

# RMAN – Advanced Scenarios

Marcin Przepiorowski | Senior Technical Principal | November, 2018

# About me

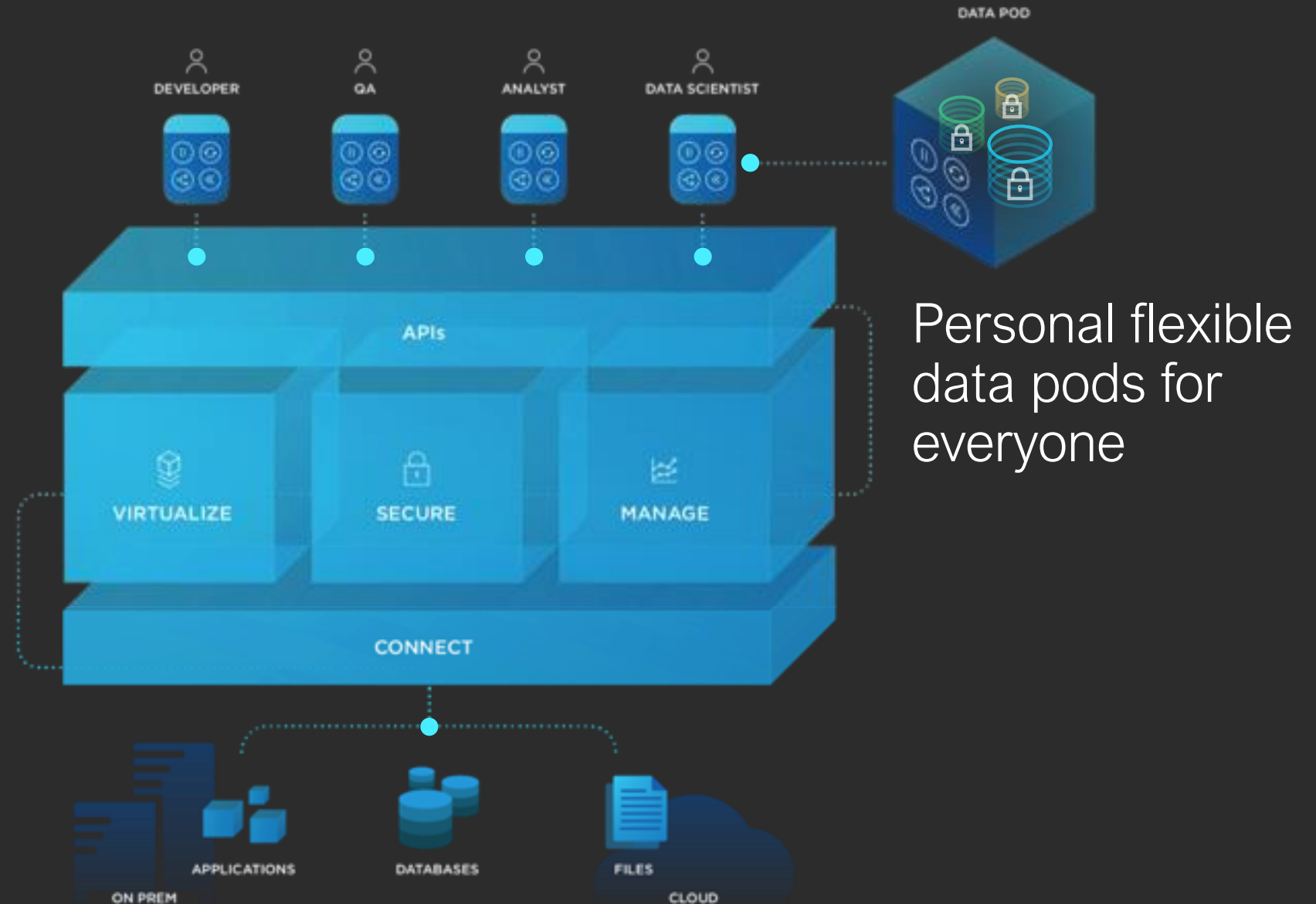
Oracle consultant/DBA since 2000

co-developer of OraSASH – free ASH/AWR like repository

Automation / DevOps freak



# Delphix Dynamic Data Platform

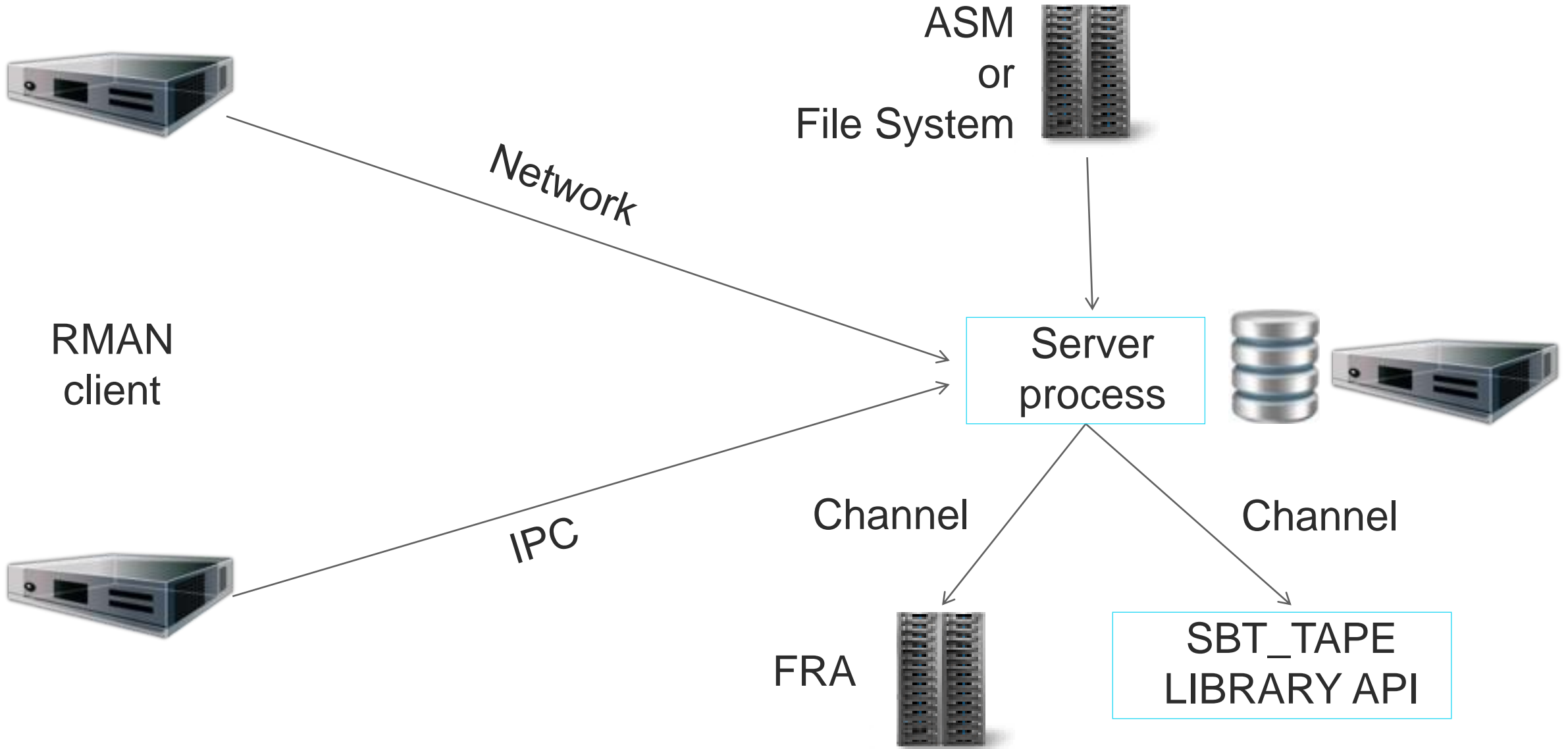


Point in time recovery

Duplication

Metadata operation

Cloud integration





# Point in time recovery

# Point in time recovery

Easy for

- a whole database (non-cdb)
- CDB plus all PDB databases
- PDB with local undo - hmmm



Common table spaces



Data table spaces





# Point in time recovery

For other cases:

- Tables space
- Table
- Single PDB (12.1, 12.2 without local undo)

There are common objects like system and undo table space and redo/archive log.

Additional resources ( disk space, CPU and memory ) are required



Common table spaces



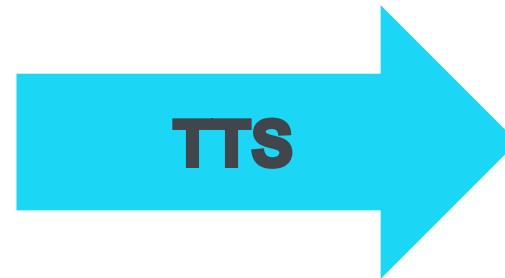
Data table spaces



Common table spaces



Data table spaces



# Point in time recovery

```
set until logseq X thread Y
```

```
set until time "to_date('2016-11-11 13:13:13','yyyy-mm-dd hh24:mi:ss')"
```

```
set until sequence 12345678
```

```
restore / recover ... database;
```

```
recover tablespace users until ..... auxiliary destination '/tmp/';
```

```
recover table "TEST"."TEST" until ..... auxiliary destination '/tmp';
```

# Backup, PITR and RAC

- Be careful with until logseq – RAC has at least 2 independent threads of redo logs
- Make sure you always archive current logs after taking backup – all threads are needed for recovery
- Backup of archive logs from standby's can be more tricky – no archive on demand is possible and archive logs from less utilized nodes of primary RAC can be shipped out of required order



# Demo 1

# Restore non-MT / CDB with single PDB PITR

```
run {  
  set until time "to_date('2018-11-06 17:42:00', 'yyyy-mm-dd hh24:mi:ss')";  
  restore database ;  
  recover database ;  
}
```

# Restore MT PDB no local undo

```
run {  
  set until time "to_date('2018-11-06 18:27:00', 'yyyy-mm-dd hh24:mi:ss')";  
  set auxiliary instance parameter file to '/tmp';  
  restore pluggable database pdbns ;  
  recover pluggable database pdbns auxiliary destination '/tmp';  
}
```

# Restore MT PDB local undo

```
run {  
  alter pluggable database pdbclone close;  
  set until time "to_date('2018-11-08 16:38:30', 'yyyy-mm-dd hh24:mi:ss')";  
  restore pluggable database pdbclone;  
  recover pluggable database pdbclone;  
  alter pluggable database pdbclone open resetlogs;  
}
```



# Multiple restore of PDB with local undo

First recovery with resetlogs works fine.

Next one finish with:

```
ORA-39889: Specified System Change Number (SCN) or timestamp is in the middle of a previous PDB RESETLOGS operation.
```

Incarnation switch for PDB is not yet implemented

As workaround use:

How to Restore - Dropped Pluggable database (PDB) in Multitenant (Doc ID 2034953.1)

(be aware it was written for 12.1 without local undo – you need amend it to 18c with local undo)

Make sure you have backup of entire database (CDB / seed / PDB / archive logs / control file )



# Duplication

```
$ rman target /
```

```
Recovery Manager: Release 2035s
```

```
connected to target database: TEST121 (DBID=3899829438)
```

```
connected to auxiliary database: CLONE (not mounted)
```

```
RMAN> duplicate database to "CLONE"
```

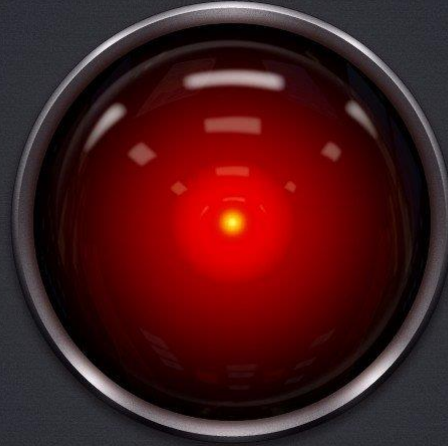
```
Starting Duplicate Db at 28-AUG-35
```

```
RMAN-00571: =====
```

```
RMAN-00569: ERROR MESSAGE STACK FOLLOWS
```

```
RMAN-00571: =====
```

```
RMAN-02001: DBA internvention detected
```



I'm sorry Dave I'm afraid I can't do that

# Duplicate database from backup

```
rman target sys/xxxxx@test18 auxiliary sys/xxxxxx@clonedb

duplicate database test18 to clonedb spfile
parameter_value_convert='test18','cloned'

set
db_file_name_convert='/u01/app/oracle/oradata/TEST18/','/u01/app/oracle/oradata/clonedb/'
set db_recovery_file_dest='/u01/app/oracle/oradata/clonedb/FRA'
set control_files='/u01/app/oracle/oradata/clonedb/control01.ctl',
'/u01/app/oracle/oradata/clonedb/control02.ctl'

backup location '/u01/app/oracle/backup/';
```

# Duplicate database from active database

```
rman target sys/xxxxx@test18 auxiliary sys/xxxxxx@clonedb
```

```
duplicate database to clonedb from active database password file
```

```
spfile parameter_value_convert='test18','clonedb'
```

```
set
```

```
db_file_name_convert='/u01/app/oracle/oradata/test18/', '/u01/app/oracle/oradata/clone/'
```

```
set db_recovery_file_dest='/u01/app/oracle/oradata/clone/fra'
```

```
set control_files='/u01/app/oracle/oradata/clone/control01.ctl';
```

```
duplicate database to clonedb pluggable database pdb18 from active database
```

```
password file spfile parameter_value_convert='test18','clonedb'
```

```
set
```

```
db_file_name_convert='/u01/app/oracle/oradata/test18/', '/u01/app/oracle/oradata/clone/'
```

```
set db_recovery_file_dest='/u01/app/oracle/oradata/clone/fra'
```

```
set control_files='/u01/app/oracle/oradata/clone/control01.ctl';
```

# Use duplicate to build standby

```
rman target sys/xxxxx@test18 auxiliary sys/xxxxxx@clonedb
```

```
duplicate target database for standby from active database  
spfile
```

```
set db_unique_name='clonedb'
```

```
set audit_file_dest='/u01/app/oracle/admin/clonedb/adump'
```

```
set db_recovery_file_dest='/u01/app/oracle/oradata/clone/fra'
```

```
nofilenamecheck;
```

# Use recover to fix standby gap

```
rman target sys/xxxxx@test18 auxiliary sys/xxxxxxx@clonedb
```

```
restore standby controlfile from service test18;
```

```
recover database from service test18 using backupset noredo;
```



# Demo 2





# Cloud integration

# AWS S3

## Oracle Secure Backup Cloud Module for Amazon S3

```
$ORACLE_HOME/jdk/bin/java -jar osbws_install.jar \  
-AWSID XXXXXXXXXXXX \  
-AWSKey xxxxxxxxxxxx \  
-walletDir $ORACLE_HOME/dbs/osbws_wallet \  
-libDir $ORACLE_HOME/lib \  
-libPlatform linux64 \  
-awsEndPoint: us-east-1
```

# AWS S3

Oracle Secure Backup Web Service Install Tool, build 12.2.0.1.0DBBKPCSBP\_2018-06-12

AWS credentials are valid.

Oracle Secure Backup Web Service wallet created in directory  
/u01/app/oracle/product/18c/db1/dbs/osbws\_wallet.

Oracle Secure Backup Web Service initialization file  
/u01/app/oracle/product/18c/db1/dbs/osbwscdb1.ora created.

Downloading Oracle Secure Backup Web Service Software Library from file osbws\_linux64.zip.

Download complete.

# AWS S3

```

RMAN> run {
2>  allocate channel ch1 type 'sbt_tape'
   parms='SBT_LIBRARY=/u01/app/oracle/product/18c/db1/lib/libosbws.so,SBT_PARMS=(OSB_WS_PFILE
   =/u01/app/oracle/product/18c/db1/dbs/osbwscdb1.ora)';
3>  backup current controlfile ;
4> }

channel ch1: SID=264 device type=SBT_TAPE
channel ch1: Oracle Secure Backup Web Services Library VER=12.2.0.2

...

piece handle=07ti4cus_1_1 tag=TAG20181113T143404 comment=API Version 2.0,MMS Version
12.2.0.2

...

Starting Control File and SPFILE Autobackup at 13-NOV-18
piece handle=c-4048071595-20181113-03 comment=API Version 2.0,MMS Version 12.2.0.2

```

# AWS S3

```
(aws)pioro-mbp:aws mprzepiorowski$ aws s3 ls marcin-s3-backup-oracle/sbt_catalog/  
PRE 07ti4cus_1_1/  
PRE c-4048071595-20181113-03/
```

# Metadata maintenance

# Metadata maintenance

- CATALOG – add a new one or change status of existing backupset
- CROSSCHECK – cross-check RMAN metadata with media ( files or backup piece on SBT media )
- DELETE – delete a metadata information and backup sets on media

# UNCATALOG

```
RMAN> change backupset 14 uncatalog;
```

```
List of Backup Pieces
```

BP Key	BS Key	Pc#	Cp#	Status	Device	Type	Piece Name
14	14	1	1	AVAILABLE	DISK		/u01/o1_mf_s_991419335_fy1267x3_.bkp

```
Do you really want to delete the above objects (enter YES or NO)? yes
```

```
uncataloged backup piece
```

```
backup piece handle=/u01/o1_mf_s_991419335_fy1267x3_.bkp RECID=14 STAMP=991419335
```

```
Uncataloged 1 objects
```

```
[oracle@oracle18 oracle]$ ls -l /u01/2018_11_05/o1_mf_s_991419335_fy1267x3_.bkp  
-rw-r-----. 1 oracle oinstall 18841600 Nov 5 18:15 /u01/o1_mf_s_991419335_fy1267x3_.bkp
```



# CATALOG

```
RMAN> catalog backuppiece '/u01/o1_mf_s_991419335_fy1267x3_.bkp';
cataloged backup piece
backup piece handle=/u01/o1_mf_s_991419335_fy1267x3_.bkp RECID=15 STAMP=991424158
```

```
RMAN> list backupset;
```

BS Key	Type	LV	Size	Device	Type	Elapsed Time	Completion Time
15	Full		17.95M	DISK		00:00:00	05-NOV-18

BP Key: 15 Status: AVAILABLE Compressed: NO Tag: TAG20181105T181535  
Piece Name: /u01/o1\_mf\_s\_991419335\_fy1267x3\_.bkp  
SPFILE Included: Modification time: 05-NOV-18  
SPFILE db\_unique\_name: TEST18  
Control File Included: Ckp SCN: 1581669 Ckp time: 05-NOV-18

# DELETE

```
RMAN> delete backupset 15;
```

```
using channel ORA_DISK_1
```

```
List of Backup Pieces
```

```
BP Key   BS Key   Pc# Cp# Status       Device Type Piece Name
```

```
-----
```

```
15      15      1    1  AVAILABLE    DISK          /u01/o1_mf_s_991419335_fy1267x3_.bkp
```

```
Do you really want to delete the above objects (enter YES or NO)? yes
```

```
deleted backup piece
```

```
backup piece handle=/u01/o1_mf_s_991419335_fy1267x3_.bkp RECID=15 STAMP=991424158
```

```
Deleted 1 objects
```

```
[oracle@oracle18 oracle]$ ls -l /u01/o1_mf_s_991419335_fy1267x3_.bkp
```

```
ls: cannot access /u01/o1_mf_s_991419335_fy1267x3_.bkp: No such file or directory
```

# CATALOG using SBT\_TAPE media

```
RMAN> allocate channel for maintenance type 'sbt_tape'  
parms='SBT_LIBRARY=/u01/app/oracle/product/18c/db1/lib/libosbws.so,SBT_PARAMS=(OSB_WS_PFILE  
=/u01/app/oracle/product/18c/db1/dbs/osbwscdb1.ora)';
```

```
RMAN> catalog device type 'sbt_tape' backuppiece '0cti6iqu_1_1';
```

```
released channel: ORA_SBT_TAPE_1
```

```
allocated channel: ORA_SBT_TAPE_1
```

```
channel ORA_SBT_TAPE_1: SID=272 device type=SBT_TAPE
```

```
channel ORA_SBT_TAPE_1: Oracle Secure Backup Web Services Library VER=12.2.0.2
```

```
cataloged backup piece
```

```
backup piece handle=0cti6iqu_1_1 RECID=16 STAMP=992169086
```

# THANK YOU

Scripts location: <https://github.com/pioro?tab=repositories>

Marcin Przepiorowski  
@pioro  
[marcin@delphix.com](mailto:marcin@delphix.com)