

# Cross-Platform Backup & Restore mit RMAN

Alexander Wiese (Oracle-SAP Database Operations, Postbank Systems AG)

DOAG Konferenz 2018



# Agenda

1. Kurzvorstellung der Postbank
2. Endianess Problematik bei einer Cross-Platform Migration
3. RMAN New Features 12c
4. Vorbereitung für Cross-Platform Migration
5. Migration einer Datenbank von AIX nach Linux
6. Fazit

# 1. Kurzvorstellung der Postbank / Postbank-Systems

# DB Privat- und Firmenkundenbank AG als größte Säule von PCB



## Private & Commercial Bank (PCB)

DB Privat- und Firmenkundenbank AG



Die Bank für Deutschland.

### Fokus

Standardisierte Finanzdienstleistungen für Privat- Geschäfts- und Firmenkunden, bundesweite Bargeldversorgung, Post- und Paketdienste, Liquiditäts- und Risikomanagement für Firmenkunden

Private & Commercial Clients  
International

### Fokus

Beratungsgeschäft, insbesondere Wertpapiere und Business Banking für Privat- und Firmenkunden in ausgewählten Märkten in Europa (Italien, Spanien, Belgien, Polen\*, Portugal\*) und Indien

Wealth Management

### Fokus

Planung, Verwaltung und Anlage von Vermögen & Finanzierung für wohlhabende, vermögende und sehr vermögende Privatkunden in Deutschland und international (Europa/Naher Osten, Asien, USA)

\* Verkauf angekündigt, um Geschäft zu fokussieren und Komplexität zu reduzieren

# Einzigartiges Angebot für Privat-, Geschäfts- und Firmenkunden

13 Millionen  
Kunden

~ 700 Tsd. Kunden  
täglich vor Ort



9 Millionen Telefonbanking-  
konten

Vor Ort & digital  
für unsere Kunden erreichbar



5 Millionen  
private Girokonten

Vielfach ausgezeichnetes  
Online-Banking



~ 2.500 mobile Finanzberater  
und Immobilienmakler

~ 5.500 Standorte bundesweit



9 Millionen Onlinebanking-  
konten

300.000  
Firmen- und Geschäftskunden



~ 850  
Postbank  
Filialen

# Oracle Database Administration bei der Postbank Systems AG

- Dediziertes Teams innerhalb der Infrastruktur Organisation (1 x Non-SAP, 1 x SAP)
- Separation of Duty (Funktionstrennung)
- Betrieb von ~1000 Datenbanken (Produktion, Qualitätssicherung, Entwicklungs-/Testsysteme) → Davon ~250 SAP Systeme
- AIX und Linux basierte Systeme
- Projektarbeit/Tagesgeschäft:
  - Daily DBA Business
  - Patching / Lifecycle
  - Backup & Recovery
  - Troubleshooting & Performance Analyse / Optimierung
  - Migration und Umzüge

## 2. Endianess Problematik bei einer Cross-Platform Migration

# Endianess

- Beschreibt das Speicherformat für Daten
- Big Endian / Little Endian
- Big Endian speichert die Daten nach Signifikanz absteigend.
  - Analog der deutschen Uhrzeit: Stunde:Minute: Sekunde
  - z.B. AIX (POWER), Solaris (SPARC)
- Little Endian speichert die Daten nach Signifikanz aufsteigend.
  - Analog des deutschen Datum: Tag.Monat.Jahr
  - z.B. Linux, Windows (x86)
- Daten können über Kreuz nicht adhoc gelesen werden

Sie müssen **konvertiert** werden → Export/Import oder RMAN



“Indian-ness”

Millard E. Carr / M. Eric Carr (2009)



# Klassisches Cross-Platform Transportable Tablespaces Verfahren (XTTS)

1. Tablespaces auf der Quelle auf READ ONLY setzen
2. Transport der Datafiles auf das Zielsystem (z.B. mittels Betriebssystem Kopierverfahren)
3. Konvertierung der Datafiles auf das Ziel-Endian Format (z.B. mittels RMAN)
4. Export der Metadaten auf der Quelle (mittels DataPump)
5. Import der Metadaten auf dem Ziel (mittels DataPump)
6. Tablespaces im Ziel auf READ WRITE setzen und Konsistenzprüfungen durchführen (physikalisch + logisch)

Dieses Verfahren ist relativ **aufwändig** und benötigt eine komplette **Downtime** des Quellsystem!

Alternative: Export/Import oder **RMAN**

### 3. RMAN New Features 12c (12.1 + 12.2)

# RMAN New Features 12.1

## Cross-Platform Backup and Restore Enhancements:

Es können Daten zwischen verschiedenen Plattformen mittels `FULL` und `INCREMENTAL` Backup transportiert werden. `INCREMENTAL` Backups können die gesamte Downtime während einer cross-platform Migration deutlich reduzieren.

## Restore & Recover Files over Network:

RMAN kann Backup Sets über das Netzwerk zum Zielhost transportieren. Dies ist u.A. sinnvoll, um eine Standby Datenbank mittels DataGuard initial aufzubauen bzw. Backup Sets zwischen zwei Servern über das Netzwerk direkt zu transportieren.

# RMAN New Features 12.2



## Cross-Platform Support over the Network :

Support für `ACTIVE DUPLICATE` mittels cross-platform Transport. Implizit kann der `RESTORE` nun cross-platform erfolgen. Hierfür wurde die Syntax im RMAN um `FROM SERVICE` und `RESTORE FROM PLATFORM <PLATFORM>` erweitert.

→ Limitierung auf Datafiles (kein FULL DATABASE ☹)

**Lösung:** Eine Kombination aus allen drei New Features von 12.1 und 12.2

## 4. Vorbereitung für Cross-Plattform Migration

# Vorbereitungen

- Oracle 12.2 RDBMS Installation auf Quelle und Ziel (Empfehlung min. **12.2.0.1.180717**)
- "Leere" Database Instance auf Ziel muss angelegt sein:

```
dbca -silent -createDatabase -gdbName THC -templateName template.dbt  
      -sysPassword # -systemPassword # -dbsnmpPassword #
```

- Netzwerkverbindung über Kreuz mittels Database Link zwischen Quelle und Ziel muss eingerichtet sein
- Schema-User (Data Owner) muss auf Ziel analog Quelle angelegt sein (SYSTEM Tablespace)
- Ein DataPump Directory muss auf Quelle und Ziel eingerichtet sein
- ggf. Reorganisation oder Tablespace Konsolidierung auf der Quelle
- Statistiken auf Quelle erzeugen:

```
EXEC DBMS_STATS.GATHER_FIXED_OBJECTS_STATS ();  
EXEC DBMS_STATS.GATHER_DICTIONARY_STATS ();
```

- ggf. OPSS\$-User auf Quelle löschen 

## 5. Cross-Platform Migration einer Datenbank von AIX nach Linux

# Cross-Platform Migration – Prepare 3

- **Variante 1:**
  - A) SET SOURCE TABLESPACES READ ONLY
  - B) RESTORE ALL DATAFILES
  - Sinnvoll bei kleineren Datenbanken
  
- **Variante 2:**
  - A) RESTORE ALL DATAFILES
  - B) SET SOURCE TABLESPACES READ ONLY
  - C) RECOVER TABLESPACES
  - Sinnvoll bei größeren Datenbanken



# Cross-Platform Migration – Prepare 1

- Vorbereitungen abgeschlossen ✓
- Zusätzliche ToDo's:

## # Falsche Segmente in Tablespaces:

```
sqlplus / as sysdba << EOF
select distinct owner from dba_segments@THC_AIX WHERE tablespace_name like 'PSAP%' AND tablespace_name NOT IN
('PSAPUNDO','PSAPTEMP');
EOF
```

## # Directories anlegen:

```
sqlplus / as sysdba << EOF
set pages 0
SELECT DISTINCT 'mkdir -p '||SUBSTR(file_name,0, INSTR(file_name,'/',-1,1)-1) FROM dba_data_files@THC_AIX WHERE
tablespace_name like 'PSAP%' AND tablespace_name NOT IN ('PSAPUNDO','PSAPTEMP');
EOF
```

# Cross-Platform Migration – Prepare 2

# DataFile Directories auf Ziel anlegen: 

```
sqlplus / as sysdba << EOF
```

```
set pages 0
```

```
SELECT DISTINCT 'mkdir -p '||SUBSTR(file_name,0, INSTR(file_name,'/',-1,1)-1) FROM dba_data_files@THC_AIX WHERE  
tablespace_name like 'PSAP%' AND tablespace_name NOT IN ('PSAPUNDO','PSAPTEMP');  
EOF
```

# Output:

```
mkdir -p /oracle/THC/sapdata1/hcm_6  
mkdir -p /oracle/THC/sapdata1/hcm_7  
mkdir -p /oracle/THC/sapdata1/hcm_8  
mkdir -p /oracle/THC/sapdata1/hcm_4  
mkdir -p /oracle/THC/sapdata1/hcmusr_1  
mkdir -p /oracle/THC/sapdata1/hcm750_1  
mkdir -p /oracle/THC/sapdata1/hcm_10  
mkdir -p /oracle/THC/sapdata1/hcm_3  
mkdir -p /oracle/THC/sapdata1/hcm_1  
mkdir -p /oracle/THC/sapdata1/hcm_5  
mkdir -p /oracle/THC/sapdata1/hcmusr_2  
mkdir -p /oracle/THC/sapdata1/hcm_2  
mkdir -p /oracle/THC/sapdata1/hcm_9
```

**Wahlweise auch Migration von/nach ASM oder Nutzung von OMF!**

# Cross-Platform Migration – Variante 1

```
# Set Source Tablespaces Read-only:

sqlplus / as sysdba << EOF
SET SERVEROUT ON
DECLARE
CURSOR cur_ts_names IS
    SELECT tablespace_name, status FROM dba_tablespaces WHERE tablespace_name like 'PSAP%' AND tablespace_name
NOT IN ('PSAPUNDO','PSAPTEMP') ORDER BY 1;
BEGIN
    FOR set IN cur_ts_names LOOP
        EXECUTE IMMEDIATE 'ALTER TABLESPACE '||set.tablespace_name||' READ ONLY';
    END LOOP;
    FOR set IN cur_ts_names LOOP
        dbms_output.put_line(CHR(10)||set.tablespace_name||': '||set.status);
    END LOOP;
END;
/
EOF
```

# Cross-Platform Migration – Variante 1

```
sqlplus / as sysdba << EOF
SET SERVEROUT ON
SET LINES 200
DECLARE
  v_string VARCHAR2(1000);
  v_network_link VARCHAR2(20):='THC_AIX';
BEGIN
  v_string := '
  DECLARE
    v_network_link VARCHAR2(20):=''||v_network_link||'';
  BEGIN
    dbms_output.put_line(chr(10)||'RUN{''||chr(10)||'ALLOCATE CHANNEL ch1 TYPE DISK;''});
    FOR set IN (SELECT file_id,file_name FROM dba_data_files@''||v_network_link||' WHERE tablespace_name
      LIKE ''PSAP%'' AND tablespace_name NOT IN (SELECT tablespace_name from dba_data_files where tablespace_name
      LIKE ''PSAPUNDO%'' OR tablespace_name LIKE ''PSAPTEMP%'') ORDER BY 1)
      LOOP dbms_output.put_line('RESTORE FOREIGN DATAFILE ''||set.file_id||'
        FORMAT ''||CHR(39)||set.file_name||CHR(39)||' FROM SERVICE ''||v_network_link||'';''');
      END LOOP;
    dbms_output.put_line('}');
  END;';
  EXECUTE IMMEDIATE v_string;
END;
/
EOF
```

# Cross-Platform Migration – Variante 1

# Output:

```
RUN{
ALLOCATE CHANNEL ch1 TYPE DISK;
RESTORE FOREIGN DATAFILE 4 FORMAT '/oracle/THC/sapdata1/hcm_1/hcm.data1' FROM SERVICE THC_AIX;
RESTORE FOREIGN DATAFILE 6 FORMAT '/oracle/THC/sapdata1/hcmusr_1/hcmusr.data1' FROM SERVICE THC_AIX;
RESTORE FOREIGN DATAFILE 8 FORMAT '/oracle/THC/sapdata1/hcm_2/hcm.data2' FROM SERVICE THC_AIX;
RESTORE FOREIGN DATAFILE 9 FORMAT '/oracle/THC/sapdata1/hcm_3/hcm.data3' FROM SERVICE THC_AIX;
RESTORE FOREIGN DATAFILE 15 FORMAT '/oracle/THC/sapdata1/hcm_4/hcm.data4' FROM SERVICE THC_AIX;
RESTORE FOREIGN DATAFILE 16 FORMAT '/oracle/THC/sapdata1/hcm_5/hcm.data5' FROM SERVICE THC_AIX;
RESTORE FOREIGN DATAFILE 17 FORMAT '/oracle/THC/sapdata1/hcm_6/hcm.data6' FROM SERVICE THC_AIX;
RESTORE FOREIGN DATAFILE 18 FORMAT '/oracle/THC/sapdata1/hcm_7/hcm.data7' FROM SERVICE THC_AIX;
RESTORE FOREIGN DATAFILE 19 FORMAT '/oracle/THC/sapdata1/hcm_8/hcm.data8' FROM SERVICE THC_AIX;
RESTORE FOREIGN DATAFILE 20 FORMAT '/oracle/THC/sapdata1/hcmusr_2/hcmusr.data2' FROM SERVICE THC_AIX;
RESTORE FOREIGN DATAFILE 21 FORMAT '/oracle/THC/sapdata1/hcm750_1/hcm750.data1' FROM SERVICE THC_AIX;
RESTORE FOREIGN DATAFILE 22 FORMAT '/oracle/THC/sapdata1/hcm_9/hcm.data9' FROM SERVICE THC_AIX;
RESTORE FOREIGN DATAFILE 23 FORMAT '/oracle/THC/sapdata1/hcm_10/hcm.data10' FROM SERVICE THC_AIX;
}
```

# Cross-Platform Migration – Variante 1

```
Starting restore at 03-NOV-18
```

```
channel ch1: starting datafile backup set restore  
channel ch1: using network backup set from service THC_AIX  
channel ch1: specifying datafile(s) to restore from backup set  
channel ch1: restoring foreign file 4 to /oracle/THC/sapdata1/hcm_1/hcm.data1  
channel ch1: restore complete, elapsed time: 00:03:35  
Finished restore at 03-NOV-18
```

```
Starting restore at 03-NOV-18
```

```
channel ch1: starting datafile backup set restore  
channel ch1: using network backup set from service PHC_AIX  
channel ch1: specifying datafile(s) to restore from backup set  
channel ch1: restoring foreign file 6 to /oracle/THC/sapdata1/hcmusr_1/hcmusr.data1  
channel ch1: restore complete, elapsed time: 00:00:08  
Finished restore at 03-NOV-18
```

```
...
```

# Cross-Platform Migration – Variante 2

```
# Set Source Tablespaces Read-only:

sqlplus / as sysdba << EOF
SET SERVEROUT ON
DECLARE
CURSOR cur_ts_names IS
    SELECT tablespace_name, status FROM dba_tablespaces WHERE tablespace_name like 'PSAP%' AND tablespace_name
NOT IN ('PSAPUNDO','PSAPTEMP') ORDER BY 1;
BEGIN
    FOR set IN cur_ts_names LOOP
        EXECUTE IMMEDIATE 'ALTER TABLESPACE '||set.tablespace_name||' READ ONLY';
    END LOOP;
    FOR set IN cur_ts_names LOOP
        dbms_output.put_line(CHR(10)||set.tablespace_name||': '||set.status);
    END LOOP;
END;
/
EOF
```

# Cross-Platform Migration – Variante 2

```
RUN{
ALLOCATE CHANNEL ch1 TYPE DISK;
RECOVER FOREIGN DATAFILE 3 FORMAT '/oracle/THC/sapdata1/undo_1/undo.data1' FROM SERVICE THC_AIX;
RECOVER FOREIGN DATAFILE 4 FORMAT '/oracle/THC/sapdata1/hcm_1/hcm.data1' FROM SERVICE THC_AIX;
RECOVER FOREIGN DATAFILE 6 FORMAT '/oracle/THC/sapdata1/hcmusr_1/hcmusr.data1' FROM SERVICE THC_AIX;
RECOVER FOREIGN DATAFILE 8 FORMAT '/oracle/THC/sapdata1/hcm_2/hcm.data2' FROM SERVICE THC_AIX;
RECOVER FOREIGN DATAFILE 9 FORMAT '/oracle/THC/sapdata1/hcm_3/hcm.data3' FROM SERVICE THC_AIX;
RECOVER FOREIGN DATAFILE 10 FORMAT '/oracle/THC/sapdata1/hcm701x_1/hcm701x.data1' FROM SERVICE THC_AIX;
RECOVER FOREIGN DATAFILE 11 FORMAT '/oracle/THC/sapdata1/hcm701x_2/hcm701x.data2' FROM SERVICE THC_AIX;
RECOVER FOREIGN DATAFILE 12 FORMAT '/oracle/THC/sapdata1/hcm701x_3/hcm701x.data3' FROM SERVICE THC_AIX;
RECOVER FOREIGN DATAFILE 13 FORMAT '/oracle/THC/sapdata1/hcm701x_4/hcm701x.data4' FROM SERVICE THC_AIX;
}
```



# Cross-Platform Migration – Metadaten 1

## - DataPump Import der Metadaten (parfile\*):

```
NETWORK_LINK=THC_AIX
LOGTIME=ALL
LOGFILE=impdp_meta.log
DIRECTORY=DP_DIR
EXCLUDE=TABLE_STATISTICS,INDEX_STATISTICS
TRANSPORT_TABLESPACES=PSAPHCM
TRANSPORT_TABLESPACES=PSAPHCM750
TRANSPORT_TABLESPACES=PSAPHCMUSR
TRANSPORT_DATAFILES='/oracle/THC/sapdata1/hcm750_1/hcm750.data1'
TRANSPORT_DATAFILES='/oracle/THC/sapdata1/hcm_1/hcm.data1'
TRANSPORT_DATAFILES='/oracle/THC/sapdata1/hcm_10/hcm.data10'
TRANSPORT_DATAFILES='/oracle/THC/sapdata1/hcm_2/hcm.data2'
TRANSPORT_DATAFILES='/oracle/THC/sapdata1/hcm_3/hcm.data3'
TRANSPORT_DATAFILES='/oracle/THC/sapdata1/hcm_4/hcm.data4'
TRANSPORT_DATAFILES='/oracle/THC/sapdata1/hcm_5/hcm.data5'
TRANSPORT_DATAFILES='/oracle/THC/sapdata1/hcm_6/hcm.data6'
TRANSPORT_DATAFILES='/oracle/THC/sapdata1/hcm_7/hcm.data7'
TRANSPORT_DATAFILES='/oracle/THC/sapdata1/hcm_8/hcm.data8'
TRANSPORT_DATAFILES='/oracle/THC/sapdata1/hcm_9/hcm.data9'
TRANSPORT_DATAFILES='/oracle/THC/sapdata1/hcmusr_1/hcmusr.data1'
TRANSPORT_DATAFILES='/oracle/THC/sapdata1/hcmusr_2/hcmusr.data2'
```

\*generate-script s. Anhang

# Cross-Platform Migration – Metadaten 2

```
# Fortschritt monitoren:
while true; do
sqlplus / as sysdba << EOF
set heading off
set feedback off
select 'Quelle: ' || object_type || ' Anzahl: ' || count(*) from dba_objects@THC_AIX where owner = 'SAPHCM' group by
object_type order by object_type;
select 'Ziel: ' || object_type || ' Anzahl: ' || count(*) from dba_objects where owner = 'SAPHCM' group by
object_type order by object_type;
EOF
sleep 60
Done
```

# Cross-Platform Migration – Metadaten 2

Quelle: INDEX Anzahl: **101104**

Quelle: LOB Anzahl: **2738**

Quelle: SEQUENCE Anzahl: **1**

Quelle: TABLE Anzahl: **82646**

Quelle: VIEW Anzahl: **10243**

Ziel: INDEX Anzahl: **101104**

Ziel: LOB Anzahl: **2738**

Ziel: SEQUENCE Anzahl: **1**

Ziel: TABLE Anzahl: **82646**

Ziel: VIEW Anzahl: **10243**

# Cross-Platform Migration – Metadaten 3

## - Bei Global Temporary Tables:

```
set pages 0
set long 2000000
spool /tmp/gtt.sql
SELECT dbms_metadata.get_ddl('TABLE', table_name, owner) || ';'
FROM dba_tables
WHERE temporary = 'Y'
AND owner = 'SAPHCM'
ORDER BY owner, table_name;

set pages 0
set long 2000000
SET LONGCHUNKSIZE 900
set linesize 900
Spool /tmp/gti.sql
SELECT dbms_metadata.get_ddl('INDEX', index_name, owner) || ';'
FROM dba_indexes
WHERE temporary = 'Y'
AND owner = 'SAPHCM'
ORDER BY owner, index_name;
```

# Cross-Platform Migration – Post Tasks 1

- Neue Dictionary Statistiken erzeugen
- Object Statistiken ebenfalls neu erzeugen oder per STATTAB exportieren/importieren
- Tablespace Check:

```
rman target /  
report schema;  
validate tablespace PSAPHCM,PSAPHCMUSR, PSAPHCM750;
```

- Tablespaces READ, WRITE:

```
SET SERVEROUT ON  
DECLARE  
CURSOR cur_ts_names IS  
    SELECT tablespace_name, status FROM dba_tablespaces WHERE tablespace_name like 'PSAP%' AND tablespace_name NOT IN  
        ('PSAPUNDO','PSAPTEMP') ORDER BY 1;  
BEGINFOR set IN cur_ts_names LOOP  
    EXECUTE IMMEDIATE 'ALTER TABLESPACE '||set.tablespace_name||' READ WRITE';  
END LOOP;  
FOR set IN cur_ts_names LOOP  
    dbms_output.put_line(CHR(10)||set.tablespace_name||': '||set.status);  
END LOOP;  
END;  
/
```

# Cross-Platform Migration – Post Tasks 2

- Default Tablespace des Schema Users anpassen
- Initiales Backup durchführen
- Fachliche Abnahmetests durchführen

# Weitere Cross-Platform Alternative

- FULL + INCREMENTAL Backup via RMAN in Filesystem (Direct via Tape z.B. TSM → **NOT SUPPORTED by IBM!**)
- DataPump Export/Import via Flatfile
- Maximale Downtime Reduzierung
- Aufwändiges Verfahren und schnelles shared-Filesystem notwendig

# Fazit

## Vorteile:

- Oracle 12c bietet sehr gute Feature für eine Cross-Platform Migration
- Downtime kann durch neue Features deutlich reduziert werden
- Einfaches Verfahren mit RMAN und DataPump via Network

## Nachteile:

- Laufzeit abhängig von Anzahl Objekten (Tables, Indexes etc.)
- Multisection und Multiplexing not supported until now ("Workaround: Multisession")
- Oracle 12.2 NewFeature for DataPump: Parallel MetaData Import → Not supported for DataPump via Network Link ☹️

## Wünsche für zukünftige Versionen:

- Cross-Platform DUPLICATE FULL DATABASE FROM ACTIVE
- Multisection Cross-Platform Restore & Recover



## Links:

<https://docs.oracle.com/en/database/oracle/oracle-database/12.2/rcmrf/index.html>

→ Oracle RMAN Documentation

<https://docs.oracle.com/en/database/oracle/oracle-database/12.2/bradv/rman-transporting-data-across-platforms.html#GUID-6D2F4279-B6CC-4BB3-AE56-4D3D40A4CBF3>

→ Oracle RMAN Documentation – Performing Cross-Platform Transport of Data Files over the network

<http://martycodes.blogspot.com/#!/2014/07/oracle-transportable-tablespaces.html>

→ BlogEntry for Transportable Tablespace upgrade Step-by-Step

# My Oracle Support

12c How Perform Cross-Platform Database Transport to different Endian Platform with RMAN Backup Sets (**Doc ID 2013271.1**)

12C - Reduce Transportable Tablespace Downtime using Cross Platform Incremental Backup (**Doc ID 2005729.1**)

Transportable Tablespace Method Does Not Transport Global Temporary Tables (**Doc ID 2369318.1**)

12.2 RMAN Cross Platform Tablespace Transport Over Network (**Doc ID 2307383.1**)

How to Migrate to different Endian Platform Using Transportable Tablespaces With RMAN (**Doc ID 371556.1**)

## Known Bugs:

27504770 - ORA-600 [ksfdhardchk\_err] by XTTS Compressed Backup

→ Patch available. Fixed in 19.1.0

25100063 - ORA-600, ORA-7745 by RMAN Backup to Platform or Restore when a large number of Tablespaces is Specified

→ Patch available (s. 27504770)

24332831 - RMAN: ORA-600 [ktu\_format\_nr no disk align] while convert from HP-UX

→ Patch available. Fixed in 18.1.0

25890056 - Fix for Bug 25670970 on 12.2 - ORA-600 [25027] on Securefile Lob Segment after XTTS

→ Patch available. Fixed in 18.1.0

Danke ...



**Kontaktdaten:**

Alexander Wiese  
Postbank Systems AG  
Lubahnstraße 2  
D-31789 Hameln

[alexander.wiese@postbank.de](mailto:alexander.wiese@postbank.de)

# Anhang - I

## - Generate Script für DataPump Parfile Teil 1:

```
set serveroutput on
set lines 32767
DECLARE
    v_statement VARCHAR2(32000);
    v_result VARCHAR2(32000);
    --#####
    --#NETWORK LINK anpassen
    --#####
    v_network_link VARCHAR2(20):='THC_AIX';
    out_parfile UTL_FILE.FILE_TYPE;
    dyncur SYS_REFCURSOR;
BEGIN
    out_parfile := utl_file.fopen('DP_DIR', 'impdp_meta.par' , 'W');
    utl_file.put_line(out_parfile, 'NETWORK_LINK='||v_network_link);
    utl_file.put_line(out_parfile, 'LOGTIME=ALL');
    utl_file.put_line(out_parfile, 'LOGFILE=impdp_meta.log');
    utl_file.put_line(out_parfile, 'DIRECTORY=DP_DIR');
    utl_file.put_line(out_parfile, 'EXCLUDE=TABLE_STATISTICS,INDEX_STATISTICS');
    OPEN dyncur FOR
        'SELECT tablespace_name FROM dba_tablespaces@'||v_network_link||
        ' WHERE tablespace_name like ''PSAP%'' AND tablespace_name NOT IN (''PSAPUNDO'', ''PSAPTEMP'') ORDER BY 1';
```

# Anhang - I

## - Generate Script für DataPump Parfile Teil 2:

```
        LOOP
            FETCH dyncur INTO v_result;
            EXIT WHEN dyncur%NOTFOUND;
            utl_file.put_line(out_parfile,'TRANSPORT_TABLESPACES='||v_result);
        END LOOP;
    CLOSE dyncur;
    OPEN dyncur FOR
        'SELECT file_name FROM dba_data_files@'||v_network_link||
        ' WHERE tablespace_name LIKE ''PSAP%'' AND tablespace_name NOT IN (''PSAPUNDO'', ''PSAPTEMP'') ORDER BY 1';
    LOOP
        FETCH dyncur INTO v_result;
        EXIT WHEN dyncur%NOTFOUND;
        utl_file.put_line(out_parfile,'TRANSPORT_DATAFILES='''||v_result||''');
    END LOOP;
    CLOSE dyncur;
    utl_file.fclose(out_parfile);
END;
/
```