

**DOAG 2010**

---

**ORACLE PLATTFORM MIGRATION**

**“CROSS PLATFORM TRANSPORTABLE  
TABLESPACES (XTTS)“**

**METHODE UND ERFAHRUNGSBERICHT**

**JOSEF LIPPERT  
FREIBERUFLICHER IT CONSULTANT  
MÜNCHEN**

---

# Wer bin ich ...

- Freiberuflicher IT Consultant
  - seit mehr als 10 Jahren
- Studium Mathematik
- IT Projekte im Bereich:
  - UNIX/Linux
  - Oracle-DB (Beratung/Administration/Performance)
  - RZ Infrastruktur Management
  - RZ Automation (z. B. Orsyp \$Universe)
  - Monitoring (z. B. BMC, HP, Microsoft)
- Email: [Josef.Lippert@jal-bs.de](mailto:Josef.Lippert@jal-bs.de)

# Agenda

---

- Migrationsmethode
- Limitierungen
- Migrations-Arbeitsschritte
- Methodischer Ablaufplan
- Erfahrungsbericht
- Fallback Szenarien
- Fragen und Antworten

# Oracle Migration

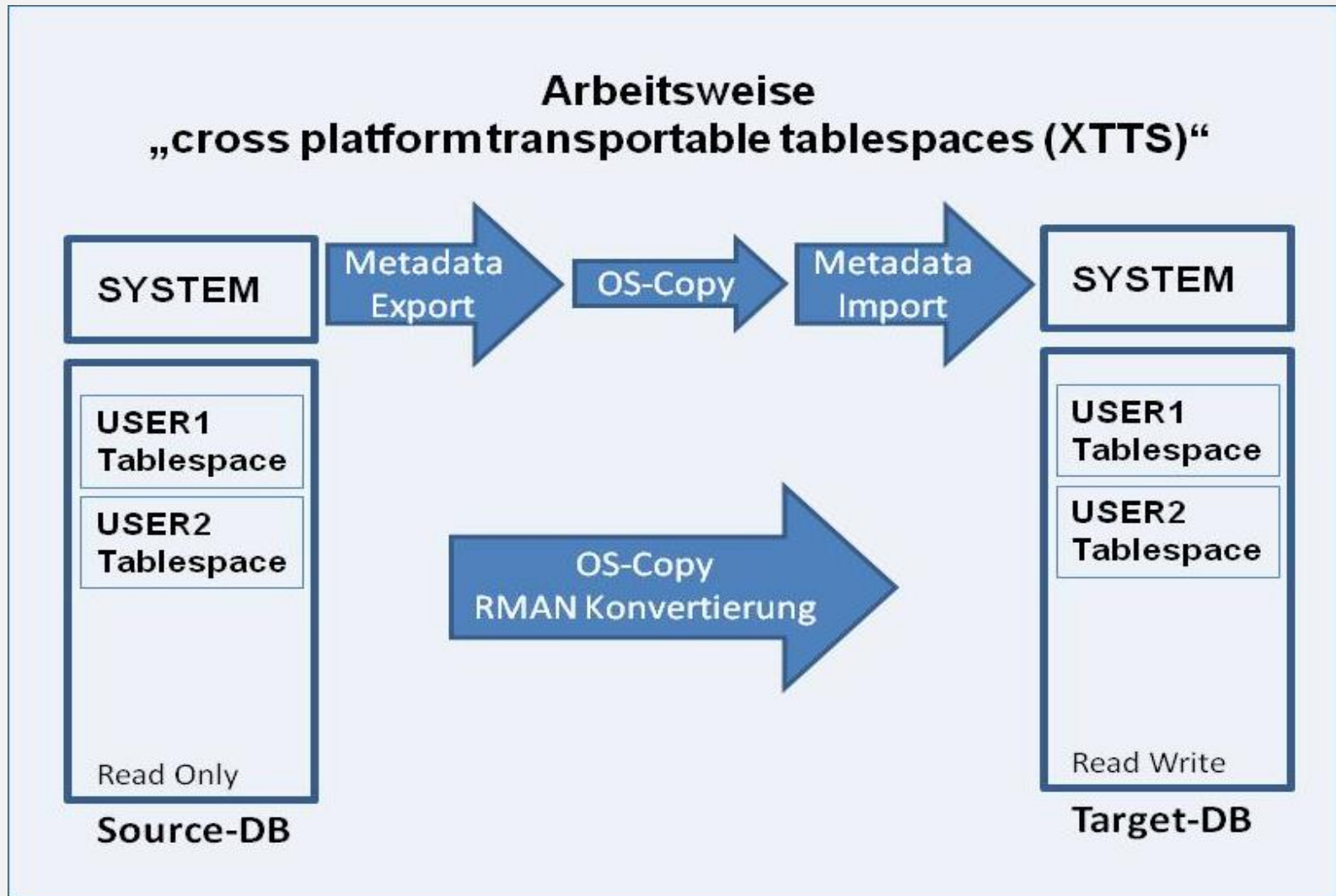
- Export/Import (bis 10g)
- Datapump
- Transportable Tablespaces
  - seit 8i
  - 10g über Plattformgrenzen
  - 10g R2 Transportable Database  
(same endian format)
- Oracle Data Guard
  - Metalink [ID 413484.1]: “Data Guard Support for Heterogeneous Primary and Physical Standbys in Same Data Guard Configuration”

# Grundidee: Migrationsmethode

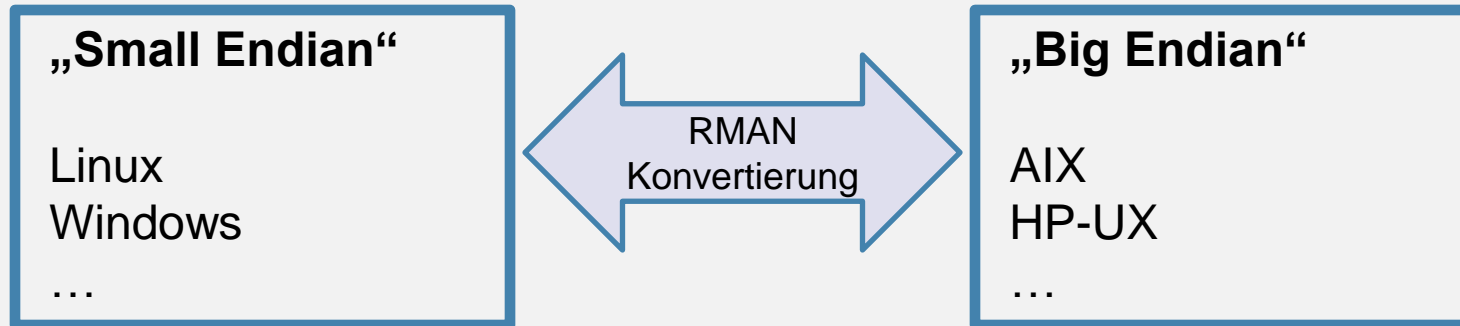
---

1. Neue, leere Datenbank auf dem Zielsystem anlegen
2. Meta-Daten extrahieren
3. Nutz-Daten (User-Tablespaces) für Zielplattform konvertieren
4. Nutz-Daten transportieren
5. Meta-Daten importieren

# Arbeitsweise „cross platform transportable tablespaces (XTTS)“



# Endian Typen



```
SQL> SELECT PLATFORM_NAME, ENDIAN_FORMAT  
2 FROM V$TRANSPORTABLE_PLATFORM order by PLATFORM_NAME;
```

PLATFORM_NAME	ENDIAN_FORMAT
AIX-Based Systems (64-bit)	Big
Apple Mac OS	Big
Apple Mac OS (x86-64)	Little
HP IA Open VMS	Little
HP Open VMS	Little
HP Tru64 UNIX	Little
HP-UX (64-bit)	Big
HP-UX IA (64-bit)	Big
IBM Power Based Linux	Big
IBM zSeries Based Linux	Big
Linux IA (32-bit)	Little
Linux IA (64-bit)	Little
Linux x86 64-bit	Little
Microsoft Windows IA (32-bit)	Little
Microsoft Windows IA (64-bit)	Little
Microsoft Windows x86 64-bit	Little
Solaris Operating System (x86)	Little
Solaris Operating System (x86-64)	Little
Solaris[tm] OE (32-bit)	Big
Solaris[tm] OE (64-bit)	Big

```
20 rows selected.
```

# RMAN Konvertierung

## RMAN-Kommando:

Wichtig: „CONFIGURE DEVICE TYPE DISK PARALLELISM ...”

```
CONVERT DATAFILE
```

```
  '/oracle/SRC/tools01.dbf',
```

```
  '/oracle/SRC/user01.dbf',
```

```
FROM PLATFORM 'Linux x86 64-bit'
```

```
PARALLELISM 2
```

```
DB_FILE_NAME_CONVERT '/oracle/SRC' , '/oracle/TGT'
```



# Tablespace Transport Check

---

```
SQL> exec dbms_tts.transport_set_check('TOOLS,  
USERS',TRUE,TRUE);
```

```
SQL> select * from transport_set_violations;
```

VIOLATIONS

-----

ORA-39908: Index DEFUSER.TEST\_PK in tablespace  
IDXTSP enforces primary constraints of table  
DEFUSER.ORA\_TEST\_TEMPLATE in tablespace  
TOOLS.

# Move procedures

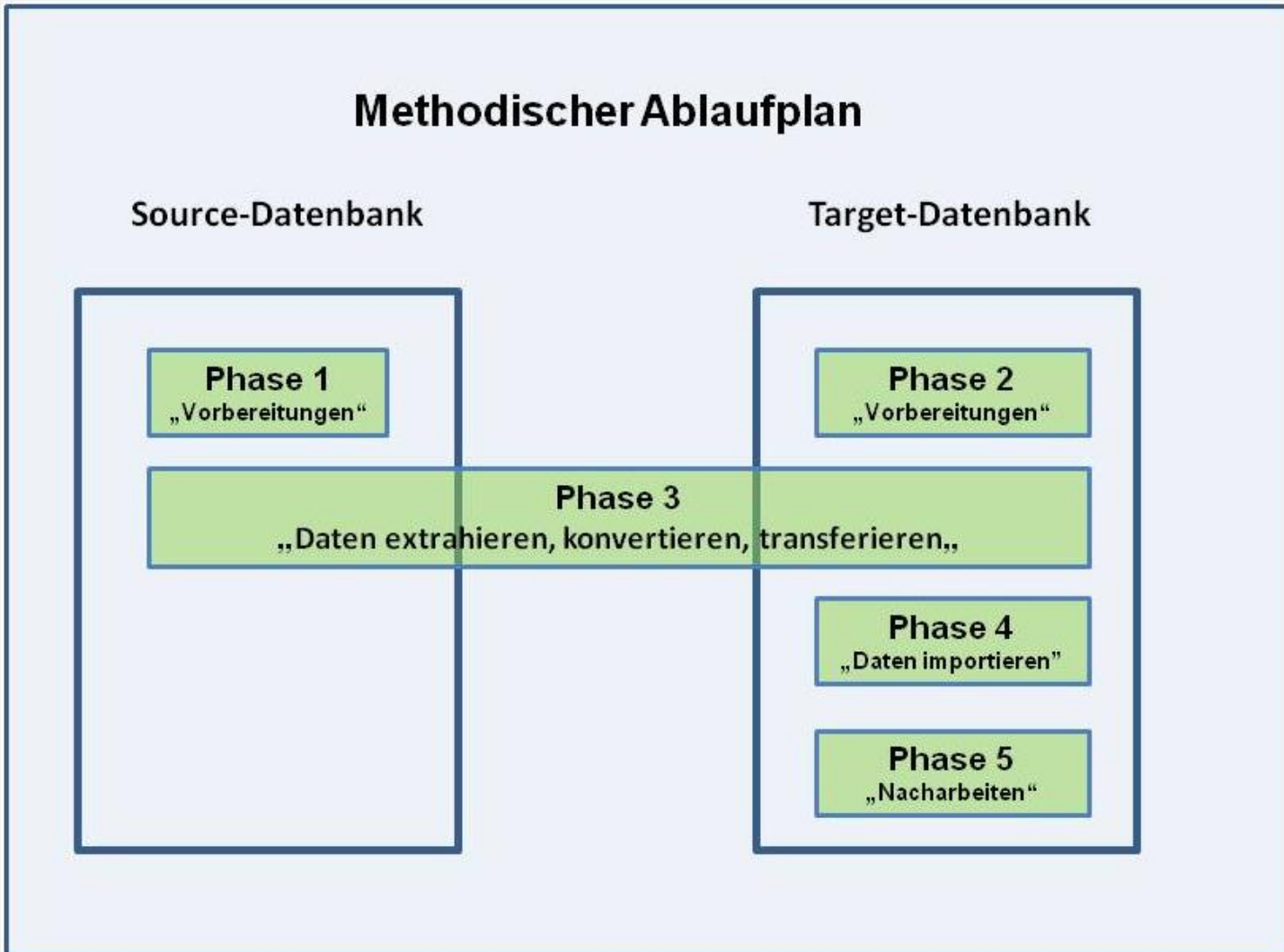
## ■ V\$SYSAUX\_OCCUPANT\_NAME

```
SQL> SELECT      occupant_name, schema_name, move_procedure, space_usage_kbytes USAGE_KB
  2 FROM          v$sysaux_occupants
  3 ORDER BY     occupant_name;
```

OCCUPANT_NAME	SCHEMA_NAME	MOVE_PROCEDURE	USAGE_KB
AO	SYS	DBMS_AW.MOVE_AWMETA	1408
AUDIT_TABLES	SYS	DBMS_AUDIT_MGMT.move_dbaudit_tables	0
AUTO_TASK	SYS		320
EM	SYSMAN	emd_maintenance.move_em_tblspc	0
EM_MONITORING_USER	DBSNMP		2432
EXPRESSION_FILTER	EXFSYS		0
JOB_SCHEDULER	SYS		384
LOGMNR	SYSTEM	SYS.DBMS_LOGMNR.D.SET_TABLESPACE	8064
LOGSTDBY	SYSTEM	SYS.DBMS_LOGSTDBY.SET_TABLESPACE	1408
ORDIM	ORDSYS	ordsys.ord_admin.move_ordim_tblspc	0
ORDIM/ORDDATA	ORDDATA	ordsys.ord_admin.move_ordim_tblspc	0
ORDIM/ORDPLUGINS	ORDPLUGINS	ordsys.ord_admin.move_ordim_tblspc	0
ORDIM/SI_INFORMTN_SCHEMA	SI_INFORMTN_SCHEMA	ordsys.ord_admin.move_ordim_tblspc	0
PL/SCOPE	SYS		1600
SDO	MDSYS	MDSYS.MOVE_SDO	0
SM/ADVISOR	SYS		24192
SM/AWR	SYS		85760
SM/OPTSTAT	SYS		59776
SM/OTHER	SYS		5952
SMON_SCN_TIME	SYS		3264
SQL_MANAGEMENT_BASE	SYS		1728
STATSPACK	PERFSTAT		0
STREAMS	SYS		1024
TEXT	CTXSYS	DRI_MOVE_CTXSYS	0
TSM	TSMSYS		0
ULTRASEARCH	WKSYS	MOVE_WK	0
ULTRASEARCH_DEMO_USER	WK_TEST	MOVE_WK	0
WM	WMSYS	DBMS_WM.move_proc	7360
XDB	XDB	XDB.DBMS_XDB.MOVEXDB_TABLESPACE	57728
XSAMD	OLAPSYS	DBMS_AMD.Move_OLAP_Catalog	0
XSOQHIST	SYS	DBMS_XSOQ.OlapiMoveProc	1408

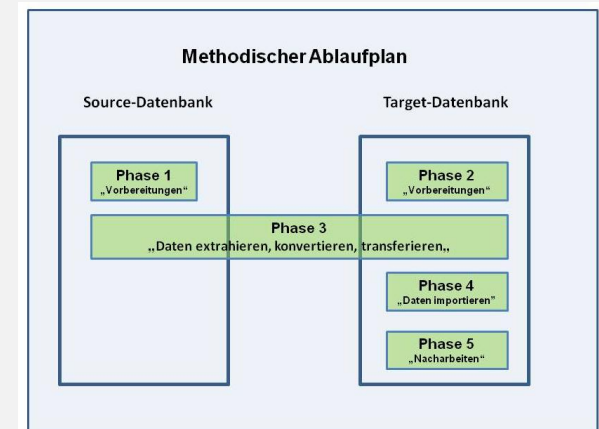
31 rows selected.

# Methodischer Ablaufplan



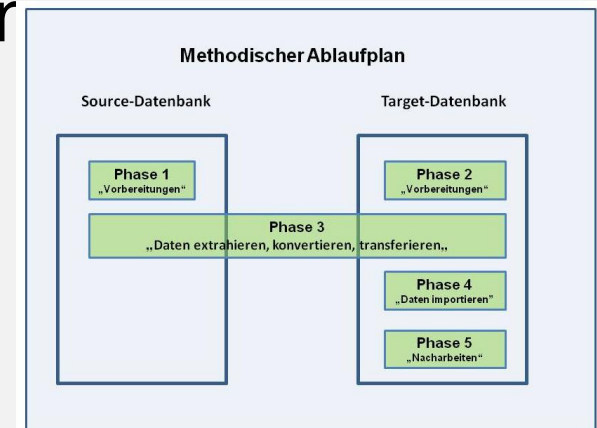
# Methodischer Ablaufplan – Phase1+2

- Ziel-Datenbank auf dem Zielsystem anlegen
- Character-Set, National-Character-Set
- compatible-Parameter
- V\$SYSAUX\_OCCUPANT\_NAME – Move procedures
- „self-contained“ prüfen
- dbms\_tdb.check\_external
- dbms\_lob.filegetname
- „Recyclebin“ leeren
- „invalide“ Objekte kompilieren



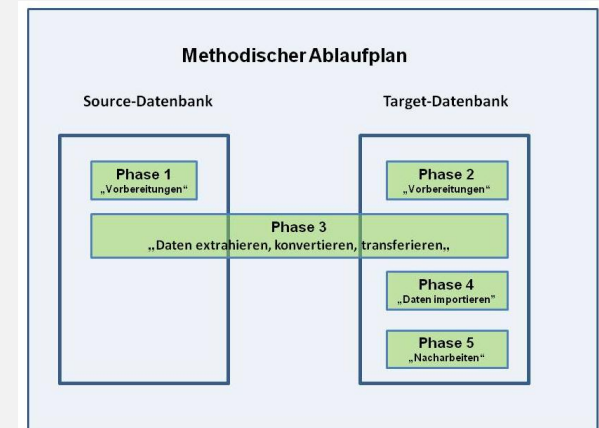
# Methodischer Ablaufplan – Phase 3

- Source-DB: Tablespaces „Read Only“
- Metadaten exportieren
- Non-Segment User-Daten exportieren
- User/Rollen/Sequenzen create-Skripte erzeugen
- Advanced Queueing Settings
- Weitere Daten (z. B. AWR) exportieren
- “datafiles”-Filesysteme mounten
- RMAN Konvertierung
- Externe Files (table, bfiles, ...) transferieren



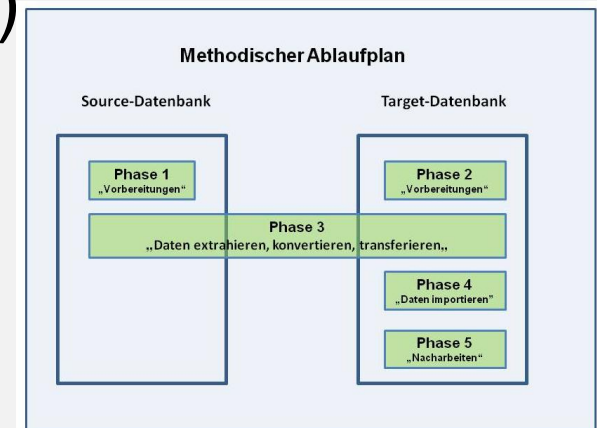
# Methodischer Ablaufplan – Phase 4

- User und Rollen anlegen
- Meta-Daten importieren
- Non-Segment User-Daten importieren
- Sonstige Daten (z. B. AWR) importieren
- Evtl. Advanced Queueing Settings anpassen



# Methodischer Ablaufplan – Phase 5

- tnsnames/LDAP anpassen
- Daten/Inhaltsvergleich zwischen Source- und Ziel-DB durchführen
- alert.log auf Fehler prüfen
- Backup durchführen
- Evtl. Memory-Check-Parameter entfernen
- Performance prüfen (SGA, etc.)
- Recovery Test durchführen



# Limitierungen (1)

- Character-Set, National-Character-Set
- „compatible“ Parameter
- Encryption-Columns werden nicht unterstützt
- „self-contained“ tablespaces
- Advanced Queue Einstellungen
- Binary\_Float and Binary\_Double (nur mit Datapump)
- XML Data (in 10g nur mit export/import)



# Limitierungen (2)

- SYSTEM-, UNDO-, SYSAUX-, TEMP-TSP Tabellen werden nicht transportiert
- SYS owned objects (packages, views, java classes, sequences, etc.) werden nicht transportiert
- Externe Objekte (parameter files, external tables, bfiles, etc.) müssen manuell bewegt und konvertiert werden
- ...

# Erfahrungsbericht (1)

- Vorbereitung, Vorbereitung, ...
- Bislang ca. 100 Datenbanken
- 5 GB – 2 TB
- Alle DBs wurden erfolgreich migriert
- Test-Migrationen durchführen
- Datenkorruption  
„db\_block\_checksum“ , „db\_block\_checking“

# Erfahrungsbericht (2)

- Tabellen mit dem Datentyp:
  - „ROWDEPENDENCIES“, „XML Datatyp“,
  - „BINARY\_FLOAT“, „BINARY\_DOUBLE“
- Oracle Text-Objekte auf dem Tablespace „SYSAUX“
- Oracle Text User Index Preferences
- Indexe auf Temp-Tables
- Komprimierte Tabellen
  - Korruptionen?
- Advanced Queueing Settings
- RMAN: „CONFIGURE DEVICE TYPE DISK PARALLELISM ...“

# Erfahrungsbericht (3) - Migrationsdauer

- Größe des Data-Dictionary
  - z. B. Partitioning Option
- CPU-Performance für die „endian“-Typ-Konvertierung mit dem RMAN
- Parallelität der Konvertierung
- I/O Subsystem
- Netz-Performance für den Filetransfer
  - RO-mount der Datenfiles zur Ziel-DB

# Erfahrungsbericht (4)

## ▪ SGA: DB Cache Size

- alter system set db\_cache\_advice = off scope=memory;
- SELECT size\_for\_estimate, buffers\_for\_estimate, estd\_physical\_read\_factor, estd\_physical\_reads/1000 FROM v\$db\_cache\_advice WHERE advice\_status = 'ON';

Cache Size (MB)	Buffers	Estd Phys Read Factor	ESTD_PHYSICAL_READS/1000
3,584	443,296	1.01	7070.031
4,032	498,708	1.00	7049.6
4,480	554,120	1.00	7022.63
4,928	609,532	.98	6890.872
5,376	664,944	.95	6646.521
5,824	720,356	.87	6140.577
6,272	775,768	.81	5722.296
6,720	831,180	.79	5564.413
7,168	886,592	.78	5501.977
7,616	942,004	.78	5494.624
8,064	997,416	.78	5480.898
8,512	1,052,828	.78	5476.055
8,960	1,108,240	.78	5448.579

# Erfahrungsbericht (5)

- Disk I/O optimieren
  - `FILESYSTEMIO_OPTIONS = { none | setall | directIO | asynch }`
- Linux (huge pages), Windows (large pages)
  - [http://en.wikipedia.org/wiki/Huge\\_pages#Huge\\_pages](http://en.wikipedia.org/wiki/Huge_pages#Huge_pages)
  - Information: `cat /proc/meminfo`
  - Metalink: HugePages on 64-bit Linux [ID 361468.1]

# Fallback Szenario 1

Source-Datenbank wird inhaltlich nicht modifiziert

- Einzige Modifikation:  
User-Tablespace „Read Only“
- Source-Datenbank einfach und schnell wieder verfügbar
- Datenverlust möglich!

# Fallback Szenario 2

---

Ziel-Datenbank wurde inhaltlich bereits modifiziert

- Rück-Konvertierung zur Source-DB möglich
- Erneute Downtime erforderlich
- Hohes Risiko, da Migrationsweg meist ungetestet



---

Danke für Ihre Aufmerksamkeit

Fragen ...