trivadis

xTTS – Lesson(s) learned

rmoug.org

Christian Gohmann

- **Principal Consultant** at Trivadis – Part of Accenture, Düsseldorf
- **Instructor** since 2014
  - O-AI (Oracle Architecture and Internals)
  - O-AI-DBA (Oracle Architecture and Internals for DBAs)
- **Tool Owner** of TVD-Backup

- Experiences with Oracle since 2006 (8i – 21c)
  - Architecture, Installation & Configuration
  - High Availability Solutions (RAC, Data Guard, GoldenGate)
  - Migration Projects
  - Backup & Recovery (RMAN, Data Pump)
  - Cloud (Amazon, Oracle)
- **Oracle ACE**

@CHGohmannDE

www.christian-gohmann.de

trivadis

FOUNDED IN
1994

16 TRIVADIS WORKSPACES

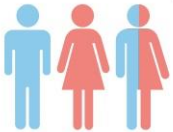
SWITZERLAND, GERMANY, AUSTRIA, DENMARK, ROMANIA

118 MILLION CHF
TURNOVER

300 SLA's
(SERVICE LEVEL AGREEMENTS)

4000
TRAINING PARTICIPANTS PER YEAR

800
CUSTOMERS



700
EMPLOYEES

5 MILLION CHF
BUDGET FOR SCIENCE AND DEVELOPMENT PER YEAR

EXPERIENCE FROM
1900 PROJECTS
PER YEAR

trivadis

Agenda

1. Introduction
2. Preparation Tasks
3. Metadata Export
4. Data File Transportation
5. Metadata Import
6. Post Tasks
7. Pitfalls

trivadis

Introduction

Initial Situation

trivadis

- Migration of **more than 100 databases from 11.2.0.4 to 19c**
  - Sizes of the databases between 10 GB and 25 TB
  - Stages: development, test, production
- **New hardware** (server, storage), but the **same endianness**
- Implementation of the **CDB architecture**
- **Downtime** of the production databases was limited to a **maximum of 2 hours**

Limitations

trivadis

- (National) character set between source and target have to be equal or compatible
  - To use compatible character sets (strict (binary) subset), special requirements have to be fulfilled
  - For national character set, no columns of the types NCHAR, NVARCHAR2, NCLOB cannot exist

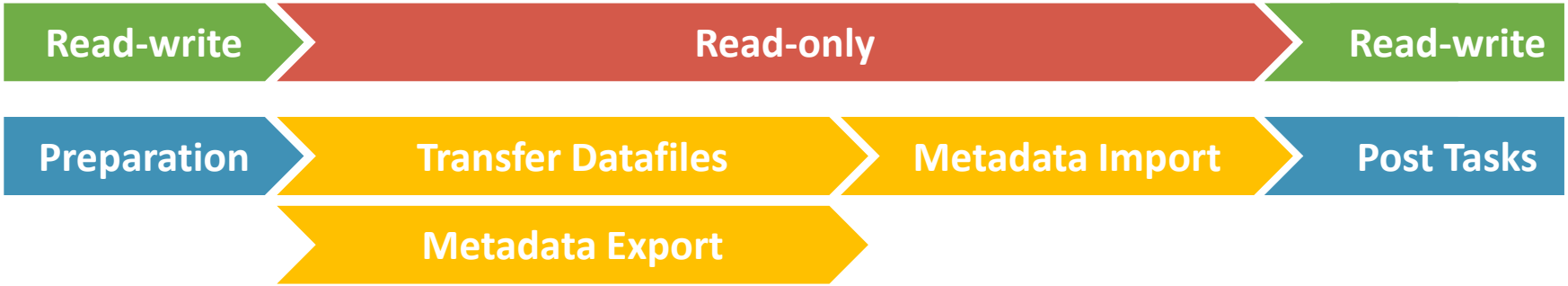
```
ORA-39123: Data Pump transportable tablespace job aborted
ORA-29345: cannot plug a tablespace into a database using an incompatible character set
```

- Tablespaces in the target with the same have to be dropped/renamed
  - Or use REMAP_TABLESPACE during the import
- To transport columns of the type TIMESTAMP WITH LOCAL TIME ZONE (TSLTZ), the database time zone has to be equal
- Target database version (parameter COMPATIBLE) has to be equal or higher than the source version
- Objects owned by SYS cannot be transported

Check MOS notes 1166564.1 and 1454872.1 for more restrictions and details.

Workflow

trivadis



- Transport set check
- Create target DB
- Create objects and users
- Perform dry-run of the metadata export (19c)
- Restore level 0 backup on the target site

- Set tablespaces read-only
- Transfer datafiles to the target site or recover final consistent incremental backup
- Export metadata

- Import metadata
- Set tablespaces read-write

- Gather statistics
- Correct OMF names
- Adjust default tablespace
- Recover standby (if required)

trivadis

Preparation Tasks

List of Tablespaces

trivadis

- Use the following query to get the list of user-defined Tablespaces

```
SQL> SELECT tablespace_name, block_size, extent_management, bigfile, encrypted
       FROM dba_tablespaces
      WHERE contents = 'PERMANENT'
            AND tablespace_name NOT IN (
                  SELECT default_tablespace FROM dba_users
                   WHERE username IN (
                            SELECT schema FROM dba_registry))
   ORDER BY tablespace_name;


TABLESPACE_NAME                BLOCK_SIZE EXTENT_MAN BIG ENC
------------------------------ ---------- ---------- --- ---
AUDIT_DATA                           8192 LOCAL      NO  NO
HIST_ARCHIVE                        16384 LOCAL      YES NO
HR_DATA                              8192 LOCAL      NO  YES
PERFSTAT_TS                          8192 LOCAL      NO  NO
USERS                                8192 LOCAL      NO  NO
```

Non-default blocksize

False postive

Transport Set Check

trivadis

- Check if the selected Tablespaces are **self-contained** – **no dependencies** to other Tablespaces

```
SQL> BEGIN
   DBMS_TTS.TRANSPORT_SET_CHECK(
      ts_list => 'AUDIT_DATA,HIST_ARCHIVE,HR_DATA,USERS',
      incl_constraints => TRUE,
      full_check => TRUE
   );
END;
/

SQL> SELECT * FROM transport_set_violations;

VIOLATIONS
--------------------------------------------------------------------------------
ORA-39907: Index HR.EMPLOYEES_IDX in tablespace HR_IDX points to table HR.EMPLOYEES in
tablespace HR_DATA.
ORA-39908: Index HR.EMPLOYEES_UNQ_CON in tablespace HR_CONS enforces primary constraints
of table HR.EMPLOYYEES in tablespace HR_DATA.
```

Two-way dependencies check

Index Tablespace not part of set

Use parameter TTS_CLOSURE_CHECK to perform or skip transport set check during the Data Pump export operation.

Check for UNUSED Columns

trivadis

- If tables with UNUSED columns exist in the database, drop these columns

```
SQL> ALTER TABLE my_tab DROP UNUSED COLUMNS;
```

- Otherwise, the following error is raised during the metadata import

```
ORA-39083: Object type TABLE:"APP"."MY_TAB" failed to create with error:
ORA-00904: "SYS_C00111_18092915:37:55$": invalid identifier
```

- Use the following query to find all tables with UNUSED columns

```
SQL> SELECT * FROM dba_unused_col_tabs ORDER BY owner, table_name
```

An unused column has no column ID and is renamed to SYS_C00002_21081310:09:11$.

Former column ID

Timestamp of the change

Create Target DB/PDB

trivadis

- It is recommended to use the **same character set between source and target**

```
SQL> SELECT property_name, property_value FROM database_properties
      WHERE property_name LIKE '%CHARACTERSET';

PROPERTY_NAME                  PROPERTY_VALUE
------------------------------ ------------------------------
NLS_CHARACTERSET               WE8MSWIN1252
NLS_NCHAR_CHARACTERSET         AL16UTF16
```

- In the case of a CDB, the PDB can use a **subset character set** of the CDB
  - Use AL32UTF8 as character set for the CDB
- PDBs with a different character set **cannot be created directly**
  - Instead an unplugged **PDB archive or a remote clone** is required
  - Create a **dummy CDB** with the correct character set
- Verify/install **required database components**

Create Objects

trivadis

- As preparation for the user creation and the transport, dependent objects have to be created

Directories DBA_DIRETORIES	**Global Temporary Tables** DBA_TABLES	**Network ACLs** DBA_NETWORK_ACLS
Profiles DBA_PROFILES	**Public Database Links** DBA_DB_LINKS	**Public Synonyms** DBA_SYNONYMS
Roles DBA_ROLES	**Scheduler Job Classes** DBA_SCHEDULER_JOB_CLASSES	**SQL Plan Baselines** DBA_SQL_PLAN_BASELINES
SQL Profiles DBA_SQL_PROFILES		

- Data Pump Export/Import can be used to transport these objects
  - Or use DBMS_METADATA.GET_DDL to create DDL statements (SYS is required)

```
SQL> SELECT DBMS_METADATA.GET_DDL('PROFILE', 'MY_PROFILE' FROM dual;
```

When DBMS_METADATA is used for database links, the passwords are obfuscated (Doc ID 1905221.1).

Create User 1/2

trivadis

- Create all required users in the target database
  - Otherwise, ORA-29342 is raised during the metadata import

```
ORA-39123: Data Pump transportable tablespace job aborted
ORA-29342: user HR does not exist in the database
```

- A temporarily remapping of the default tablespace is required

```
$> impdp … DUMPFILE = users.dmp REMAP_TABLESPACE = HR_DATA:DUMMY_TS,…
```

- Data Pump does not import grants on SYS objects
  - MOS Note: Data Pump: GRANTs On SYS Owned Objects Are Not Transferred With Data Pump And Are Missing In The Target Database (Doc ID 1911151.1)
  - To avoid errors during the compilation of PL/SQL objects and views, grant the missing grants manually

Easiest way to transport the users is to use Data Pump.

Create User 2/2

trivadis

- Query to generate GRANT commands for all grants on SYS objects

```
SQL> SELECT 'GRANT ' || privilege || ' "SYS"."' || table_name ||
            '" TO "' || grantee || '";' AS "GRANT_COMMAND"
       FROM dba_tab_privs
      WHERE owner = 'SYS'
            AND grantee IN (SELECT username
                              FROM dba_users
                             WHERE oracle_maintained = 'N');

GRANT_COMMAND
--------------------------------------------------------------------------------
GRANT SELECT "SYS"."V_$SESSION" TO "APP_USER";
```

New column in Oracle 12c

Starting with Oracle 12c, use the column ORACLE_MAINTAINED of DBA_USERS to exclude system users like SYS, SYSTEM, DBSNMP, etc.

trivadis

Metadata Export

Test Mode for Transportable Tablespaces

trivadis

- **Estimation of the required time** for the metadata export was not possible before 19c without setting the **tablespaces** to **read-only**

```
Data Pump transportable tablespace job aborted
ORA-39185: The transportable tablespace failure list is

ORA-29335: tablespace 'HR_DATA' is not read only
```

- Starting with 19c, a **test mode as part of the closure check** was added

```
$> expdp … TRANSPORTABLE_TABLESPACES = HR_DATA,USERS TTS_CLOSURE_CHECK = TEST_MODE
```

- Generated dump file **cannot be used for the metadata import**

```
Dump file set is unusable. TEST_MODE requested.
```

Parallelize Metadata Operations

trivadis

- Before 21c, only one Data Pump worker was supported

```
ORA-39002: invalid operation
ORA-39047: Jobs of type TRANSPORTABLE cannot use multiple execution streams.
```

- In 21c, all defined Data Pump workers (**PARALLEL** parameter) are exporting/importing metadata
- Each worker processes **one type of metadata** at the same time

```
20-APR-21 08:28:27.212: W-1 Startup on instance 1 took 1 seconds
20-APR-21 08:28:29.079: W-2 Startup on instance 1 took 1 seconds
...
20-APR-21 08:29:02.686: W-1 Processing object type TRANSPORTABLE_EXPORT/TABLE
20-APR-21 08:29:04.601: W-2 Processing object type TRANSPORTABLE_EXPORT/CONSTRAINT/CONSTRAINT
20-APR-21 08:29:04.671: W-2      Completed 50 CONSTRAINT objects in 0 seconds
20-APR-21 08:29:11.621: W-2 Processing object type TRANSPORTABLE_EXPORT/POST_INSTANCE/PROCACT_INSTANCE
20-APR-21 08:29:11.650: W-2      Completed 15 PROCACT_INSTANCE objects in 0 seconds
20-APR-21 08:29:12.304: W-2 Processing object type TRANSPORTABLE_EXPORT/POST_INSTANCE/PROCDEPOBJ
20-APR-21 08:29:12.334: W-2      Completed 10 PROCDEPOBJ objects in 0 seconds
20-APR-21 08:29:47.203: W-1      Completed 57 TABLE objects in 0 seconds
```

Set Tablespaces read-only

trivadis

- Before the metadata export can be executed, all involved tablespace have to be switched to read-only

```
SQL> ALTER TABLESPACE HR_DATA READ ONLY;
```

- This command will hang indefinitely, if active transactions exist
  - Internal wait event is *unbound tx*
- Check if pending in-doubt transactions exist and terminate them

```
SQL> SELECT * FROM dba_2pc_pending;
```

If possible, deactivate the listener and restart the database (in restricted mode) before starting the whole operation.

Metadata Export

trivadis

- Starting with Oracle 10g Data Pump (expdp) is used to export the metadata of the transportable tablespaces
  - Role DATAPUMP_EXP_FULL_DATABASE is required

```
$> expdp … TRANSPORT_TABLESPACES = TS1,TS2
```

- Exclude of statistics with EXCLUDE = STATISTICS does not work in this mode (Doc ID 1517267.1)

```
$> expdp … EXCLUDE = TABLE_STATISTICS,INDEX_STATISTICS,USER_PREF_STATISTICS
```

- For older databases the legacy Export Utility (exp) has to be used

```
$> exp … TRANSPORT_TABLESPACE = Y TABLESPACES = TS1,TS2
```

Exclude statistics to speed up the metadata export.

trivadis

Data File Transportation

DBMS_FILE_TRANSFER 1/2

trivadis

- Can be used to **transfer datafiles** between source and target using **database links**
  - ASM → ASM, ASM → Filesystem, Filesystem → ASM, Filesystem → Filesystem
- Introduced with Oracle 10g
- **Converts the endianness automatically** during the transfer
- **DIRECTORY objects** are used to locate the datafiles

```
SQL> CREATE DIRECTORY source_dir AS '+DATA/DB/DATAFILE';   -- Source
SQL> CREATE DIRECTORY target_dir AS '/u01/oradata/NEWDB';  -- Target
```

- **No support for OMF**
  - In the case of ASM an alias is created with the defined name

Use online datafile move to rename the transported datafiles to OMFs.

DBMS_FILE_TRANSFER 2/2

trivadis

- Support for push (PUT_FILE) and pull (GET_FILE) transfers

```
SQL> BEGIN
   DBMS_FILE_TRANSFER.PUT_FILE (
      source_directory_object => 'SOURCE_DIR', source_file_name => 'hr_data_001.dbf',
      destination_directory_object => 'TARGET_DIR', destination_file_name => 'hr_data_001.dbf',
      destination_database => 'TARGETDB_LINK.trivadistraining.com'
   );
END;
/

SQL> BEGIN
   DBMS_FILE_TRANSFER.GET_FILE (
      source_directory_object => 'SOURCE_DIR', source_file_name => 'hr_data_001.dbf',
      source_database => 'SOURCEDB_LINK.trivadistraining.com'
      destination_directory_object => 'TARGET_DIR', destination_file_name => 'hr_data_001.dbf'
   );
END;
/
```

Maximum supported file size is 2 TB. Monitor V$SESSION_LONGOPS to see the remaining time.

Foreign Tablespace/Datafile Restore

trivadis

- Starting with 12c, tablespaces and datafiles can be restored with backup sets of a foreign database
  - Multiple backupsets can be combined in one operation
- Endianness can be changed during the restore (FROM PLATFORM)
- Use TO NEW clause to generate new OMF or define your own location and names by using the FORMAT clause

```
RMAN> RUN {
   RESTORE FROM PLATFORM 'Linux x86 64-bit' FOREIGN TABLESPACE "USERS","HR_DATA"
   TO NEW | FORMAT '/u01/oradata/%d/%N_%f.dbf'
   FROM BACKUPSET '/backup/tts_inc0_users.bak', '/backup/tts_inc0_hr_data.bak';
}

RMAN> RUN {
   RESTORE FROM PLATFORM 'Linux x86 64-bit' ALL FOREIGN DATAFILES
   TO NEW | FORMAT '/u01/oradata/%d/%N_%f.dbf'
   FROM BACKUPSET '/backup/tts_inc0_users.bak', '/backup/tts_inc0_hr_data.bak';
}
```

All datafiles in the backup set(s)

FROM PLATFORM clause is mandatory when the backup is from databases below 12c – even when the platform is the same.

Foreign Datafile Recovery

trivadis

- After the restore of a foreign tablespace/datafile, the datafiles can be recovered with incremental backups
- Each foreign datafile has to be recovered
  - No support for wildcards
  - Combine all datafile locations in one command

```
RMAN> RUN {
   RECOVER FROM PLATFORM 'Linux x86 64-bit'
   FOREIGN DATAFILECOPY '/u01/oradata/%d/users_001.dbf',
                        '/u01/oradata/%d/hr_data_001.dbf'
   FROM BACKUPSET '/backup/tts_inc0_users.bak', '/backup/tts_inc0_hr_data.bak';
}
```

- Unfortunately, there is currently no way to track foreign datafiles from the database
  - In Oracle 21c, view V$FOREIGN_DATAFILE_COPY was added, but it is empty (Bug?)

Incremental Backups

trivadis

- **Incremental backups** are used to **minimize the downtime** of the datafile transport
- **Level 0** and subsequential **level 1 hot backups** are created on the source and restored on the target
- After setting the tablespaces read-only, a **final consistent level 1** is created and restored
  - This reduces the downtime dramatically
- Oracle provides a Perl script to automate this (Doc. ID 2471245.1)

```
$> perl xttdriver.pl --backup
$> perl xttdriver.pl --restore
```

- **Image copies** are used for the **level 0 backup**
  - A stage location is required to temporarily store the copies and backups

Use a shared filesystem (for example NFS) to share the copies and backups between source and target system.

Data Guard

trivadis

- If you want to have a **in-sync standby database** after importing the metadata, **restore the datafiles also on the standby site**
  - Imported datafiles are not automatically copied to the standby site
- DBMS_FILE_TRANSFER cannot connect to a standby database
  - As a workaround, use a dummy database on the target server
- As alternative, recreate the standby database or restore the new tablespaces
  - RECOVER STANDBY DATABASE command was added in 18c

```
-- Whole standby database
RMAN> RECOVER STANDBY DATABASE FROM SERVICE 'primary_tns_alias';

-- Tablespaces
RMAN> RESTORE TABLESPACE "USERS" FROM SERVICE 'primary_tns_alias';
```

Redo apply has to be stopped, before the recovery can be started.

trivadis

Metadata Import

Metadata Import

trivadis

- Provide the list of transported datafiles using TRANSPORT_DATAFILES parameter
  - Wildcards (*,?) for file names are supported starting with Oracle 12c Release 2
  - Remapping of Schemas is also possible durint the import

```
$> impdp … TRANSPORT_DATFILES = '/u01/oradata/DB/hr_data_001.dbf','…' \
           REMAP_SCHEMAS = HR:HR_NEW EXCLUDE = TRIGGER
```

- When the dump was created with the legacy Export Utility, use imp to import it

```
$> imp … DATFILES = '/u01/oradata/DB/hr_data_001.dbf','…' \
         TABLESPACES = HR_DATA,… TRANSPORT_TABLESPACE = Y \
         FROMUSER = HR TOUSER = HR_NEW
```

- Import all other non-transported objects using normal import mode
- Minimize the risk of ORA-600 by excluding triggers

Starting with 19c, tablespaces can be shared again between databases. Use parameter TRANSPORTABLE = KEEP_READ_ONLY to activate the pre-12.2 behavior.

Set Tablespaces read-write

trivadis

- Set the transported tablespaces in the target database to read-write

```
SQL> ALTER TABLESPACE HR_DATA READ WRITE;
```

trivadis

Post Tasks

Gather Statistics

trivadis

- If statistics were skipped during export or import, it is time to gather them
- Beside object statistics, update data dictionary and fixed object statistics as well

```
SQL> EXEC DBMS_STATS.SET_GLOBAL_PREFS('DEGREE', 8);
SQL> EXEC DBMS_STATS.SET_GLOBAL_PREFS('CONCURRENT','AUTOMATIC');

SQL> BEGIN
   DBMS_STATS.GATHER_DICTIONARY_STATS();
   DBMS_STATS.GATHER_FIXED_OBJECTS_STATS();
   DBMS_STATS.GATHER_DATABASE_STATS(
      options => 'GATHER',
      estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE,
      method_opt => 'FOR ALL COLUMNS SIZE SKEWONLY',
      granularity => 'ALL',
      cascade => TRUE,
      degree => 8,
      no_invalidate => DBMS_STATS.AUTO_INVALIDATE
   );
END;
/
```

Gather statistics for the important schemas using GATHER_SCHEMA_STATS first.

Restore Default Tablespaces

trivadis

- Restore the original default tablespaces of the transported user

```
SQL> ALTER USER "HR" DEFAULT TABLESPACE "HR_DATA";
```

- Genarate the required commands on the source database

```
SQL> SELECT 'ALTER USER "' || username || '" DEFAULT TABLESPACE "' ||
            default_tablespace || '";'
       FROM dba_users
      WHERE username IN ('HR', '…');
```

- Update the database default tablespace, if required

```
SQL> ALTER DATABASE DEFAULT TABLESPACE "USERS";
```

Correct OMF Names

trivadis

- Only required when OMFs are used, and the **file locations and names are not correct** because of the used transfer mode
- Starting with 12c **online datafile move** can be used to move/rename the datafiles online
  - Omit TO clause to create new OMF datafiles

```
-- Datafile number
SQL> ALTER DATABASE MOVE DATAFILE n;

-- Datafile path
SQL> ALTER DATABASE MOVE DATAFILE '+DATA/DB/hr_data_001.dbf';
```

Online datafile move is slower than an offline move using RMAN.

trivadis

Pitfalls

Tables with TSLTZ Columns

trivadis

- Columns of the type TIMESTAMP WITH LOCAL TIME ZONE (TSLTZ) can only be transported, when source and target use the same database time zone
  - These tables are automatically skipped in 12c and higher

```
W-1 Processing object type TRANSPORTABLE_EXPORT/TABLE
ORA-39360: Table "TTS_DEMO"."TSLTZ_TAB1" was skipped due to transportable
import and TSLTZ issues resulting from time zone mismatch.
```

- Change the database time zone to the value of the source database
  - This is only possible when not already tables with TSLTZ columns exist

```
SQL> ALTER DATABASE SET TIME_ZONE = '+05:00';
```

- If not possible, use normal Data Pump Export/Import to transport the data

Create the table before the recompilation of PL/SQL objects and views.

Global Temporary Tables

trivadis

- Global Temporary Tables (GTT) are not transported
  - These tables do not belong to a tablespace
- Data Pump export and import have to be used to transport them to the target database

```
$> expdp … INCLUDE=TABLE:"IN (SELECT table_name FROM dba_tables WHERE temporary
= 'Y')"
```

- Or create them manually either as preparation or post task

```
SQL> SELECT DBMS_METADATA.GET_DDL('TABLE','<Name>','<Schema>') FROM dual;
```

Create the table before the recompilation of PL/SQL objects and views.

Tables with XMLTYPE Columns

trivadis

- Transportation of tables with XMLTYPE columns is supported with 12.2.0.1 and higher
- In previous versions, following error is raised during the metadata import

```
ORA-39139: Data Pump does not support XMLType objects in version
"<SCHEMA_NAME>"."<TABLE_NAME>" will be skipped.
```

- Table exclude is required for the TTS operation

```
$> expdp … EXCLUDE=TABLE:"IN (SELECT table_name FROM dba_tab_columns WHERE
data_type = 'XMLTYPE')"
```

- Normal Data Pump export/import is required to transport them
- MOS note Is it supported to do a Transport Tablespace (TTS) Import with Data Pump on a tablespace with binary XML objects ? (Doc ID 1908140.1) describes this issue

Perform the exclude during the export to speed up the metadata export.

Further Information

Oracle Database 21c – Transporting Data
https://docs.oracle.com/en/database/oracle/oracle-database/21/admin/transporting-data.html#GUID-1901E9C3-8FCE-4D4E-AB65-34D703474E52

Master Note for Transportable Tablespaces (TTS) -- Common Questions and Issues (Doc ID 1166564.1)
https://support.oracle.com/epmos/faces/DocumentDisplay?id=1166564.1

My Oracle Support
https://support.oracle.com

Questions and answers..

Christian Gohmann
Principal Consultant

Tel. +49-211-58 6664 702
christian.gohmann@trivadis.com

@CGohmannDE

trivadis

For more information about this Session, please contact:

Christian Gohmann

christian.gohmann@trivadis.com

https://www.christian-gohmann.de

For details about joining RMOUG, please go to our Join Us Page:

rmoug.org/Join Us

or contact:

Tim Mishek, Membership Director

membershipdir@rmoug.org