



ONE PDB TO GO, PLEASE!

CHRISTIAN GOHMANN
PRINCIPAL CONSULTANT
TRIVADIS GERMANY GMBH

MICHIGAN ORACLE USERS SUMMIT 2021

MONDAY OCTOBER 25 - THURSDAY OCTOBER 28, 2021
VIRTUAL EVENT



3 HALLO, GRÜEZI, HI!



CHRISTIAN GOHMANN

- Principal Consultant, Instructor and Tool Owner of db*BACKUP
- Working with Oracle since 2006
- Focused on High Availability Solutions, Migration Projects, Backup & Recovery and Cloud Technologies
- Oracle ACE





1994

FOUNDATION

> 22000

TRAINING PARTICIPANTS
PER YEAR



21000



ORDERS
PER YEAR



> 750

CUSTOMERS PER
YEAR



123 Mio.

TURNOVER

> 300

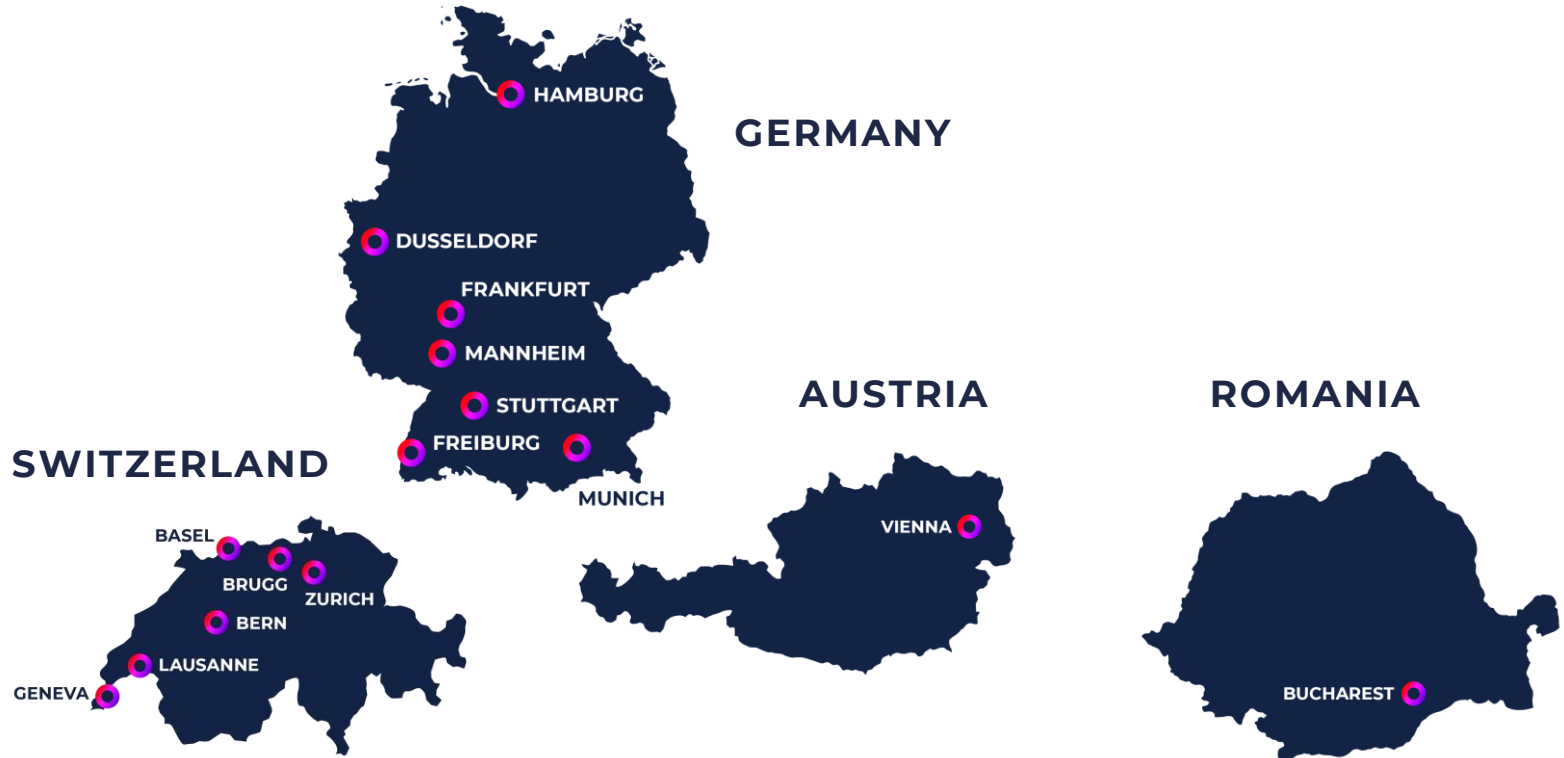
ACTIVE SLAs



trivadis

Part of Accenture

5 OUR WORKSPACES



6 AGENDA

- 1. Local / Remote Cloning**
- 2. Unplug / Plug-in PDB**
- 3. Refreshable PDB**
- 4. Snapshot Carousel**
- 5. RMAN Enhancements**

LOCAL / REMOTE CLONING

8 REQUIREMENTS & RESTRICTIONS 1/2

- User with **CREATE PLUGGABLE DATABASE privilege** is required
- Source PDB must be **opened read-only or read-write**
 - For <= 12.1.0.2 read-only was the only supported way
 - **Local Undo Mode** is required to use a read-write PDB (>= Oracle 12c Release 2)
- CDB must run in **ARCHIVELOG Mode**
 - Cloning with NOARCHIVELOG Mode is possible, but neither supported nor documented
- **Character set, endianness** and the **same installed options** are required
 - Starting with 12c Release 2 the target character set of the CDB can be a superset

9 REQUIREMENTS & RESTRICTIONS 2/2

- If the target CDB has a higher version, the PDB must be upgraded
 - Downgrading a PDB is not possible

```
$> $ORACLE_HOME/bin/dbupgrade -c PDB01
```


10 LOCAL CLONING 1/2

- Clones an **existing PDB within the same CDB**
 - After the cloning the new PDB is closed
 - A default service named by the PDB is automatically created (**don't use it**)
- Easiest way is to **use Oracle Managed Files (OMF)**
 - To use a different OMF location, use CREATE_FILE_DEST parameter
 - For non-OMF paths, use FILE_NAME_CONVERT parameter to adjust paths
- Example

```
SQL> CREATE PLUGGABLE DATABASE PDB2 FROM PDB1;  
SQL> CREATE PLUGGABLE DATABASE PDB2 FROM PDB1 CREATE_FILE_DEST='/u02/oradata';
```



NO DATA keyword can be used to create a **clone without data**.

11 LOCAL CLONING 2/2

- Starting with 18c **DBCA** can be used to **clone a local PDB** or to **plug-in a PDB**
 - Only available in **silent mode**

```
$> dbca -silent -createPluggableDatabase -sourceDB CDB1 \  
      -createPDBFrom PDB -pdbName PDB01_CLONE -sourcePDB PDB01
```

```
Prepare for db operation
```

```
13% complete
```

```
Creating Pluggable Database
```

```
15% complete
```

```
19% complete
```

```
23% complete
```

```
31% complete
```

```
53% complete
```

```
Completing Pluggable Database Creation
```

```
60% complete
```

```
Executing Post Configuration Actions
```

```
100% complete
```

```
Pluggable database "PDB01_CLONE" plugged successfully.
```

```
Look at the log file "/u00/app/oracle/cfgtoollogs/dbca/CDB1/PDB01_CLONE/CDB1.log" for further details.
```

12 SNAPSHOT COPY 1/2

- Instead of copying all datafiles belonging to the PDB, a storage snapshot is used
 - New PDB depends on the storage snapshot
 - Unplugging a PDB based on a snapshot is not possible
 - Dropping the PDB snapshot is not possible

```
SQL> CREATE PLUGGABLE DATABASE PDB1 ... SNAPSHOT COPY;
```

- Storage snapshots are supported by ACFS, ZFS Storage Appliance and Direct NFS Client storage
 - On Exadata ASM configured with sparse ASM Grid Disks is also supported

13 SNAPSHOT COPY 2/2

- A snapshot copy **PDB can be materialized** to remove the dependency to the used storage snapshot

```
SQL> ALTER PLUGGABLE DATABASE MATERIALIZE;
```

- Storage snapshots are supported by ACFS and ZFS Storage Appliance
- Local file systems, network file systems (NFS) or clustered file systems with enabled DirectNFS can be used, when they support sparse files
 - Initialization parameter CLONEDB must be set to TRUE

```
SQL> ALTER SYSTEM SET clonedb = TRUE SCOPE = SPFILE;
```



On local file systems, PDB must be read-only during snapshot clone (< 19c).

14 REMOTE CLONING 1/2

- Clones an existing **PDB of a remote CDB into the local CDB**
 - Cloning a non-CDB into a PDB is also possible when the source database is 12c or higher
- A **Database Link** is used to connect to the remote (non-)CDB
 - Remote user can be either a common or local user (in the target PDB)
 - Minimum privileges: CREATE SESSION, CREATE PLUGGABLE DATABASE
- Example:

```
SQL> CREATE PLUGGABLE DATABASE PDB1_CLONE FROM PDB1@SOURCE_PDB;  
SQL> ALTER PLUGGABLE DATABASE PDB1_CLONE OPEN;
```

- Add keyword AS PROXY to create a **Proxy PDB** (Oracle 12c Release 2)



After cloning a non-CDB run `$ORACLE_HOME/rdbms/admin/noncdb_to_pdb.sql` to upgrade the Data Dictionary of the new PDB (< 21c, Replay Upgrade).

15 REMOTE CLONING 2/2

- With **19c** a remote clone can be created with the **DBCA in silent mode**

```
$> dbca -silent -createPluggableDatabase -sourceDB CDB2 -pdbName PDB01_CLONE \  
-createFromRemotePDB -remotePDBName PDB01 -remoteDBConnString "CDB1.world" \  
-remoteDBSYSDBAUserName SYS -remoteDBSYSDBAUserPassword manager \  
-dbLinkUsername SYSTEM -dbLinkUserPassword manager
```

TNS Alias or
EZCONNECT

Prepare for db operation

50% complete

Create pluggable database using remote clone operation

100% complete

Pluggable database "PDB01_CLONE" plugged successfully.

Look at the log file "/u00/app/oracle/cfgtoollogs/dbca/CDB2/PDB01_CLONE/CDB21.log" for further details.



Do not use SYS as user for the database link.

PDB UNPLUG / PLUG-IN

17 PDB UNPLUG 1/2

- **Disassociates a PDB from its CDB**
 - An **unplugged PDB is still part of the CDB** and its backup
 - The **only operation** for an unplugged PDB is **DROP PLUGGABLE DATABASE**
- Easy way to **move one PDB to another CDB**
 - Or as **archive solution** instead of a final Data Pump dump
- An unplugged PDB can be **used as base for new PDBs**
- Two ways to unplug a PDB, depending on the used file extension
 - **XML Metadata File:** XML Manifest and Datafiles must be copied separately
 - **PDB Archive File:** Compressed archive with XML Manifest and Datafiles



PDB Archive Files were introduced with Oracle 12c Release 2. Usage requires additional time and CPU resources.

18 PDB UNPLUG 2/2

- Example

```
SQL> ALTER PLUGGABLE DATABASE PDB1 CLOSE IMMEDIATE INSTANCES = ALL;  
  
-- XML Metadata File  
SQL> ALTER PLUGGABLE DATABASE PDB1 UNPLUG INTO '/stage/pdb1_20180913.xml';  
  
-- PDB Archive File  
SQL> ALTER PLUGGABLE DATABASE PDB1 UNPLUG INTO '/stage/pdb1_20180913.pdb';
```



You can change the file extension from .pdb to .zip and extract all files (XML File, Datafiles) with an unzip tool.

19 PDB PLUG-IN

- Creates a **new PDB based on the unplugged PDB**
- Datafiles are copied (default) or moved to the correct location (e.g OMF location) or will stay at the current location if NOCOPY is used
- Check Plug-in compatibility with DBMS_PDB.CHECK_PLUG_COMPATIBILITY

```
-- XML Metadata File
SQL> CREATE PLUGGABLE DATABASE PDB1 USING '/stage/pdb1_20180913.xml' NOCOPY;

-- PDB Archive File
SQL> CREATE PLUGGABLE DATABASE PDB1 USING '/stage/pdb1_20180913.pdb' MOVE;
```

- Violations are visible through PDB_PLUG_IN_VIOLATIONS view
- Search for entries with status PENDING, purging was introduced with 18c



Use AS CLONE to plug in one unplugged PDB multiple times to avoid ORA-65122.

REFRESHABLE PDB

21 REFRESHABLE PDB

- Introduced with Oracle 12c Release 2 (**Engineered Systems and Cloud only**)
- **Read-only clone** of an existing PDB, which is **refreshed in a regular interval**
 - Refreshable PDB must be **closed during each Refresh**
 - Uses a **Database Link** to connect to source PDB
- **Archive Logs** and **Redo** information are used to synchronize the Refreshable PDB
- **Role conversion** (switchover) is possible with Oracle 18c, but no direct failover

```
SQL> ALTER PLUGGABLE DATABASE PDB01 REFRESH MODE MANUAL  
FROM REFRESH_PDB@TARGET_PDB SWITCHOVER;
```

- Last sync of Redo information from source PDB is required
- Conversion of a Refreshable PDB to a normal PDB is possible, but not vice-versa



Foreign Archive Logs are written to the subdirectory **foreign_archive_log** (OMF) within the Fast Recovery Area.

22 REFRESH MODES 1/2

- Provide RESFRESH MODE keyword to create a Refreshable PDB

```
SQL> CREATE PLUGGABLE DATABASE REFRESH_PDB FROM PDB1@SOURCE_PDB  
      REFRESH MODE MANUAL;
```

-- If required, you can open the Refreshable PDB read-only.

```
SQL> ALTER PLUGGABLE DATABASE REFRESH_PDB OPEN READ ONLY;
```

- Supported Refresh Modes:
 - NONE (default, deactivated)
 - MANUAL
 - EVERY x MINUTES | HOURS (1 Minute is the lowest possible interval)



Current configuration is visible in the columns REFRESH_MODE and REFRESH_INTERVAL of DBA_PDBS.

23 REFRESH MODES 2/2

- For the automatic Refresh, a **DBMS Scheduler Job** is created to initiate the Refresh

```
SQL> SELECT owner, job_name, repeat_interval
       FROM dba_scheduler_jobs
       WHERE job_name like '%REFRESH';
```

```
OWNER JOB_NAME REPEAT_INTERVAL
-----
SYS REFRESH_PDB_4204314029_REFRESH  FREQ = MINUTELY; INTERVAL = 1
```

PDB Name

PDB DBID



The PDB is not closed automatically.

24 ALTERNATE ARCHIVE LOG SOURCE

- If **Archive Logs are missing**, Refresh will fail with generic **ORA-65345 error**

ORA-65345: cannot refresh pluggable database

- Solutions:

1. Restore Archive Logs on the source site to their original location
2. Set parameter REMOTE_RECOVERY_FILE_DEST within Refreshable PDB to the location of the restored Archive Logs

```
SQL> ALTER SESSION SET CONTAINER = REFRESH_PDB;  
SQL> ALTER PLUGGABLE DATABASE OPEN READ ONLY;  
SQL> ALTER SYSTEM SET remote_recovery_file_dest = '/u01/arcs';  
SQL> ALTER PLUGGABLE DATABASE CLOSE IMMEDIATE;
```



If REMOTE_RECOVERY_FILE_DEST is set, only Archive Logs in the provided location are considered for Recovery.

25 REFRESHABLE PDB - FAILOVER

- When you try to do a „Failover“ by deactivating the Refresh Mode for the Refreshable PDB, it will fail

```
SQL> ALTER PLUGGABLE DATABASE REFRESH MODE NONE;  
ORA-17627: ORA-12514: TNS:listener does not currently know of service requested  
in connect descriptor  
ORA-17629: Cannot connect to the remote database server
```

- With the help of the REMOTE_RECOVERY_FILE_DEST parameter it is possible to do it

```
SQL> ALTER SYSTEM SET remote_recovery_file_dest = '/u01/arcs';  
SQL> ALTER PLUGGABLE DATABASE CLOSE IMMEDIATE;  
SQL> ALTER PLUGGABLE DATABASE REFRESH MODE NONE;  
SQL> ALTER PLUGGABLE DATABASE OPEN READ WRITE;
```

Empty directory



Because of unpublished Bug 24434583 in 12.2.0.1 reading Archive Logs from the specified location will fail – fixed with 18.1 (Doc ID 2408829.1).

SNAPSHOT CAROUSEL

27 SNAPSHOT CAROUSEL

- Introduced with Oracle 18c (**Engineered Systems and Cloud only**)
- **Fixed sized set** of manually or automatically created **PDB Snapshots**
 - A PDB Snapshot is a Point-in-Time copy of a PDB
 - Internally stored as **PDB Archive Files**
 - System-generated PDB Snapshots names are prefixed with SNAP_
- **Oldest PDB Snapshot is overwritten**, when the configured maximum is reached
 - Maximum number of kept Snapshots can be configured - but not higher than 8 (default)

```
SQL> ALTER PLUGGABLE DATABASE SET MAX_PDB_SNAPSHOTS = 4;
```



Check CDB_PROPERTIES or DATABASE_PROPERTIES to get the configured value of MAX_PDB_SNAPSHOTS.

28 PDB SNAPSHOT HANDLING 1/2

- Create a **new PDB Snapshot**, either with system-generated or user-defined name

```
SQL> ALTER PLUGGABLE DATABASE SNAPSHOT;  
SQL> ALTER PLUGGABLE DATABASE SNAPSHOT pdb1_snap_20180912;
```

- **Activate automatic creation** of new PDB Snapshots
 - Maximum interval is either 3000 minutes or 2000 hours
 - Setting Snapshot Mode to NONE deactivates the feature

```
SQL> ALTER PLUGGABLE DATABASE SNAPSHOT MODE EVERY 12 HOURS;  
SQL> ALTER PLUGGABLE DATABASE SNAPSHOT MODE NONE;
```



Information about PDB Snapshots are visible in DBA_PDB_SNAPSHOTS and DBA_PDBS (SNAPSHOT_MODE, SNAPSHOT_INTERVAL).

29 PDB SNAPSHOT HANDLING 2/2

- Create a **PDB based on a PDB Snapshot**

```
SQL> CREATE PLUGGABLE DATABASE PDB1_CLONE FROM PDB1  
      USING SNAPSHOT pdb1_snap_20180912 [SNAPSHOT COPY];
```

- **Drop all PDB Snapshots** by setting MAX_PDB_SNAPSHOTS to 0

```
SQL> ALTER PLUGGABLE DATABASE SET MAX_PDB_SNAPSHOTS = 0;
```

- Or **drop specific PDB Snapshots** manually

```
SQL> ALTER PLUGGABLE DATABASE DROP SNAPSHOT pdb1_snap_20180912;
```

30 SNAPSHOT CREATION – BEHIND THE SCENE 1/2

1. A **Local Snapshot Clone of the PDB** is created

```
SQL> CREATE PLUGGABLE DATABASE "SNAP_2984345588_986670161" FROM "PDB1"  
      CREATE_FILE_DEST = '/u00/app/oracle/oradata' SNAPSHOT COPY  
      KEystore IDENTIFIED BY EXTERNAL STORE;
```

2. If SNAPSHOT COPY clause is not supported, a **Local Normal Clone** is created

```
SQL> CREATE PLUGGABLE DATABASE "SNAP_2984345588_986670161" FROM "PDB1"  
      CREATE_FILE_DEST = '/u00/app/oracle/oradata'  
      KEystore IDENTIFIED BY EXTERNAL STORE;
```



If the used file system does not support a snapshot copy, all datafiles are copied physically.

31 SNAPSHOT CREATION – BEHIND THE SCENE 2/2

3. PDB clone is **unplugged as PDB Archive File**

```
SQL> ALTER PLUGGABLE DATABASE "SNAP_2984345588_986670161"  
      UNPLUG INTO '/u00/app/oracle/oradata/snap_2984345588_5022845.pdb';
```

4. PDB clone and its datafiles are **dropped**

```
SQL> DROP PLUGGABLE DATABASE "SNAP_2984345588_986670161" INCLUDING DATAFILES;
```

RMAN ENHANCEMENTS

33 DUPLICATE PLUGGABLE DATABASE 1/2

- From 18c onwards, it is possible to **duplicate one PDB using RMAN**
 - Only from **active database duplication** is supported
- Skipping Tablespaces of the PDB is not supported
- **REMOTE_RECOVERY_FILE_DEST** parameter must be set on target CDB
 - Location is used to restore foreign archive logs

```
SQL> ALTER SYSTEM SET remote_recovery_file_dest = '/u01/remote_recovery_area';
```

- After the duplication, the **PDB is opened read-write**



Active Database Duplication requires a connection to target and auxiliary using a service name and the same password.

34 DUPLICATE PLUGGABLE DATABASE 2/2

- Example

```
RMAN> CONNECT TARGET sys/oracle@SOURCE_DB  
RMAN> CONNECT AUXILIARY sys/oracle@TARGET_DB  
RMAN> DUPLICATE PLUGGABLE DATABASE L18CEEC1_PDB1 AS NEW_PDB TO TARGET_DB  
FROM ACTIVE DATABASE;
```

Name of
new PDB

Name of
source PDB

Name of
target CDB



Use AS clause to define a new name for the duplicated PDB.

FURTHER INFORMATION

36 LINKS

- **Oracle Administrator Guide 12c Release 2 (12.2)**
<https://docs.oracle.com/en/database/oracle/oracle-database/12.2/admin/managing-a-multitenant-environment.html#GUID-93F1E584-D309-4301-82E0-AD0E60D4977C>
- **Oracle Multitenant Administrator Guide 18c**
<https://docs.oracle.com/en/database/oracle/oracle-database/18/multi/index.html>
- **Oracle Multitenant Administrator Guide 19c**
<https://docs.oracle.com/en/database/oracle/oracle-database/19/multi/index.html>
- **Oracle Multitenant Administrator Guide 21c**
<https://docs.oracle.com/en/database/oracle/oracle-database/21/multi/index.html>
- **My Oracle Support**
<https://support.oracle.com>

37 QUESTIONS AND ANSWERS



CHRISTIAN GOHMANN

Principal Consultant

 +49 211 58 6664 702

 christian.gohmann@trivadis.com

 <https://www.linkedin.com/in/christian-gohmann/>

 @CGohmannDE

 <https://www.christian-gohmann.de>



trivadis Part of **Accenture**

SAVE THE DATE

- ASCEND CONFERENCE 2022

June 12-15, 2022

the Aria in Las Vegas, NV

<https://ascendusersconference.com>



- MOUS 2022

October 26, 2022

Schoolcraft College - VisTaTech Center,
18600 Haggerty Rd, Livonia, MI

<https://www.mous.us>



THANK YOU

www.mous.us



SURVEYS

- Session Surveys

Please complete the session survey for this session using the Zoom session survey link.

The survey link will be provided via email once the webinar is closed.



THANK YOU

WWW.MOUS.US

