranged in a cluster and share one logical disk array located in a single data center. When the primary node fails, the failover node acts as the primary node. Once the primary node is repaired, you must either switch back or designate that repaired server as the failover node. Once the failover period has exceeded ten 24-hour periods, the failover node must be licensed. In addition, only one failover node per clustered environment is at no charge even if

- Quelle: <https://www.oracle.com/a/ocom/docs/data-recovery-licensing-070587.pdf>

SEHA - Überblick

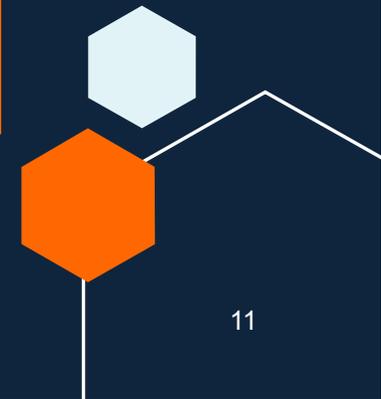
ORDIX AG



SEHA - Setup (1)

1. SE2-Datenbank auf einem Cluster-Knoten anlegen
 - Als "Single-Instance-Datenbank"
 - CDB und Non-CDB möglich
 - DBCA unterstützt den SEHA-Setup (noch) nicht
2. Ggf. Password-Datei und spfile ins ASM kopieren
 - Beispiel (Password-Datei)

```
$ asmcmd pwcopu /u00/app/oracle/dbs/orapwSEHADB +DATA/SEHADB/orapwSEHADB  
$ srvctl modify database -db SEHADB -pwfile +DATA/SEHADB/orapwSEHADB
```

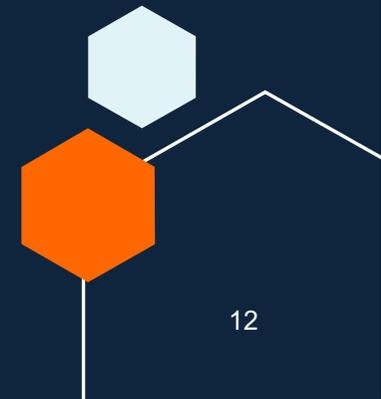


SEHA - Setup (2)

- Liste der Knoten, auf denen die SEHA-Datenbank laufen darf, definieren

```
$ srvctl modify database -db SEHADB -node node1,node2
```

- Die notwendigen Verzeichnisse (ADR etc.) auf dem 2.Knoten werden angelegt, sobald die Instanz das erste Mal auf dem Knoten gestartet wird.



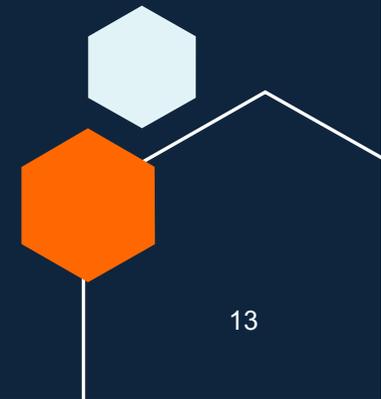
Verschieben einer Datenbank

```
$ srvctl status database -db SEHADB
Instance SEHADB is running on node node1

$ srvctl relocate database -db SEHADB -node node2

$ srvctl status database -db SEHADB
Instance SEHADB is running on node node2
```

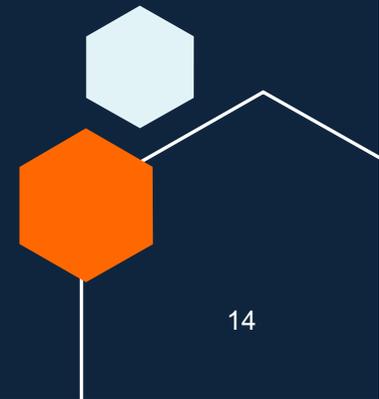
- Die Datenbank wird auf node1 runtergefahren ("SHUTDOWN IMMEDIATE")
 - Dauer: abhängig von den offenen Transaktionen (Rollback beim "SHUTDOWN IMMEDIATE")
- Die Datenbank-Instanz wird auf dem 2.Knoten gestartet
- **Es gibt immer eine Auszeit für die Applikationen.**
 - SEHA ist nicht "RAC One Node"



Datenbank-Failover

ORDIX AG

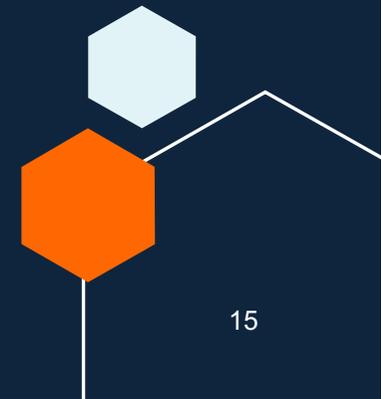
- Alles ganz einfach:
 - Die Oracle Clusterware stellt fest, dass der Knoten ausgefallen ist und fährt die DB-Instanz automatisch auf einem anderen Knoten hoch. 😊



Wie melden sich die Anwendungen an einer SEHA-DB an?

- Über den SCAN-Namen
- Mit einem Datenbank-Service (Applikations-Service)

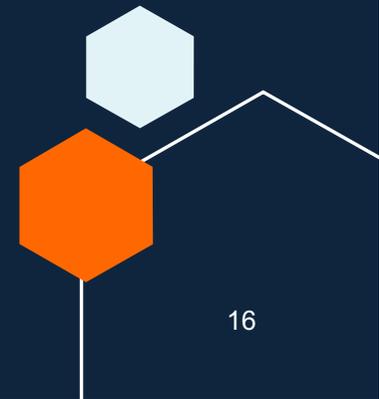
```
SEHA_APP =  
  (DESCRIPTION =  
    (ADDRESS = (PROTOCOL = TCP) (HOST = sehacluster) (PORT=1521))  
    (CONNECT_DATA =  
      (SERVER = DEDICATED)  
      (SERVICE_NAME = SEHA_APP_SVC)    )  
  )
```



Hochverfügbarkeits-Optionen für die Anwendungen

- Transparent Application Failover (TAF)
- Fast Application Notification (FAN)
- Fast Connection Failover (FCF)

- (Transparent) Application Continuity ist nicht verfügbar
 - Lizenzvoraussetzungen:
Enterprise Edition
+ RAC oder RAC One Node oder Active Data Guard

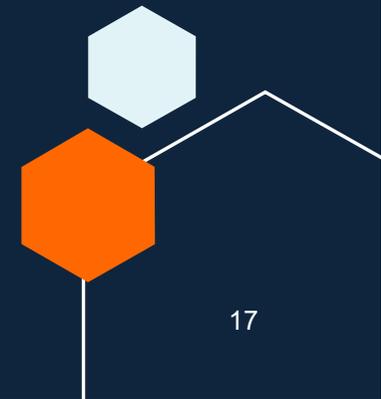


- Wenn man eine SEHA-Datenbank manuell startet, kann man den Server angeben, auf dem sie laufen soll:

```
$ srvctl start database -db SEHADB -node node2
```

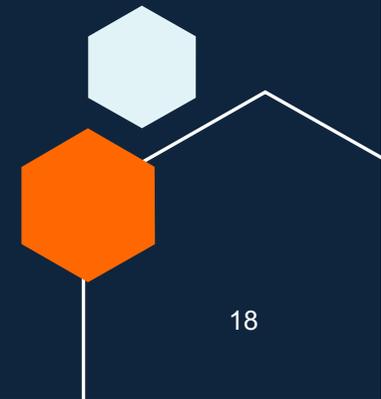
- Wenn man versucht, eine Enterprise-Edition-Datenbank wie eine SEHA-Datenbank zu konfigurieren, kommt eine Fehlermeldung 😞

```
PRCD-2088 : failed to configure the single instance database EEHACDB  
with multiple nodes because it is not a Standard Edition database
```



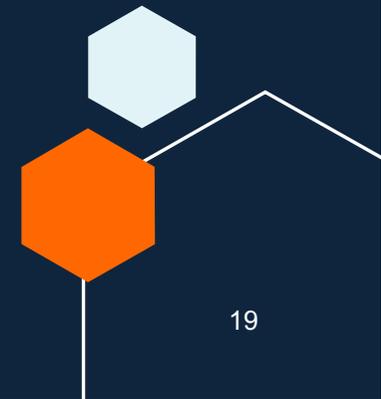
Patchen einer SEHA-Datenbank - grober Ablauf

1. Out-of-place-patching auf dem "passiven Knoten"
2. Oracle-Home auf dem passiven Knoten in der Cluster Registry anpassen
 - "srvctl modify database .. -oraclehome .."
3. Datenbank auf den "passiven Knoten" verschieben
4. Datapatch ausführen
5. Out-of-place-patching auf dem anderen Knoten
6. Oracle-Home auf dem anderen Knoten in der Cluster Registry anpassen
 - "srvctl modify database .. -oraclehome .."



Agenda

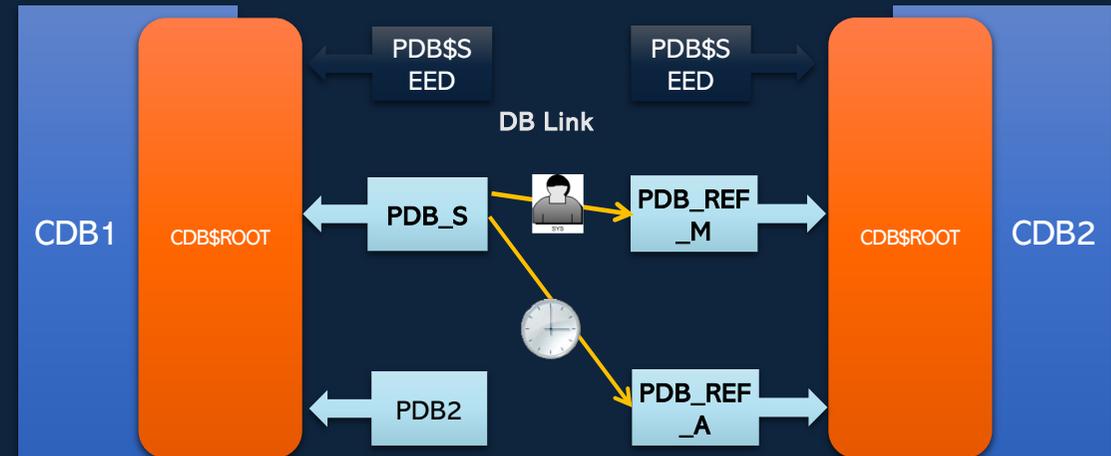
- Einführung
- Standard Edition High Availability („SEHA“)
- **Hochverfügbarkeit mittels „Refreshable PDB“**
- „SEMAA“
- Zusammenfassung & weitere Informationen



Grundlagen (1)

ORDIX AG

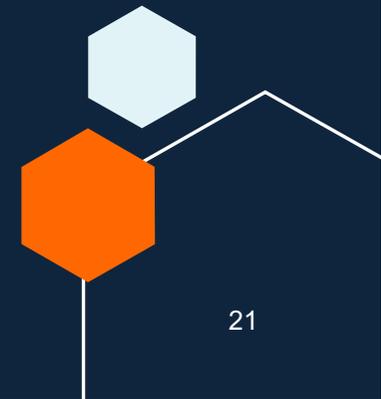
- Idee: eine "remote PDB" wird regelmäßig aktualisiert
 - Initial: Ausgangs PDB kopieren
 - Anschließend: regelmäßig die Änderungen übertragen (Online-Redo-Logs und Archive Logs)



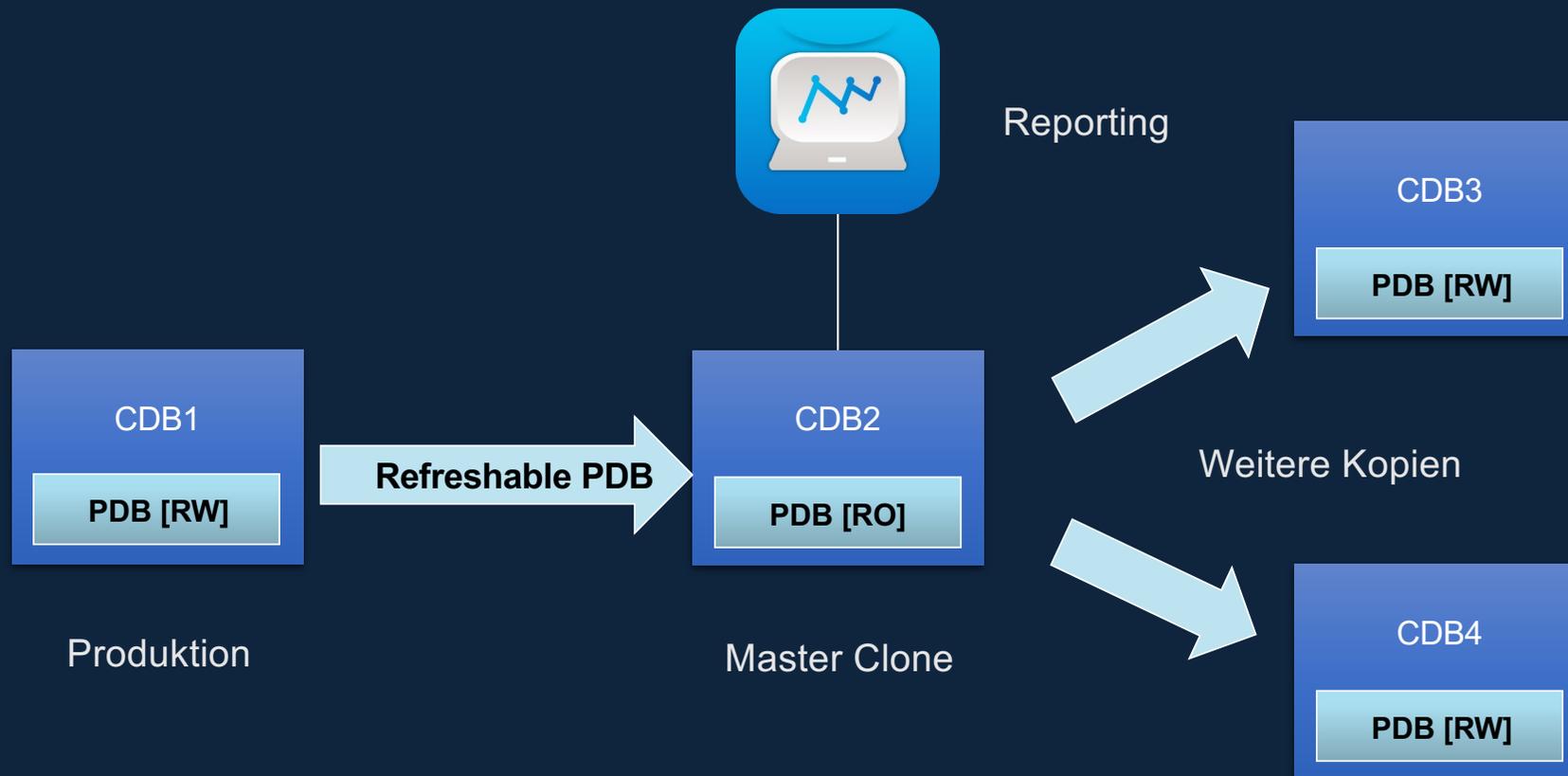
- Anwendungsmöglichkeiten
 - Reporting (Refreshable PDB "read-only" öffnen)
 - Refreshable PDB als Quelle für weitere Kopien
 - Hochverfügbarkeit für SE2 ("Data Guard Lite")

Grundlagen (2)

- Eine Refreshable PDB kann nur "read-only" geöffnet werden
 - Eine Aktualisierung (Refresh) kann nur erfolgen, wenn die Refreshable PDB geschlossen ist
- Die Aktualisierung kann automatisch (Scheduler-Job) oder manuell erfolgen
- Quelle und Ziel müssen in verschiedenen CDBs sein
- Die Quelle kann auch eine Non-CDB sein
 - Hilfreich für Migrationen in die Multitenant Architektur



Anwendungsfälle



Setup (1)

- Anforderungen an die Quell-CDB
 - Local Undo
 - Archivelog Modus

1. "Common User" in der Quell-CDB anlegen

```
SQL> CREATE USER C##REFRESH identified by <PW>;  
SQL> GRANT CREATE SESSION,RESTRICTED SESSION,SYSOPER to C##REFRESH  
2 CONTAINER=ALL;  
SQL> GRANT CREATE PLUGGABLE DATABASE TO C##REFRESH;
```

2. "Public Datenbank-Link vom Ziel zur Quelle (CDB\$ROOT oder Quell-PDB) anlegen

```
SQL> CREATE PUBLIC DATABASE LINK <SOURCE_CDB>  
2 CONNECT TO C##REFRESH IDENTIFIED BY '<PW>' USING '<TNS>';
```

Setup (2)

3. Refreshable PDB anlegen

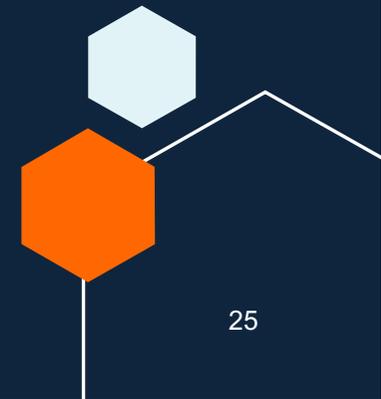
```
SQL> CREATE PLUGGABLE DATABASE <PDB> FROM <PDB>@<SOURCE_CDB>  
2 file_name_convert= ... REFRESH MODE EVERY 240 MINUTES;
```

- Mögliche Refresh-Modi:
 - REFRESH MODE EVERY <n> MINUTES
 - REFRESH MODE MANUAL
 - REFRESH NONE
- Bei automatischen Refresh wird in der Ziel-CDB ein Datenbank-Job (Scheduler) angelegt

Kürzestes Intervall:
1 Minute

- Rman sichert eine "Refreshable PDB" **NICHT**
- Eine "Refreshable PDB" kann nicht im read-write-Modus geöffnet werden
 - Schutz vor versehentlichem Öffnen

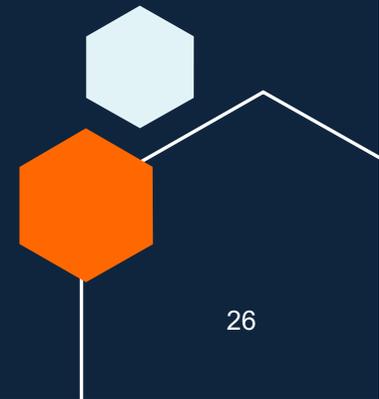
```
ORA-65341: cannot open pluggable database in read/write mode
```



Aktualisierung einer "Refreshable PDB"

- Die PDB muss geschlossen sein
 - Ansonsten wird die Aktualisierung beim automatischen Refresh übersprungen
- Manueller Refresh

```
SQL> ALTER PLUGGABLE DATABASE <PDB> CLOSE IMMEDIATE;  
SQL> ALTER PLUGGABLE DATABASE <PDB> REFRESH;
```

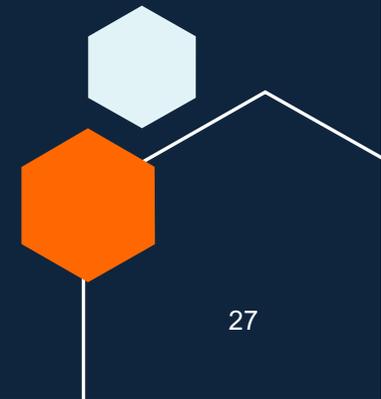


Wie macht man aus einer "Refreshable PDB" eine "normale PDB"?

- **Wichtig:** die Quell-Datenbank muss erreichbar sein
- Auf der Ziel CDB:
 - Refresh deaktivieren
 - PDB öffnen

```
SQL> ALTER PLUGGABLE DATABASE <PDB> REFRESH MODE NONE ;  
SQL> ALTER PLUGGABLE DATABASE <PDB> OPEN ;
```

- Eine Rückkehr zur "Refreshable PDB" ist danach nicht mehr möglich



```
SQL> select PDB_NAME, REFRESH_MODE, REFRESH_INTERVAL, LAST_REFRESH_SCN  
2      from cdb_pdbs;
```

PDB_NAME	REFRES	REFRESH_INTERVAL	LAST_REFRESH_SCN
PDB\$SEED	NONE		
SE2PDB	AUTO	1	1672347

Von der Quelle

```
SQL> select scn_to_timestamp(1672347) from dual@<source_db>;
```

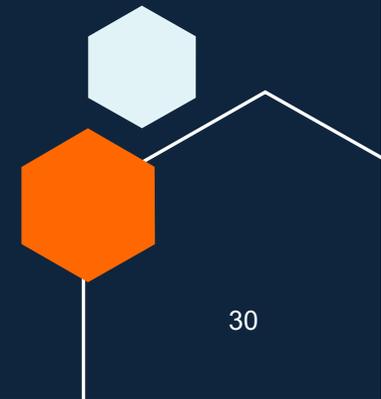
```
SCN_TO_TIMESTAMP(1672347)
```

```
-----  
16-AUG-20 01.07.16.000000000 PM
```

```
SE2PDB(4):alter pluggable database refresh
2020-08-16T11:58:12.076936+02:00
Applying media recovery for pdb-4099 from SCN 1672311 to SCN 1672347
Remote log information: count-1
thr-1,seq-20,logfile-
/u02/fast_recovery_area/SE2CDBA/foreign_archive/SE2PDB/2020_08_16/o1_mf_1_20_hml0s
mw3_.arc,los-1592646,nxs-18446744073709551615,maxblks-366859
SE2PDB(4):Media Recovery Start
2020-08-16T11:58:12.077338+02:00
SE2PDB(4):Serial Media Recovery started
SE2PDB(4):max_pdb is 5
2020-08-16T11:58:12.154566+02:00
SE2PDB(4):Media Recovery Log
/u02/fast_recovery_area/SE2CDBA/foreign_archive/SE2PDB/2020_08_16/o1_mf_1_20_hml0s
mw3_.arc
2020-08-16T11:58:12.665203+02:00
SE2PDB(4):Incomplete Recovery applied until change 1672347 time 08/16/2020 11:58:10
2020-08-16T11:58:12.668710+02:00
SE2PDB(4):Media Recovery Complete (SE2CDBB)
SE2PDB(4):Completed: alter pluggable database refresh
```

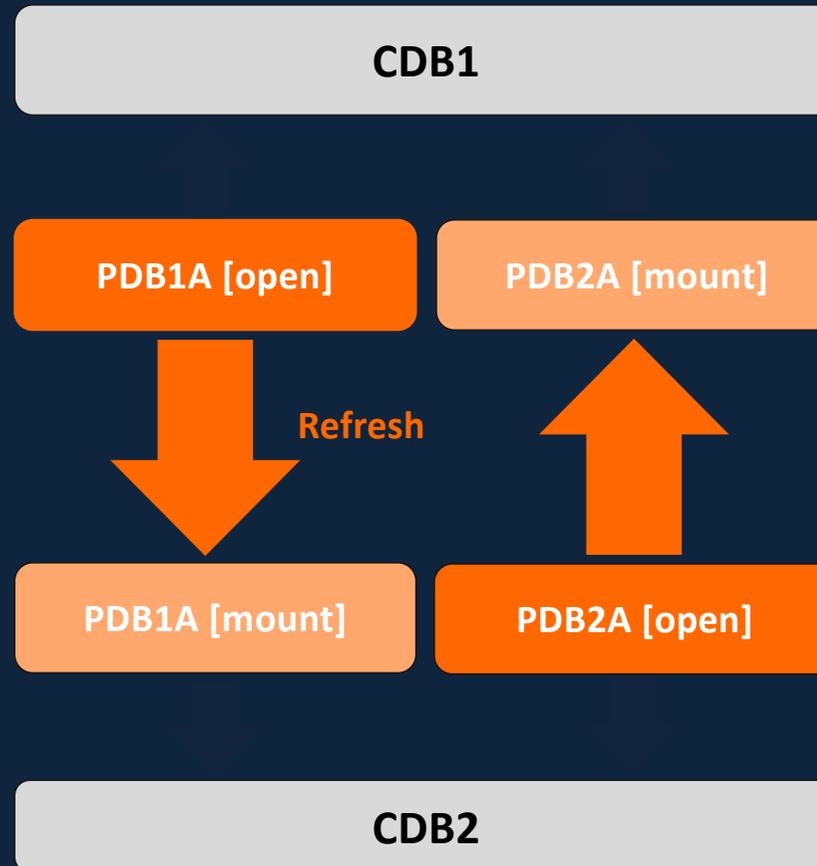
Hinweise zum "Refresh"

- Beim Refresh greift die Refreshable PDB über den Datenbank-Link auf Redo-Informationen der Quelle zu
 - Quell-Datenbank muss erreichbar sein
- Bei längeren Refresh-Intervallen müssen die Archivelogs der Quelle verfügbar sein
 - → ggf. Archive-Log-Deletion-Policy im RMAN ändern
 - Ggf. Über `REMOTE_RECOVERY_FILE_DEST` die Archivelogs für die Ziel-Datenbank verfügbar machen
 - Wenn die Archivelogs nicht verfügbar sind, schlägt der Refresh fehl'
 - Refreshable PDB neu aufbauen oder Archivelogs aus dem Backup holen



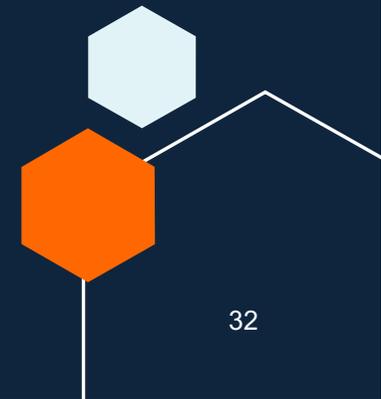
Refreshable PDB für Hochverfügbarkeit

- "Refreshable PDB" als "Standby-PDB"
- Refresh-Intervall: 1 Minute
- Idee:
 - 2 CDBs sichern sich gegenseitig ab



Refreshable PDB für Hochverfügbarkeit - Einschränkungen

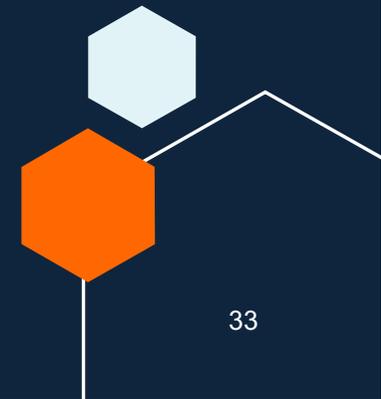
- Ein "Switchover" ist nicht möglich
 - Ausnahme "Refreshable-PDB Switchover bei der Exadata oder in der OCI"
- Es gibt keinen Observer
 - Kein automatischer Failover
 - Aber per Skript möglich
- Nach einem Failover muss die Standby-PDB neu aufgesetzt werden ("kein REINSTATE")
- Wenn die ehemalige "Primary CDB" wieder verfügbar ist, ist eine Split-Brain-Situation möglich
- Der Wechsel der "Refreshable-PDB" in den Read-Write-Modus ist schwierig, wenn die Quell-CDB nicht verfügbar ist



Refreshable PDB für HA - Setup

- "Standby-CDB" anlegen
- Common User für den Refresh in beiden Datenbanken anlegen
- Datenbank-Link von der Standby-DB zur Primary-CDB anlegen
- Refreshable PDB anlegen (Refresh-Interval: 1 Minute)
- Die Refreshable PDB muss im Mount-Status bleiben

```
SQL> CREATE PLUGGABLE DATABASE PDB04
  2 FROM PDB04@CDB1.markusdba.local
  3 file_name_convert=
  4 ('/u01/oradata/CDB1/PDB04', '/u01/oradata/CDB2/PDB04')
  5 REFRESH MODE EVERY 1 MINUTES;
```

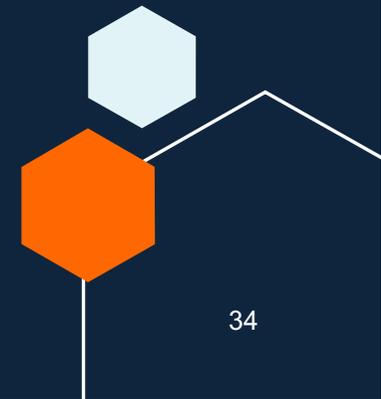


Refreshable PDB für HA - Failover

- Der folgende Befehl funktioniert NICHT, wenn die Quelle nicht mehr erreichbar ist:

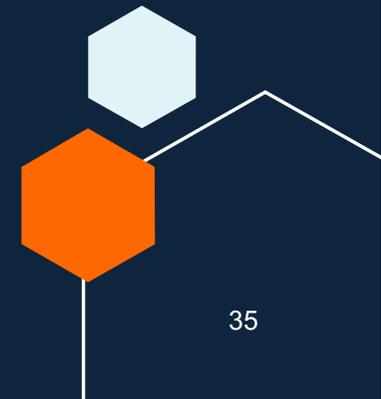
```
SQL> ALTER PLUGGABLE DATABASE <PDB> REFRESH MODE NONE;
```

- Was tun?



Refreshable PDB für HA - Failover - Workarounds

- **Variante 1: "Refreshable PDB" read-only öffnen und nochmal kopieren**
 - Option 1: "Full Clone"
 - Option 2: "Snapshot Clone" und später eine "normale PDB" daraus machen ("ALTER PLUGGABLE DATABASE .. MATERIALIZE;")
 - **Probleme:** Plattenplatz, Dauer, Lizenz (max. 3 PDBs)
- **Variante 2: den Datenbank-Link ändern**
 - Quelle: <https://frankgerasch.de/2022/07/oracle-refreshable-pdb-clone-als-standby-db/>



Failover - "Refreshable PDB" nochmal kopieren

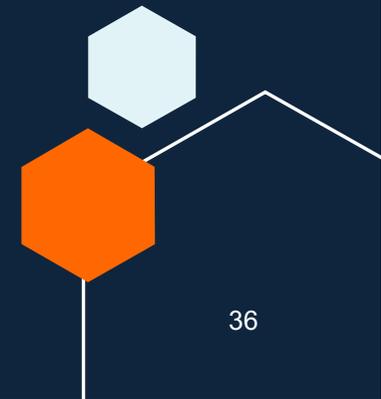
- Normale Kopie ("Full clone")

```
SQL> ALTER PLUGGABLE DATABASE <pdb_name> OPEN READ ONLY;  
SQL> CREATE PLUGGABLE DATABASE <work_pdb> from <pdb_name>  
2 FILE_NAME_CONVERT=(...);
```

- "Snapshot Clone,,

```
SQL> ALTER PLUGGABLE DATABASE <pdb_name> OPEN READ ONLY;  
SQL> CREATE PLUGGABLE DATABASE <work_pdb> from <pdb_name>  
2 FILE_NAME_CONVERT=(...) SNAPSHOT COPY;
```

- Bei ASM nicht möglich
- Ggf. muss der Parameter "CLONEDB=TRUE" gesetzt werden



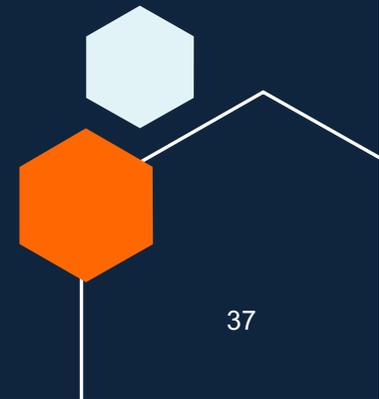
Failover - Workaround: Datenbank-Link ändern (1)

- Dummy-PDB in der Standby-CDB anlegen

```
SQL> create pluggable database dummysdb admin user pdbadmin  
2 identified by manager roles=(DBA)  
3 file_name_convert=('pdbseed', 'DUMMYPDB');  
Pluggable database created.  
SQL> alter pluggable database dummysdb open;  
Pluggable database altered.
```

- "Refresh-Datenbank-Link" zur Dummy-PDB umbiegen

```
SQL> DROP PUBLIC DATABASE LINK SE2CDB_REFRESH;  
Database link dropped.  
  
SQL> CREATE PUBLIC DATABASE LINK SE2CDB_REFRESH  
2 CONNECT TO C##REFRESH IDENTIFIED BY manager USING  
3 'galois.markusdba.internal:1521/dummysdb';  
Database link created.
```



Failover - Workaround: Datenbank-Link ändern (2)

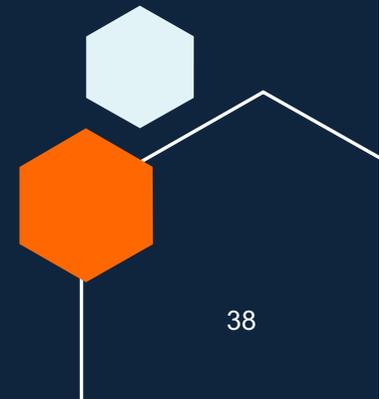
- Refreshmode der Standby-PDB ändern

```
SQL> alter pluggable database SE2PDB refresh mode none;  
Pluggable database altered.
```

- Standby-PDB öffnen

```
SQL> alter pluggable database SE2PDB open;  
Pluggable database altered.
```

Cleverer Idee von Frank Gerasch!

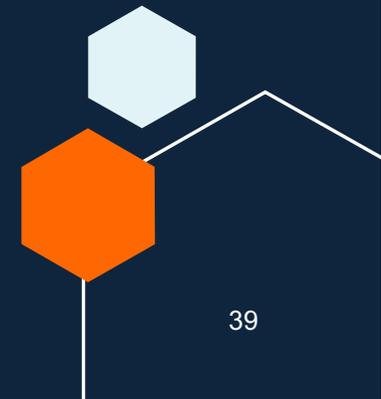


Refreshable PDB für HA - Monitoring

- Zeitliche Differenz zwischen Quelle und Ziel prüfen (auf der Standby-CDB)

```
SQL> select systimestamp - scn_to_timestamp(a.last_refresh_scn) DELTA
2   from dual@SE2CDB,
3   (select LAST_REFRESH_SCN from cdb_pdbs where pdb_name='SE2PDB') a;

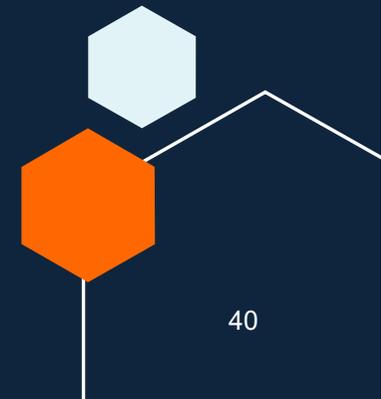
DELTA
-----
+000000000 00:00:30.585794000
```



Client-Connect

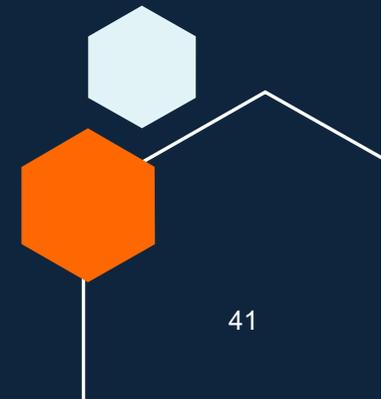
- Application Service verwenden (nicht den Default-Service der PDB)
- Beide Hosts in der tnsnames.ora eintragen

```
PDB_APP.MARKUSDBA =
  (DESCRIPTION =
    (RETRY_COUNT=20) (RETRY_DELAY=3)
    (ADDRESS_LIST=
      (LOAD_BALANCE = OFF )
      (ADDRESS = (PROTOCOL = TCP) (HOST = node1) (PORT=1521))
      (ADDRESS = (PROTOCOL = TCP) (HOST = node2) (PORT=1521))
    )
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = PDB_APP_SVC)
    )
  )
```



Agenda

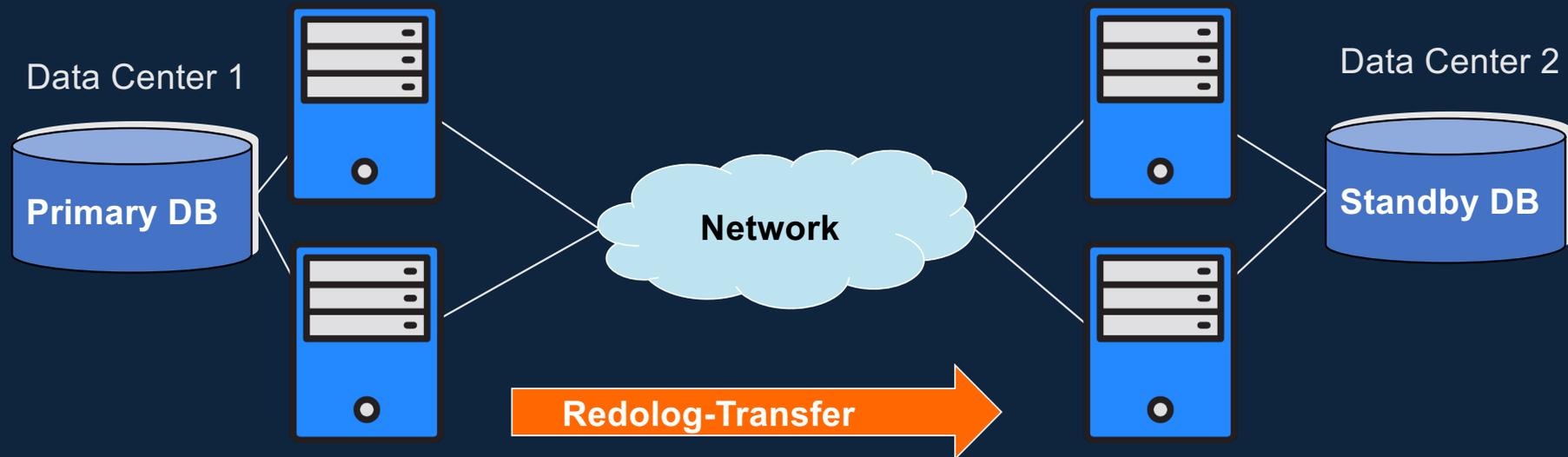
- Einführung
- Standard Edition High Availability („SEHA“)
- Hochverfügbarkeit mittels „Refreshable PDB“
- „SEMAA“
- Zusammenfassung & weitere Informationen



Enterprise Edition: MAA Gold

ORDIX AG

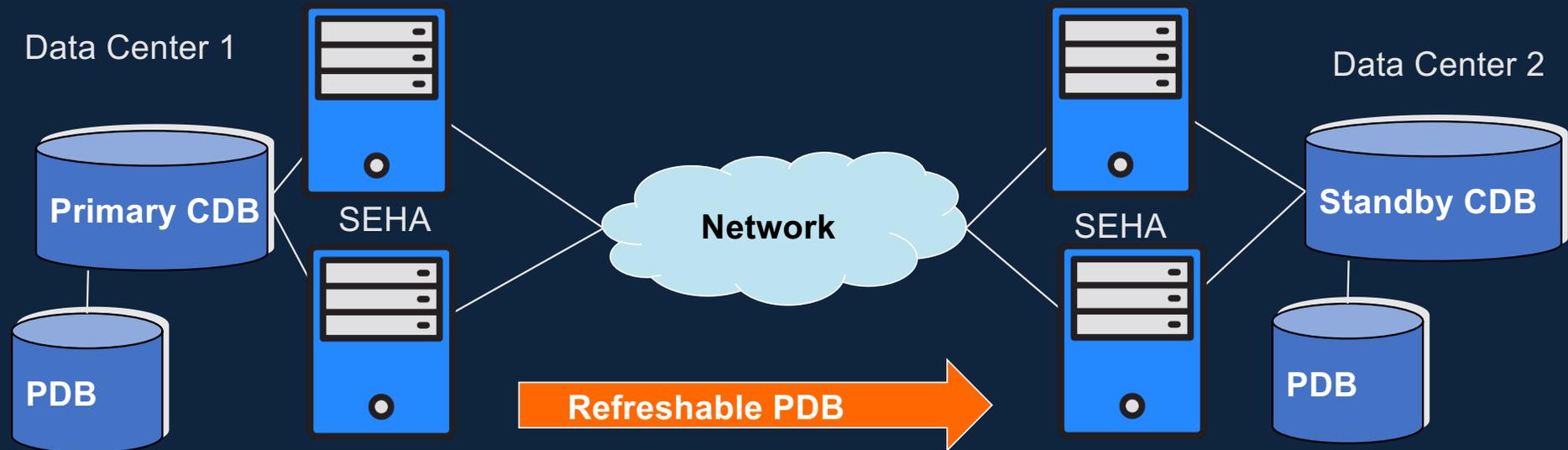
- RAC + DataGuard



SE2 Maximum Availability Architecture („SEMAA“)

ORDIX AG

- SEHA + Refreshable PDB



„SEMAA“ - Setup

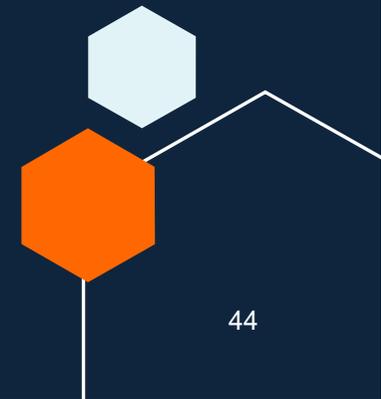
- Es gelten die gleichen Regeln und Einschränkungen wie bei SEHA und "Refreshable PDB"

1. Ablauf

1. "Primär-CDB" anlegen
2. "Primär-CDB " zur "SEHA-CDB" machen
3. "Standby-CDB" anlegen
4. "Standby-CDB " zur "SEHA-CDB" machen
5. "Primär-PDB" anlegen
6. "Standby-PDB" als "Refreshable PDB" anlegen

2. Tnsnames.ora

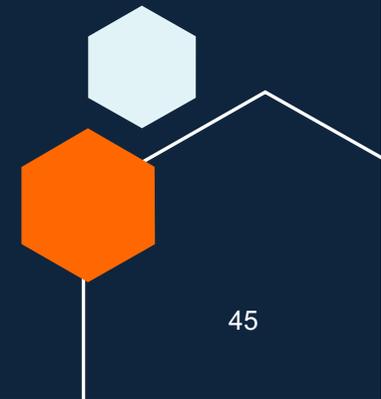
1. Application Service verwenden
2. Beide SCAN-Namen angeben



Agenda

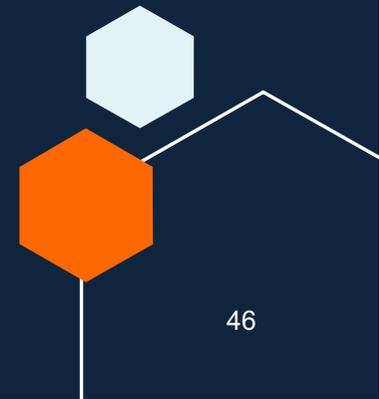
ORDIX AG

- Einführung
- Standard Edition High Availability („SEHA“)
- Hochverfügbarkeit mittels „Refreshable PDB“
- „SEMAA“
- **Zusammenfassung & weitere Informationen**



Zusammenfassung

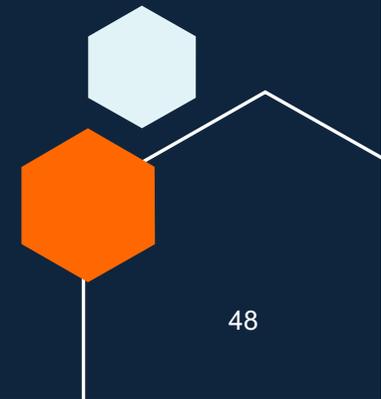
- "SEHA" und "Refreshable PDBs" ermöglichen hochverfügbare Datenbanken für die SE2
- **SEHA**
 - Failover Cluster "out of the box"
 - Einfach aufzusetzen
- **"Refreshable PDBs"**
 - (theoretisch) maximal eine Minute Datenverlust
 - Kein Switchover möglich
 - Failover nicht einfach "out of the box"



**Es ist keine "Enterprise Edition - Hochverfügbarkeit",
aber es kann für viele Kunden ausreichen.**

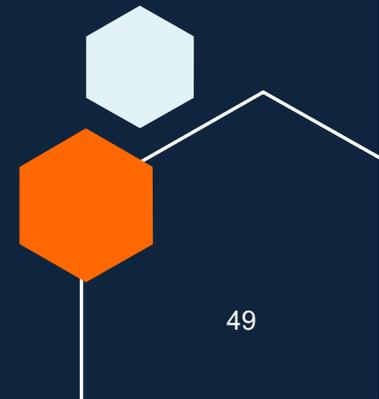
Lizenzierung

- Standard Edition High Availability (SEHA) benötigt keine zusätzliche Lizenz
 - Beide Server können mit 2 Sockets laufen
 - 10-days-Rule beachten
- Grid Infrastructure benötigt keine Lizenz
- Refreshable PDB benötigt keine separate Lizenz
 - Beide Server müssen lizenziert sein
 - Maximal 3 PDBs pro CDB



Lizenzierung - Wie immer gilt:

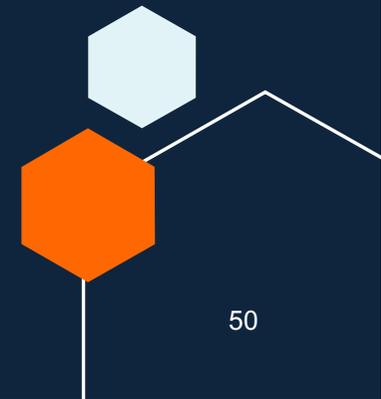
- Prüfen Sie die aktuelle Dokumentation und die aktuellen Lizenzbedingungen
- Schauen Sie in Ihre Verträge mit Oracle
- **Bitten Sie den Oracle-Vertrieb um eine offizielle Aussage**



Weitere Informationen (1)

▪ Oracle Dokumentation

- Oracle Database 19c - Database Installation Guide - Chapter 10 "Installing Standard Edition High Availability"
- Oracle Database 19c "SEHA – Known issues"
<https://docs.oracle.com/en/database/oracle/oracle-database/19/rnrdbm/linux-platform-issues.html#GUID-E462F73F-E37B-4543-B55F-F5CD9543D417>
- Oracle Database 19c – Multitenant Administrators Guide – Section 7.5 "About Refreshable Clone PDBs"



Weitere Informationen (2)

▪ MOS-Notes

- Some Data File Missing From DBA_DATA_FILES or CDB_DATA_FILES in Refreshable PDB (Doc ID 2610743.1)
- How to create refreshable PDB (Doc ID 2441949.1)
- RMAN Skipping Backup of Refreshable PDB (Doc ID 2567138.1)
- Standard Edition High Availability(SEHA) FAQ (Doc ID 2895233.1)

▪ Weitere Informationen von Oracle

- Standard Edition 2 – We Heard You! Announcing: Standard Edition High Availability - <https://blogs.oracle.com/maa/standard-edition-2-announcing-standard-edition-high-availability>
- Licensing Data Recovery Environments: <http://www.oracle.com/us/corporate/pricing/data-recovery-licensing-070587.pdf>

Weitere Informationen (3)

▪ Blog Posts (Englisch)

- <https://www.markusdba.net/2020/05/11/standard-edition-high-availability-first-impressions/>
- <https://gotodba.com/2020/09/02/seha-standard-edition-high-availability-in-oracle-db/>

▪ Blog Posts (Deutsch)

- <https://www.carajandb.com/blog/2020/standard-edition-high-availability/>
- <https://blog.ordix.de/neue-hochverfuegbarkeitsloesung-standard-edition-high-availability-seha-fuer-die-standard-edition-2-se2>
- <https://www.markusdba.de/2022/02/13/von-18c-se2-rac-nach-19c-seha/>
- <https://frankgerasch.de/2022/07/oracle-refreshable-pdb-clone-als-standby-db>

ORDIX AG

Aktiengesellschaft für
Softwareentwicklung, Schulung,
Beratung und Systemintegration

Zentrale Paderborn
Karl-Schurz-Straße 19a
33100 Paderborn
Tel.: 05251 1063-0
Fax: 0180 1 67349 0

Seminarzentrum Wiesbaden
Kreuzberger Ring 13
65205 Wiesbaden
Tel.: 0611 77840-00

info@ordix.de
<https://www.ordix.de/>



**Vielen Dank für
Ihre Aufmerksamkeit**

Markus Flechtner
mfl@ordix.de

Folgen Sie uns für News aus der IT-Welt

ORDIX AG



Folgen Sie dem Unternehmen auf LinkedIn –
um keine Events oder Blogbeiträge zu
verpassen

