

ORACLE®



SAP-Migrationen mit Cross Platform Transportable Tablespaces

Jan Brosowski
Andris Perkons
Oracle Systems Sales Consulting Europe North

DOAG Konferenz 2017

Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Agenda

- 1 ➤ Was ist TTS? Warum TTS?
- 2 ➤ Durchführung
- 3 ➤ Tuning und Troubleshooting
- 4 ➤ Verwandte Verfahren
- 5 ➤ Dokumentation

Was ist TTS? Warum TTS?

Eine Einordnung

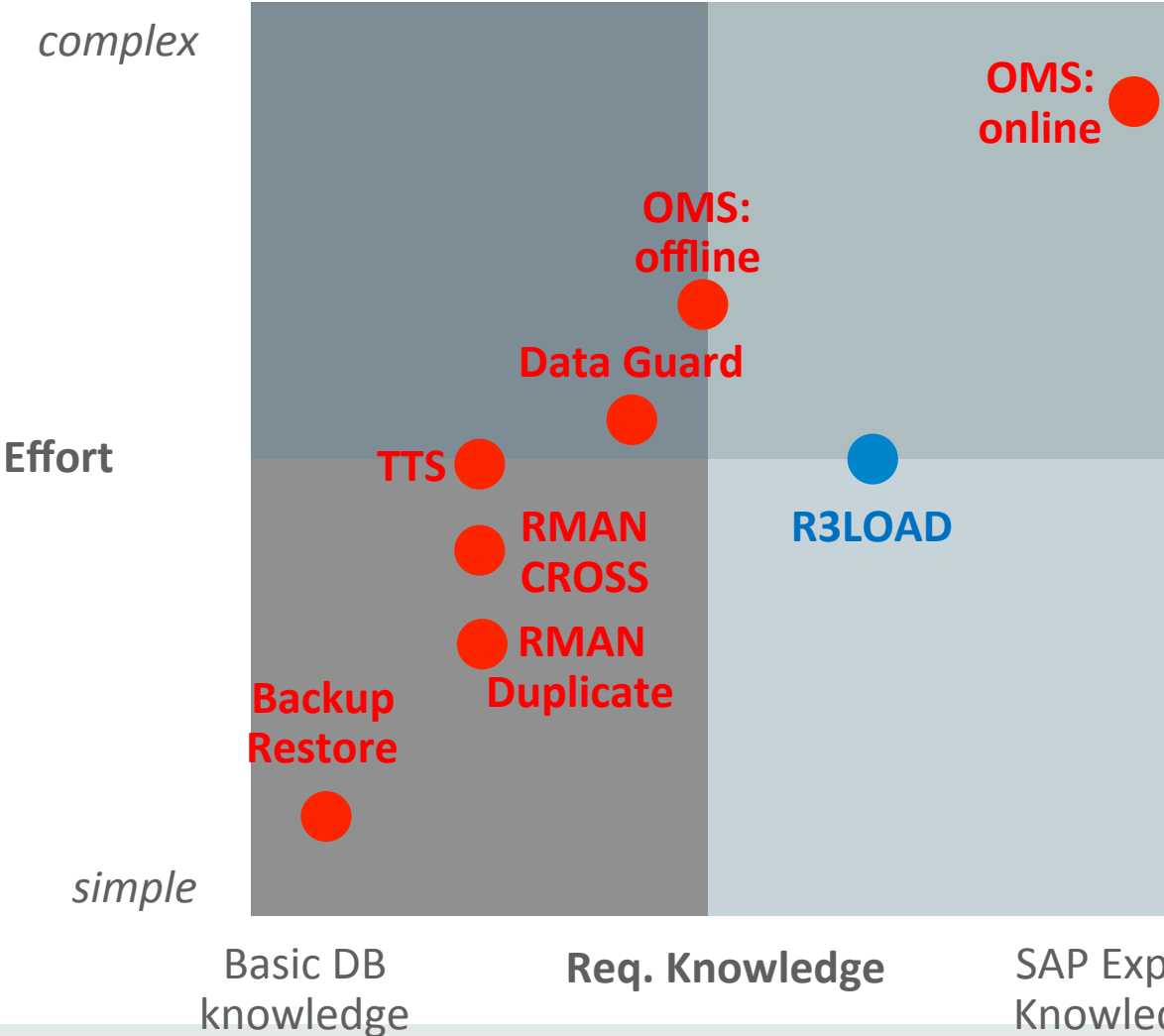
Was ist TTS? Warum TTS?

- Transportable Tablespaces
 - Transport ganzer Tablespaces von einer Oracle-DB in eine andere
 - Kein neues Verfahren. Seit 8i verfügbar, seitdem stetig verbessert
 - 10g: Cross Platform (gleiche Endianness)
 - 11g: Cross-endian, inkrementeller Transport
 - 12c: Cross-platform backup/restore
 - 12.2: weitere Verbesserungen (z.B. TDE verschlüsselte Tablespaces, Cross-endian)

Was ist TTS? Warum TTS?

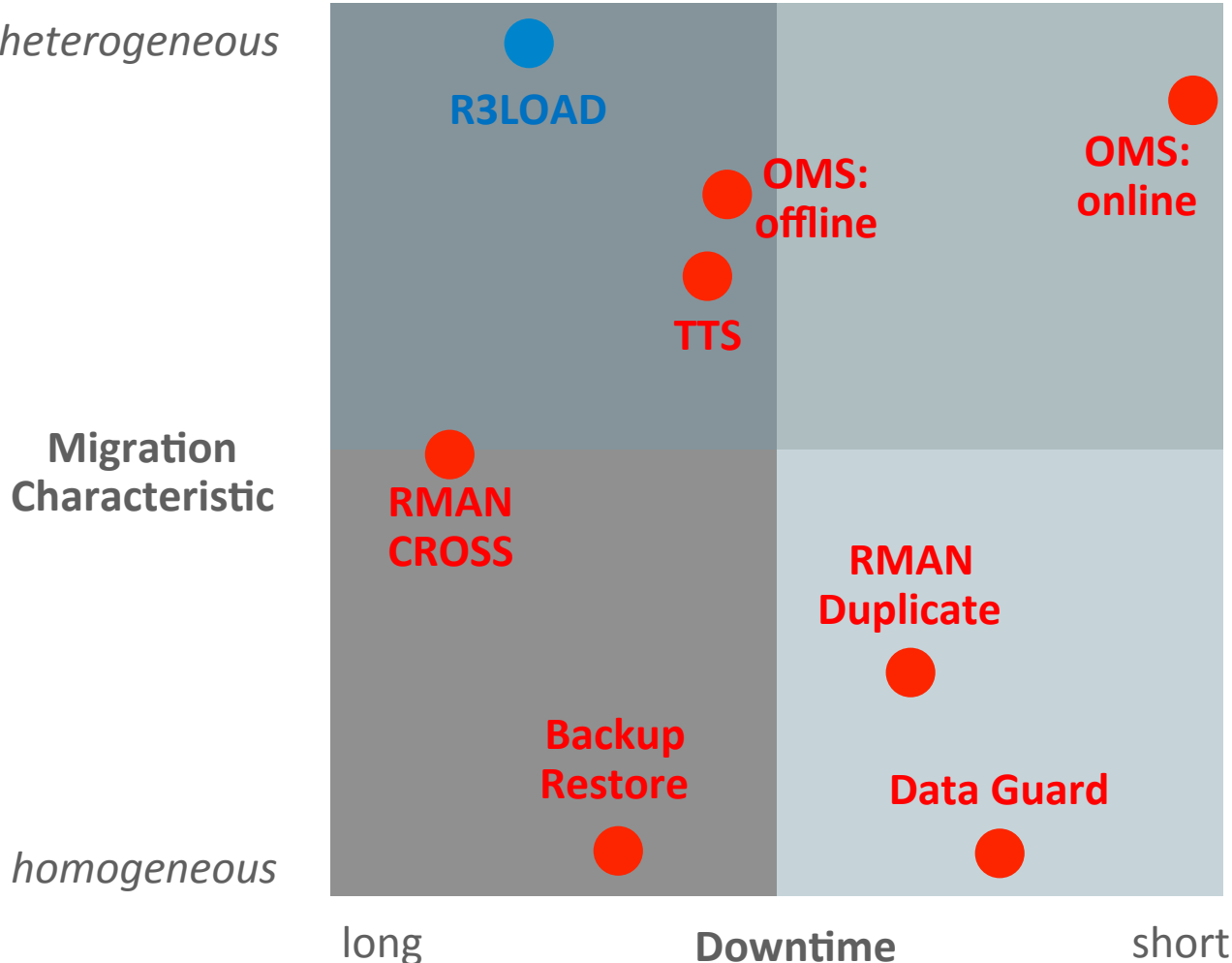
- Warum TTS?
 - Wenn's schnell gehen muss
 - Schnelle Bereitstellung von Testsystemen (z.B. Proof-of-Concept)
 - Kein Entladen/Laden von Daten
 - Wenn's einfach sein soll
 - Kein tiefes SAP-KnowHow notwendig
 - Insgesamt nicht besonders komplex
 - Kurze Projektlaufzeit
 - DB-Releaseupgrade implizit möglich

Einordnung: Aufwand und Komplexität



OMS:
Oracle Migration Service

Einordnung: Downtime und Heterogenität



OMS:
Oracle Migration Service

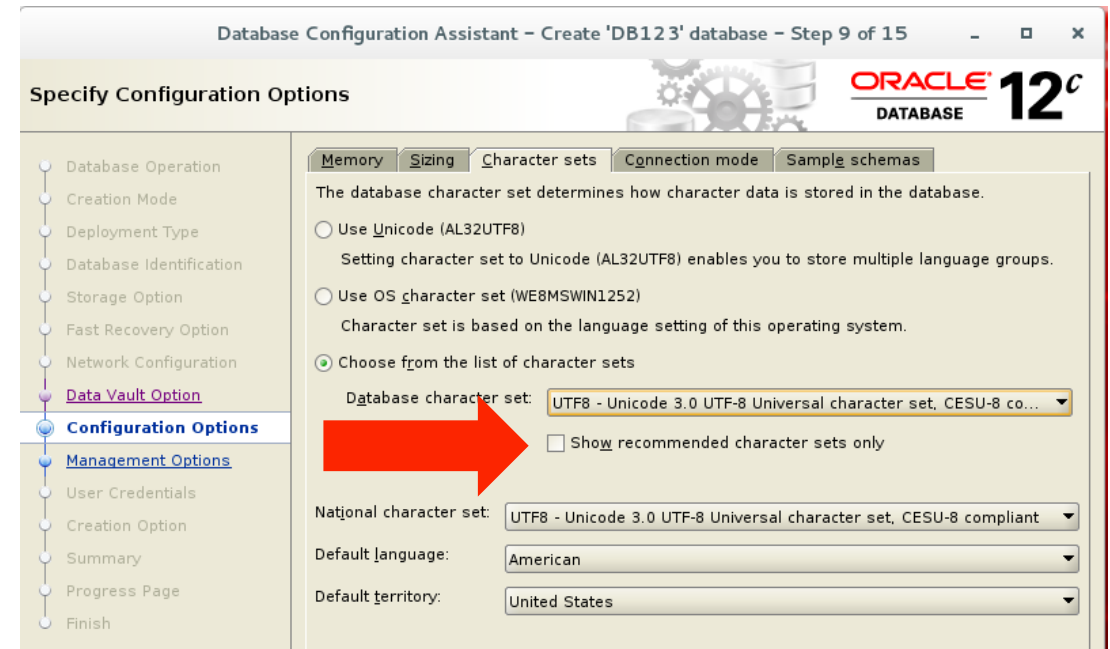


Voraussetzungen für TTS

- Characterset und National Characterset der Quell- und Ziel-DB müssen identisch sein

```
SQL> select value from v$nls_parameters  
where parameter in  
( 'NLS_CHARACTERSET', 'NLS_NCHAR_CHARACTERSET' );
```

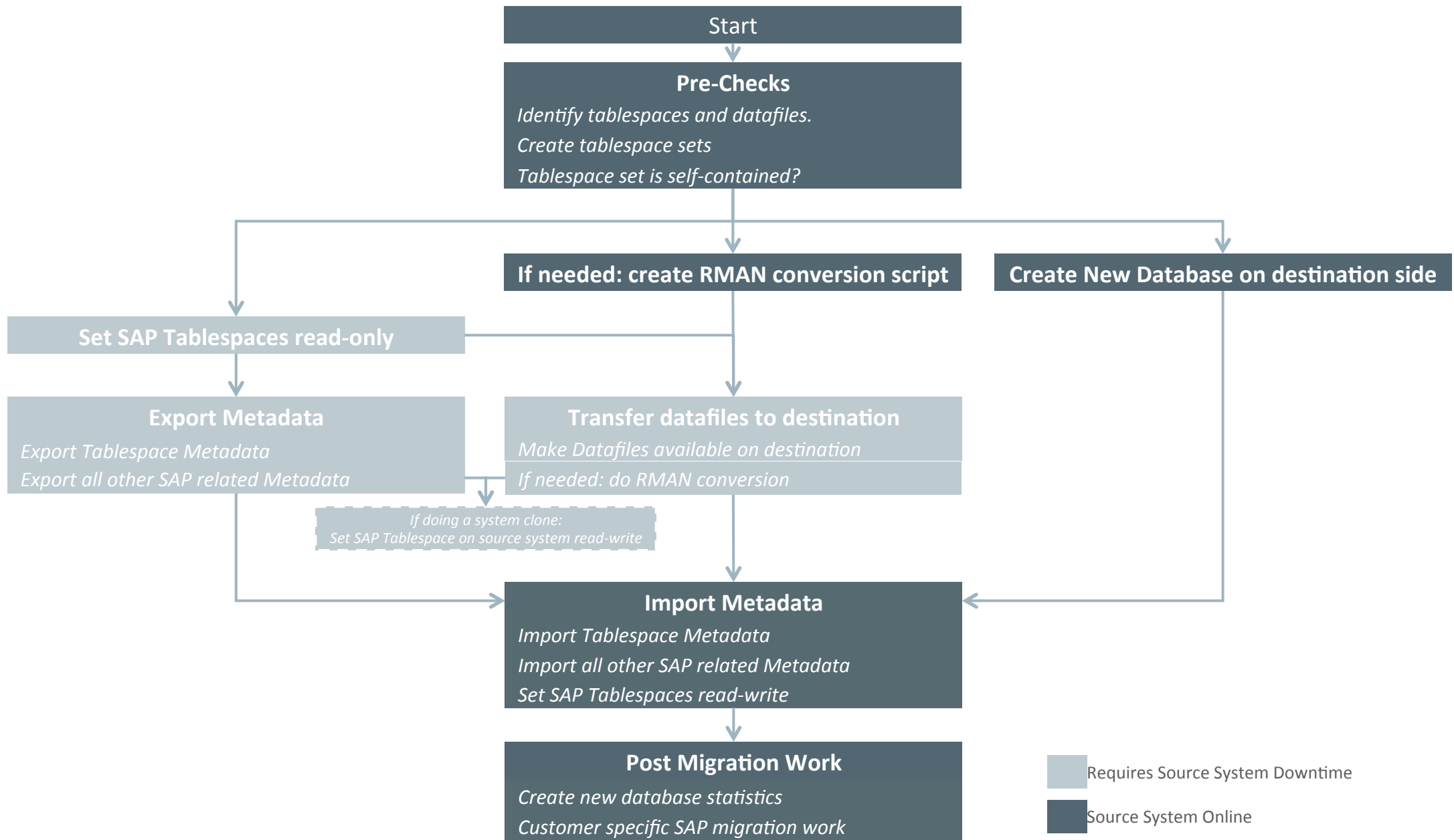
- Non-Unicode: WE8DEC, UTF8
- Unicode: UTF8, UTF8
- Bei Anlegen der Ziel-DB mit dbca:
 - “Advanced configuration” wählen



Voraussetzungen für TTS

- Einige Datentypen (XMLTypes) können nicht transportiert werden
 - Bei SAP-Datenbanken i.d.R. kein Problem
- Tabellen mit encrypted columns können nicht transportiert werden
 - TDE auf TSP-Ebene möglich
 - 11g: Gleiche Endianness
 - 12.2: Cross-Endian
- Objekte, die SYS gehören, werden nicht transportiert

Durchführung



Ablauf

- Vorbereitungen
- Export
- Transport
- Import

Vorbereitungen

- Ziel-DB anlegen

- Ziel-DB muss die selbe Timezone-Datei haben wie die Quelle

```
SQL> select version from v$timezone_file;
```

- Ggfs Upgrade der TZ-Datei durchführen

- DB-User anlegen

- SAPSR3, evtl. weitere User

- Achtung: Als “default tablespace” kann (vorübergehend) nicht PSAPSR3USR verwendet werden!

- Benötigte Rollen anlegen (z.B. SAPDBA und SAPCONN)

- SAP-Hinweise 134592, 834917

Vorbereitungen

- Transport-Sets festlegen
 - Transport-Set: Tablespaces, die gemeinsam transportiert werden sollen

- Prüfen

```
SQL> execute sys.dbms_tts.transport_set_check('TSP1,TSP2,...,TSPn',true);
```

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

```
VIOLATIONS
```

```
ORA-39907: Index SAPSR3.DKKOP~0 in tablespace PSAPSR3 points to table SAPSR3.DKKOP in tablespace PSAPSR3A1.
```

- Probleme beheben, sofern vorhanden (oder Transport-Set erweitern)

Exkurs: Wie viele Transport-Sets?

```
SQL> select tablespace_name, count(*) from dba_tables
where tablespace_name like 'PSAP%' group by
tablespace_name order by 2 desc;
```

TABLESPACE_NAME	COUNT (*)
PSAPSR3	27565
PSAPSR3700	339
PSAPSR3DB	310
PSAPSR3USR	9

Hier reicht ein Transport-Set, da im Prinzip alle Objekte auf einen Tablespace konzentriert sind

```
SQL> select tablespace_name, count(*) from dba_tables
where tablespace_name like 'PSAP%' group by
tablespace_name order by 2 desc;
```

TABLESPACE_NAME	COUNT (*)
PSAPSR3A2	44802
PSAPSR3A0	29212
PSAPSR3A3	11096
PSAPSR3	4018
PSAPSR3USR	3160
PSAPSR3A4	2813
PSAPSR3A1	2778
PSAPSR3701	389

Hier bieten sich vier Transport-Sets an

Vorbereitungen

- Database Directory erzeugen (Quell- und Ziel-DB)

```
SQL> create directory ttsp as '/oracle/DEM/ttsp';
```

- Weitere übliche Vorbereitungen
 - Einbindung / Test der Ziel-DB in Backup und Monitoring etc.

Export

- Parameter-Datei für Metadaten der Tablespaces

```
$ cat ttsexp.par
directory=ttsp
dumpfile=tbs.dmp
transport_tablespaces=PSAPSR3,PSAPSR3700,PSAPSR3USR
transport_full_check=Y
status=60
logfile=ttsexp.log
exclude=TABLE_STATISTICS,INDEX_STATISTICS
COMPRESSION=NONE
```

Export

- Parameter-Datei für restliche Metadaten

```
$ cat ttsexpdict.par
full=y
exclude=table:"in( select table_name from dba_tables where owner in
('SAPSR3','OPS$DEMADM'))"
exclude=tablespace, TABLE_STATISTICS, INDEX_STATISTICS
directory=ttsp
dumpfile=dict.dmp
status=60
logfile=ttsexpdict.log
COMPRESSION=NONE
```

Export

- SAP stoppen
- SAP-Tablespaces READ ONLY setzen
- Metadaten-Export starten

```
expdp "'/ as sysdba'" PARFILE=ttsexpdict.par >nohup.tts.out 2>&1 &  
expdp "'/ as sysdba'" PARFILE=ttsexp.par >nohup.dict.out 2>&1 &
```

- Hinweis: Die Laufzeit des Metadaten-Exports eines Transport-Sets ist abhängig von der Anzahl der Objekte

Transport

- Datafiles der SAP-Tablespaces kopieren
 - Kann parallel zum Export der Metadaten stattfinden
 - Z.B. auf einen NAS-Filer, der auf Quell- und Zielsystem sichtbar ist
 - Optimierung: Direkter Transport der Datafiles
 - Quell-DB auf Filesystem, Ziel auf Filesystem oder ASM
 - SSH/SCP (Filesystem -> Filesystem)
 - NFS-Freigabe der SAPDATA-Verzeichnisse des Quellsystems an Zielsystem (Ziel: ASM)
- Downtime für das Quellsystem kann mit Abschluss des Transports beendet werden

Transport

- Beispiel: Konversion von Filesystem auf ASM

```
$cat convert.par
CONVERT DATAFILE
'/net/oraldom/oracle/DEM/sapdata1/sr3700.data1',
(...)
'/net/oraldom/oracle/DEM/sapdata1/sr3.data9'
FROM PLATFORM='Solaris[tm] OE (64-bit)'
DB_FILE_NAME_CONVERT
  '/net/oraldom/oracle/DEM/sapdata1', '+DATA/DEM/DATAFILE'
PARALLEL=4;
```

```
$ rman target / cmdfile=convert.par log=convert.log
```



Quizfrage:



Was stimmt hier nicht?

Import

- Erst Tablespace-Metadaten ...

```
$ cat ttsimp.par
directory=ttsp
dumpfile=tbs.dmp
logfile=ttsimp.log
status=60
transport_datafiles='+DATA/DEM/DATAFILE/sr3700.data1',
(...)
'+DATA/DEM/DATAFILE/sr3.data9'

$ impdp ''/ as sysdba'' PARFILE=ttsimp.par
```


Import

- ... dann restl. Metadaten

```
$ cat ttspimpdict.par
```

```
directory=ttsp
```

```
Dumpfile=dict.dmp
```

```
logfile=ttspimpdict.log
```

```
$ impdp "'/ as sysdba'" PARFILE=ttsimp.par
```

Nacharbeiten

- Tablespaces READ WRITE setzen
- DB-Benutzer anpassen (default tablespace)
- Statistiken erzeugen
- “Übliche” Nacharbeiten nach SAP-Systemkopie

Tuning und Troubleshooting

Tuning

- Parameter gemäss SAP-Hinweis einpflegen*
 - “event=‘38068 level 100’”: Faktor 4 beim impdp (auf langsamer HW)

* 11.2.0.x: 1431798
12.1.0.2: 1888485

Tuning

- Auf Kompression verzichten
 - COMPRESSION=NONE (default: METADATA_ONLY)
 - Performancegewinn beim expdp: Bis zu Faktor 10 (alte HW)

Tuning

- Statistiken nicht transportieren
 - Quellsystem: `EXCLUDE=TABLE_STATISTICS,INDEX_STATISTICS`
 - Zielsystem: `_optimizer_compute_index_stats=false`
 - Statistiken nachträglich parallelisiert erzeugen

Tuning

- Parallelisieren
 - Metadatenexport kann bei TTSP nicht mit “PARALLEL=n” parallelisiert werden
 - Aber: export/import mehrerer Transportsets kann parallel erfolgen
 - Lohnt eher bei altem SAP-Tablespace-Layout

Troubleshooting

- Views

- DBA_DATAPUMP_JOBS

- Info zu DP jobs

- DBA_DATAPUMP_SESSIONS

- Info zu DP sessions
- Alternativ:

- `SQL> select process,event from v$session where module like 'Data Pump%';`

- V\$SESSION_LONGOPS

Troubleshooting

- SYSTEM-Tablespace läuft voll
 - Grund: Lob-Segmente, die zu den DP-Jobs gehören (z.B. nach Abbrüchen):

```
SQL> select table_name from dba_lobs where segment_name='SYS_LOB0001318070C00100$$';  
TABLE_NAME  
-----  
SYS_IMPORT_TRANSPORTABLE_02
```

- Beobachtung: Je 100k Objekten werden ca. 1,2GB Plattenplatz in SYSTEM benötigt

Troubleshooting

- SYSTEM-Tablespace läuft voll
 - Abhilfe lt. MOS 3306014.1:

```
SQL> SELECT owner_name, job_name, operation, job_mode,  
state, attached_sessions  
FROM dba_datapump_jobs  
where STATE='NOT RUNNING' and ATTACHED_SESSIONS=0;
```

```
SQL> drop table <name des DP-Jobs>
```

Troubleshooting

- Job wurde abgebrochen, aber DBA_DATAPUMP_JOBS sagt “EXECUTING”

```
SQL> SELECT job_name, operation, job_mode, state, attached_sessions  
FROM dba_datapump_jobs;
```

JOB_NAME	OPERATION	JOB_MODE	STATE	ATTACHED_SESSIONS
-----	-----	-----	-----	-----
SYS_EXPORT_TRANSPORTABLE_01	EXPORT	TRANSPORTABLE	EXECUTING	0

Troubleshooting

- Job wurde abgebrochen, aber DBA_DATAPUMP_JOBS sagt "EXECUTING"
- Abhilfe:

```
$ expdp "'/ as sysdba'" attach=SYS_EXPORT_TRANSPORTABLE_01
```

```
<output>
```

```
Export> kill_job
```

```
Are you sure you wish to stop this job ([yes]/no): yes
```

Verwandte Verfahren

Verwandte Verfahren, Verfeinerungen

- Cross Platform Incremental Backup
 - Ab 11g verfügbar
 - Verkürzt die Downtime im Prinzip auf den Metadatentransport
 - Vorgehen (verkürzt)
 - Vorbereitung: Datafiles auf Zielsystem übertragen, convert durchführen, wenn notwendig
 - „Roll Forward“-Phase: Inkrementelles Backup erzeugen und auf Quellsystem applizieren (ggfs. convert)
 - Transport-Phase:
 - Tablespaces auf Quellsystem READ ONLY setzen
 - Noch ein „Roll Forward“
 - Transport der Metadaten
 - Tablespaces auf Zielsystem READ WRITE setzen

Verwandte Verfahren, Verfeinerungen

- Cross Platform RMAN backup/restore
 - Ab 12c verfügbar
 - Im Prinzip das selbe wie TTSP, aber einfacher in der Handhabung
 - Läuft tendenziell länger

Verwandte Verfahren, Verfeinerungen

- Transportable Database
 - Schneller, da kein Metadaten-export/import
 - Nicht cross-endian möglich
 - Einige Objekte werden nicht transportiert:
 - TEMP Tablespaces
 - Externe Tabellen und Directories
 - PW-Files

Verwandte Verfahren, Verfeinerungen

- Full Transportable Export/Import (ab 12c)
 - Wie TTSP, aber transportiert auch alle nicht-Tablespace-Objekte (view, synonyms, grants, etc)
 - Importiert auch über DB-Links
 - NETWORK_LINK=<dblink_name>
 - Quellsystem muss mindestens 11.2.0.3 sein
 - Whitepaper:
 - <http://www.oracle.com/technetwork/database/enterprise-edition/full-transportable-wp-12c-1973971.pdf>

Verwandte Verfahren, Verfeinerungen

- BR-Tools-Unterstützung
 - Vorhanden, aber umständlich*

* Disclaimer: Habe das letzte Mal vor Jahren nachgesehen...

Dokumentation

Dokumentation

- **Transporting Tablespaces Between Databases (11g)**
https://docs.oracle.com/cd/E11882_01/server.112/e25494/tspaces.htm#ADMIN11394
- **Transporting Data Across Platforms (11g)**
https://docs.oracle.com/cd/E25054_01/backup.1111/e10642/rcmxplat.htm
- **Transporting Data (12.1)**
<https://docs.oracle.com/database/122/ADMIN/transporting-data.htm>
- SAP-Hinweis 1035051 - Transportable tablespaces
– Etwas alt, aber immer noch brauchbar

MOS-Notes

- Known Issues for Cross Platform Transportable Tablespaces XTTS (Doc ID 2311677.1)
- 11G - Reduce Transportable Tablespace Downtime using Cross Platform Incremental Backup (Doc ID 1389592.1)
- 12C - Reduce Transportable Tablespace Downtime using Cross Platform Incremental Backup (Doc ID 2005729.1)
- How to restore a pre-12c backup to a cross-platform, cross-endian 12c database (Doc ID 1644693.1)
- 12.2 RMAN Cross Platform Transport Of TDE-encrypted tablespace (Doc ID 2307358.1)
- 12.2 RMAN Cross Platform Tablespace Transport Over Network (Doc ID 2307383.1)

Links

- Immer wieder lesenswert:

<https://mikedietchde.com/>

Fokus primär auf DB-Upgrades, aber auch gute TTSP-Artikel, wie z.B:

<https://mikedietchde.com/2017/10/05/oow-2017-slides-download-migrate-100-tb-databases-less-one-day/>

Integrated Cloud

Applications & Platform Services

ORACLE®