

【RMAN】TSPITR--RMAN 表空间基于时间点的自动恢复

1.1 BLOG 文档结构图

└─ 【RMAN】TSPITR--RMAN 表空间基于时间点的自动恢复
└─ 1.1 BLOG 文档结构图
└─ 1.2 前言部分
└─ 1.2.1 导读
└─ 1.2.2 实验环境介绍
└─ 1.2.3 本文简介
└─ 1.3 TSPITR的相关知识点归纳
└─ 1.3.1 TSPITR 简介
└─ 1.3.2 何时使用 TSPITR
└─ 1.4 实验部分
└─ 1.4.1 源库做备份操作
└─ 1.4.2 建立测试表并做 truncate 误操作
└─ 1.4.3 采用 logminer 找回误删除的时间点
└─ 1.4.4 执行 TSPITR之前的检查
└─ 1.4.4.1 检查是否自包含
└─ 1.4.4.2 检查哪些对象执行 TSPITR后将被删除
└─ 1.4.5 执行 TSPITR
└─ 1.4.6 online 表空间并且导入丢失的对象
└─ 1.5 与 TSPITR 有关的 OCP 试题部分
└─ 1.6 总结
└─ 1.7 RMAN 系列参考文章
└─ 1.8 about me

1.2 前言部分

1.2.1 导读

各位技术爱好者，看完本文后，你可以掌握如下的技能，也可以学到一些其它你所不知道的知识，~O(∩_∩)O~：

① TSPITR 表空间基于时间点的自动恢复

② logminer 的简单应用

1. 2. 2 实验环境介绍

目标库：11.2.0.3 RHEL6.5

ORACLE_SID: ora11g

```
[oracle@rhel6_lhr ~]$ sqlplus / as sysdba

SQL*Plus: Release 11.2.0.3.0 Production on 星期日 5月 24 15:19:35 2015

Copyright (c) 1982, 2011, Oracle. All rights reserved.

连接到:
Oracle Database 11g Enterprise Edition Release 11.2.0.3.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options

15:19:35 SQL> select * from v$version;

BANNER
-----
Oracle Database 11g Enterprise Edition Release 11.2.0.3.0 - 64bit Production
PL/SQL Release 11.2.0.3.0 - Production
CORE 11.2.0.3.0 Production
TNS for Linux: Version 11.2.0.3.0 - Production
NLSRTL Version 11.2.0.3.0 - Production

已用时间: 00: 00: 00.01
15:19:37 SQL> archive log list;
数据库日志模式 存档模式
自动存档 启用
存档终点 USE_DB_RECOVERY_FILE_DEST
最早的联机日志序列 36
下一个存档日志序列 38
当前日志序列 38
15:19:53 SQL>
15:20:37 SQL> show parameter name

NAME                                TYPE        VALUE
-----
db_file_name_convert               string
db_name                            string      ora11g
db_unique_name                     string      ora11g
global_names                       boolean     FALSE
instance_name                      string      ora11g
lock_name_space                    string
log_file_name_convert              string
processor_group_name               string
service_names                      string      ora11g
15:20:41 SQL>
```

1.2.3 本文简介

OCF 考完已经接近快 1 年了，本来 TSPITR 是属于 OCF 的内容，应该早把这个实验做了，可是由于各种原因一直没有做这个实验，上周 CUUG 的公开课上讲的就是 TSPITR 的相关内容，好吧，借着周末就把这个实验做了吧，blog 后也附上了一些与 OCF 有关的试题，用于网友自测。

关于 TSPITR 的视频参考：：<http://yunpan.cn/QCwUAI9bn7g7w> 提取码：af2d

1.3 TSPITR 的相关知识点归纳

1.3.1 TSPITR 简介

TSPITR(表空间时间点恢复)用于将一个或多个表空间恢复到过去某个时间点的状态，而其他表空间仍然保持现有状态。

TSPITR 相关的概念和术语：

- (1) **TSPITR (Tablespace Point-In-Time Recover)**。TSPITR 是表空间时间点恢复的英文缩写格式，它表示将一个或多个表空间恢复到过去时间点的状态，而其他表空间仍然保持现有状态。
- (2) TSPITR 实现方法。当实现表空间时间点恢复时，既可以使用用户管理的表空间时间点恢复方法，也可以使用 RMAN 管理的表空间时间点恢复。
- (3) **DBPITR (Database Point-In-Time Recovery)**。DBPITR 是数据库时间点恢复的英文缩写格式，它表示将数据库的所有表空间恢复到过去时间点的状态。注意，DBPITR 只适用于 ARCHIVELOG 模式。
- (4) 主数据库(Primary Database)。主数据库是指用于存放应用系统数据的 Oracle 数据库，也被称为产品数据库或目标数据库。当执行 TSPITR 时，主数据库是指包含被恢复表空间的数据库。
- (5) 恢复集(Recovery Set)。恢复集是指在主数据库上需要执行 TSPITR 的表空间集合。注意，当在恢复集的表空间上执行 TSPITR 时，要求这些表空间必须是自包含的。
- (6) 辅助数据库(Auxiliary Database)。辅助数据库是主数据库的一个副本数据库。当执行 TSPITR 时，辅助数据库用于将恢复集表空间恢复到过去时间点。注意，辅助数据库的所有物理文件都是从主数据库备份中取得，并且辅助数据库必须包含 SYSTEM 表空间、UNDO 表空间以及恢复集表空间的备份文件。
- (7) 辅助集(Auxiliary Set)。辅助集是指辅助数据库所需要的、除了恢复集表空间文件之外的其他文件集合。当执行 TSPITR 时，辅助数据库除了需要恢复集表空间的备份文件之外，还需要备份控制文件、SYSTEM 表空间的备份文件、UNDO 表空间的备份文件。

1.3.2 何时使用 TSPITR

- TSPITR 可在下列情况下使用：
 - 恢复因错误的 TRUNCATE TABLE 语句而丢失的数据
 - 从表逻辑损坏中恢复
 - 撤消只影响部分数据库的批作业或 DML 语句的结果
 - 将逻辑方案恢复到与物理数据库其余部分不同的时间点
- TSPITR 使用可移动表空间和数据泵，提供了以下新功能和特性：
 - TSPITR 可用于恢复已删除的表空间
 - 可反复执行 TSPITR 恢复到表空间联机之前的多个时间点，而无需使用恢复目录

在 Oracle Database 11g 发行版2 之前，TSPITR 使用**导出和导入**进行处理。TSPITR 现在使用**可移动表空间和数据泵**。由于底层技术的这一变化，现在可使用 TSPITR 来恢复已删除的表空间。此外，可反复执行 TSPITR 恢复到不同的时间点，而无需使用恢复目录。

1.4 实验部分

1.4.1 源库做备份操作

```
[oracle@rhel6_lhr ~]$ ORACLE_SID=ora11g  
[oracle@rhel6_lhr ~]$ sqlplus / as sysdba
```

```
SQL*Plus: Release 11.2.0.3.0 Production on 星期日 5月 24 19:27:25 2015
```

```
Copyright (c) 1982, 2011, Oracle. All rights reserved.
```

```
已连接到空闲例程。
```

```
19:27:25 SQL> startup  
ORACLE 例程已经启动。
```

Total System Global Area 409194496 bytes
Fixed Size 2228864 bytes
Variable Size 327159168 bytes
Database Buffers 71303168 bytes
Redo Buffers 8503296 bytes
数据库装载完毕。
数据库已经打开。
19:27:42 SQL> exit
从 Oracle Database 11g Enterprise Edition Release 11.2.0.3.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options 断开

[oracle@rhel6_lhr ~]\$ ORACLE_SID=orallg

[oracle@rhel6_lhr ~]\$ rman target /

恢复管理器: Release 11.2.0.3.0 - Production on 星期日 5月 24 19:36:47 2015

Copyright (c) 1982, 2011, Oracle and/or its affiliates. All rights reserved.

已连接到目标数据库: ORA11G (DBID=4270446895)

RMAN> backup database;

启动 backup 于 2015-05-24 19:36:53
使用目标数据库控制文件替代恢复目录
分配的通道: ORA_DISK_1
通道 ORA_DISK_1: SID=142 设备类型=DISK
通道 ORA_DISK_1: 正在启动全部数据文件备份集
通道 ORA_DISK_1: 正在指定备份集内的数据文件
输入数据文件: 文件号=00001 名称=/u01/app/oracle/oradata/orallg/system01.dbf
输入数据文件: 文件号=00002 名称=/u01/app/oracle/oradata/orallg/sysaux01.dbf
输入数据文件: 文件号=00005 名称=/u01/app/oracle/oradata/orallg/example01.dbf
输入数据文件: 文件号=00003 名称=/u01/app/oracle/oradata/orallg/undotbs01.dbf
输入数据文件: 文件号=00004 名称=/u01/app/oracle/oradata/orallg/users01.dbf
输入数据文件: 文件号=00007 名称=/u01/app/oracle/oradata/orallg/DWII_CNY_BK_F_01.dbf
输入数据文件: 文件号=00008 名称=/u01/app/oracle/oradata/orallg/DWII_DPA_F_01.dbf
输入数据文件: 文件号=00009 名称=/u01/app/oracle/oradata/orallg/DWII_DPA_I_01.dbf
输入数据文件: 文件号=00010 名称=/u01/app/oracle/oradata/orallg/DWII_DPA_S_01.dbf
输入数据文件: 文件号=00011 名称=/u01/app/oracle/oradata/orallg/DWII_SOR_F_01.dbf
输入数据文件: 文件号=00012 名称=/u01/app/oracle/oradata/orallg/DWII_SOR_I_01.dbf
输入数据文件: 文件号=00013 名称=/u01/app/oracle/oradata/orallg/DW_USER.dbf
输入数据文件: 文件号=00014 名称=/u01/app/oracle/oradata/orallg/SQCHECK.dbf
输入数据文件: 文件号=00015 名称=/u01/app/oracle/oradata/orallg/SD_CNY_D_01.dbf
输入数据文件: 文件号=00016 名称=/u01/app/oracle/oradata/orallg/SD_CNY_F_01.dbf
输入数据文件: 文件号=00017 名称=/u01/app/oracle/oradata/orallg/SD_DPA_D_01.dbf
输入数据文件: 文件号=00018 名称=/u01/app/oracle/oradata/orallg/SD_DPA_F_01.dbf
输入数据文件: 文件号=00019 名称=/u01/app/oracle/oradata/orallg/SD_SORT_T_01.dbf
输入数据文件: 文件号=00020 名称=/u01/app/oracle/oradata/orallg/DWII_FXDM_F_01.dbf
输入数据文件: 文件号=00021 名称=/u01/app/oracle/oradata/orallg/SD_SOR_T_01.dbf
输入数据文件: 文件号=00006 名称=/u01/app/oracle/oradata/orallg/aa.dbf
通道 ORA_DISK_1: 正在启动段 1 于 2015-05-24 19:36:54
通道 ORA_DISK_1: 已完成段 1 于 2015-05-24 19:37:19
段句柄=/u01/app/oracle/flash_recovery_area/ORA11G/backupset/2015_05_24/o1_mf_nnndf_TAG20150524T193654_bp3fypc1_.bkp 标记=TAG20150524T193654 注释=NONE
通道 ORA_DISK_1: 备份集已完成, 经过时间:00:00:25
完成 backup 于 2015-05-24 19:37:19

启动 Control File and SPFILE Autobackup 于 2015-05-24 19:37:19
段 handle=/u01/app/oracle/flash_recovery_area/ORA11G/autobackup/2015_05_24/o1_mf_s_880573039_bp3fzhgy_.bkp comment=NONE
完成 Control File and SPFILE Autobackup 于 2015-05-24 19:37:22

RMAN> list backupset;

备份集列表
=====

BS 关键字 类型 LV 大小 设备类型 经过时间 完成时间

49 Full 1.18G DISK 00:00:20 2015-05-24 19:37:14
BP 关键字: 49 状态: AVAILABLE 已压缩: NO 标记: TAG20150524T193654

段名:/u01/app/oracle/flash_recovery_area/ORA11G/backupset/2015_05_24/ol_mf_nnndf_TAG20150524T193654_bp3fypc1_.bkp
备份集 49 中的数据文件列表

文件	LV	类型	Ckp SCN	Ckp 时间	名称
1		Full	1596722	2015-05-24 19:36:54	/u01/app/oracle/oradata/orallg/system01.dbf
2		Full	1596722	2015-05-24 19:36:54	/u01/app/oracle/oradata/orallg/sysaux01.dbf
3		Full	1596722	2015-05-24 19:36:54	/u01/app/oracle/oradata/orallg/undotbs01.dbf
4		Full	1596722	2015-05-24 19:36:54	/u01/app/oracle/oradata/orallg/users01.dbf
5		Full	1596722	2015-05-24 19:36:54	/u01/app/oracle/oradata/orallg/example01.dbf
6		Full	1596722	2015-05-24 19:36:54	/u01/app/oracle/oradata/orallg/aa.dbf
7		Full	1596722	2015-05-24 19:36:54	/u01/app/oracle/oradata/orallg/DWII_CNY_BK_F_01.dbf
8		Full	1596722	2015-05-24 19:36:54	/u01/app/oracle/oradata/orallg/DWII_DPA_F_01.dbf
9		Full	1596722	2015-05-24 19:36:54	/u01/app/oracle/oradata/orallg/DWII_DPA_I_01.dbf
10		Full	1596722	2015-05-24 19:36:54	/u01/app/oracle/oradata/orallg/DWII_DPA_S_01.dbf
11		Full	1596722	2015-05-24 19:36:54	/u01/app/oracle/oradata/orallg/DWII_SOR_F_01.dbf
12		Full	1596722	2015-05-24 19:36:54	/u01/app/oracle/oradata/orallg/DWII_SOR_I_01.dbf
13		Full	1596722	2015-05-24 19:36:54	/u01/app/oracle/oradata/orallg/DW_USER.dbf
14		Full	1596722	2015-05-24 19:36:54	/u01/app/oracle/oradata/orallg/SQCHECK.dbf
15		Full	1596722	2015-05-24 19:36:54	/u01/app/oracle/oradata/orallg/SD_CNY_D_01.dbf
16		Full	1596722	2015-05-24 19:36:54	/u01/app/oracle/oradata/orallg/SD_CNY_F_01.dbf
17		Full	1596722	2015-05-24 19:36:54	/u01/app/oracle/oradata/orallg/SD_DPA_D_01.dbf
18		Full	1596722	2015-05-24 19:36:54	/u01/app/oracle/oradata/orallg/SD_DPA_F_01.dbf
19		Full	1596722	2015-05-24 19:36:54	/u01/app/oracle/oradata/orallg/SD_SORT_T_01.dbf
20		Full	1596722	2015-05-24 19:36:54	/u01/app/oracle/oradata/orallg/DWII_FXDM_F_01.dbf
21		Full	1596722	2015-05-24 19:36:54	/u01/app/oracle/oradata/orallg/SD_SOR_T_01.dbf

BS 关键字 类型 LV 大小 设备类型 经过时间 完成时间

50 Full 9.67M DISK 00:00:01 2015-05-24 19:37:20
BP 关键字: 50 状态: AVAILABLE 已压缩: NO 标记: TAG20150524T193719

段名:/u01/app/oracle/flash_recovery_area/ORA11G/autobackup/2015_05_24/ol_mf_s_880573039_bp3fzhgy_.bkp
包含的 SPFILE: 修改时间: 2015-05-24 19:27:36
SPFILE db_unique_name: ORA11G
包括的控制文件: Ckp SCN: 1596737 Ckp 时间: 2015-05-24 19:37:19

RMAN> report schema;

db_unique_name 为 ORA11G 的数据库的数据库方案报表

永久数据文件列表

文件大小 (MB)	表空间	回退段数据文件名称
1 770	SYSTEM	*** /u01/app/oracle/oradata/orallg/system01.dbf
2 580	SYS_AUX	*** /u01/app/oracle/oradata/orallg/sysaux01.dbf
3 285	UNDOTBS1	*** /u01/app/oracle/oradata/orallg/undotbs01.dbf
4 96	USERS	*** /u01/app/oracle/oradata/orallg/users01.dbf
5 345	EXAMPLE	*** /u01/app/oracle/oradata/orallg/example01.dbf
6 5	AA	*** /u01/app/oracle/oradata/orallg/aa.dbf
7 10	DWII_CNY_BK_F_01	*** /u01/app/oracle/oradata/orallg/DWII_CNY_BK_F_01.dbf
8 10	DWII_DPA_F_01	*** /u01/app/oracle/oradata/orallg/DWII_DPA_F_01.dbf
9 10	DWII_DPA_I_01	*** /u01/app/oracle/oradata/orallg/DWII_DPA_I_01.dbf
10 10	DWII_DPA_S_01	*** /u01/app/oracle/oradata/orallg/DWII_DPA_S_01.dbf
11 10	DWII_SOR_F_01	*** /u01/app/oracle/oradata/orallg/DWII_SOR_F_01.dbf
12 10	DWII_SOR_I_01	*** /u01/app/oracle/oradata/orallg/DWII_SOR_I_01.dbf
13 10	DW_USER	*** /u01/app/oracle/oradata/orallg/DW_USER.dbf
14 10	SQCHECK	*** /u01/app/oracle/oradata/orallg/SQCHECK.dbf
15 10	SD_CNY_D_01	*** /u01/app/oracle/oradata/orallg/SD_CNY_D_01.dbf
16 10	SD_CNY_F_01	*** /u01/app/oracle/oradata/orallg/SD_CNY_F_01.dbf
17 10	SD_DPA_D_01	*** /u01/app/oracle/oradata/orallg/SD_DPA_D_01.dbf
18 10	SD_DPA_F_01	*** /u01/app/oracle/oradata/orallg/SD_DPA_F_01.dbf

```
19  10      SD_SORT_T_01      ***      /u01/app/oracle/oradata/orallg/SD_SORT_T_01.dbf
20  10      DWII_FXDM_F_01      ***      /u01/app/oracle/oradata/orallg/DWII_FXDM_F_01.dbf
21  10      SD_SOR_T_01         ***      /u01/app/oracle/oradata/orallg/SD_SOR_T_01.dbf
```

临时文件列表

=====				
文件大小 (MB)		表空间	最大大小 (MB)	临时文件名称

1	83	TEMP	32767	/u01/app/oracle/oradata/orallg/temp01.dbf

RMAN>

RMAN> list backupset summary;

备份列表

=====									
关键字	TY	LV	S	设备类型	完成时间	段数	副本数	压缩标记	

49	B	F	A	DISK	2015-05-24 19:37:14	1	1	NO	TAG20150524T193654
50	B	F	A	DISK	2015-05-24 19:37:20	1	1	NO	TAG20150524T193719

RMAN>

RMAN>

1.4.2 建立测试表并做 truncate 误操作

AA 表空间下建立 2 个表，并建立一个索引在 users 表空间中，注意做误操作之前先切换一下日志。

```
19:38:44 SQL> alter system switch logfile;

系统已更改。

已用时间:  00: 00: 00.01
19:38:50 SQL> alter system switch logfile;

系统已更改。

已用时间:  00: 00: 00.01
19:38:54 SQL> CREATE TABLE LHR.TSPITR_TEST  TABLESPACE AA  AS SELECT * FROM SCOTT.EMP;

表已创建。

已用时间:  00: 00: 00.11
19:39:45 SQL> CREATE TABLE LHR.TSPITR_TEST1 TABLESPACE AA  AS SELECT * FROM SCOTT.EMP  where rownum<=2;

表已创建。

已用时间:  00: 00: 00.01
19:39:52 SQL> create index lhr.TSPITR_test_index  on LHR.TSPITR_TEST(empno)  tablespace users;

索引已创建。

已用时间:  00: 00: 00.00
19:39:58 SQL> select * from  LHR.TSPITR_TEST ;

      EMPNO ENAME      JOB              MGR HIREDATE              SAL              COMM              DEPTNO
-----
```


http://blog.itpub.net/26736162

7369	SMITH	CLERK	7902	1980-12-17	00:00:00	800		20
7499	ALLEN	SALESMAN	7698	1981-02-20	00:00:00	1600	300	30
7521	WARD	SALESMAN	7698	1981-02-22	00:00:00	1250	500	30
7566	JONES	MANAGER	7839	1981-04-02	00:00:00	2975		20
7654	MARTIN	SALESMAN	7698	1981-09-28	00:00:00	1250	1400	30
7698	BLAKE	MANAGER	7839	1981-05-01	00:00:00	2850		30
7782	CLARK	MANAGER	7839	1981-06-09	00:00:00	2450		10
7788	SCOTT	ANALYST	7566	1987-04-19	00:00:00	3000		20
7839	KING	PRESIDENT		1981-11-17	00:00:00	5000		10
7844	TURNER	SALESMAN	7698	1981-09-08	00:00:00	1500	0	30
7876	ADAMS	CLERK	7788	1987-05-23	00:00:00	1100		20
7900	JAMES	CLERK	7698	1981-12-03	00:00:00	950		30
7902	FORD	ANALYST	7566	1981-12-03	00:00:00	3000		20
7934	MILLER	CLERK	7782	1982-01-23	00:00:00	1300		10

已选择 14 行。

已用时间： 00: 00: 00.05
19:40:16 SQL> alter system switch logfile;

系统已更改。

已用时间： 00: 00: 00.71
19:40:50 SQL> select * from v\$log;

GROUP#	THREAD#	SEQUENCE#	BYTES	BLOCKSIZE	MEMBERS	ARC	STATUS	FIRST_CHANGE#	FIRST_TIME	NEXT_CHANGE#	NEXT_TIME
1	1	64	52428800	512	1	YES	INACTIVE	1596806	2015-05-24 19:38:50	1596809	2015-05-24 19:38:54
2	1	65	52428800	512	1	YES	ACTIVE	1596809	2015-05-24 19:38:54	1596912	2015-05-24 19:40:50
3	1	66	52428800	512	1	NO	CURRENT	1596912	2015-05-24 19:40:50	2.8147E+14	

已用时间： 00: 00: 00.01
19:40:58 SQL> truncate table LHR.TSPITR_TEST ;

表被截断。

已用时间： 00: 00: 00.09
19:41:15 SQL>
19:41:15 SQL> select * from LHR.TEST_TSPITR;

未选定行

已用时间： 00: 00: 00.00
19:41:38 SQL>

19:41:38 SQL> select * from v\$logfile;

GROUP#	STATUS	TYPE	MEMBER
3	ONLINE		/u01/app/oracle/oradata/ora11g/redo03. log
2	ONLINE		/u01/app/oracle/oradata/ora11g/redo02. log
1	ONLINE		/u01/app/oracle/oradata/ora11g/redo01. log

已用时间： 00: 00: 00.00

19:42:24 SQL>
19:42:24 SQL> CREATE TABLE LHR.TSPITR_TEST2 TABLESPACE AA AS SELECT * FROM SCOTT.EMP;

表已创建。

已用时间： 00: 00: 00.01
19:43:56 SQL> CREATE TABLE LHR.TSPITR_TEST3 TABLESPACE AA AS SELECT * FROM SCOTT.EMP;

表已创建。

已用时间： 00: 00: 00.02
19:44:04 SQL> CREATE TABLE LHR.TSPITR_TEST4 TABLESPACE AA AS SELECT * FROM SCOTT.EMP;

表已创建。

已用时间: 00: 00: 00.01
19:44:10 SQL>

这里可以看出执行误操作的时候 sql 的时间是 19:41:15 左右，而且由于我们是测试库，没有什么用户操作，所以 redo 很少，从 v\$log 中可以看出，误操作肯定在 66 号日志中。

1. 4. 3 采用 logminer 找回误删除的时间点

生产环境下为了尽量减少对数据的影响，我们需要准确定位到误操作的时间点，所以可以采用 logminer 来找回误操作的时间点，当然可以查询其他的视图来获取，但是都不是很准确，比如查询 dba_objects 中的 LAST_DDL_TIME，dba_tab_modifications 中的 timestamp，也可以查询 v\$sql，可以根据情况来决定，但是 logminer 是通用的，也是很准确的，关于 logminer 本来也很早就想写了，一直没写，后边有机会再写吧，今天我们就暂时看看怎么使用它吧。

采用 logminer 来找回误操作的时间点：

```
19:44:10 SQL> execute dbms_logmnr.add_logfile('/u01/app/oracle/oradata/orallg/redo03.log',dbms_logmnr.new);

PL/SQL 过程已成功完成。

已用时间: 00: 00: 00.08
19:45:02 SQL> exec dbms_logmnr.start_logmnr(options=>dbms_logmnr.dict_from_online_catalog);

PL/SQL 过程已成功完成。

已用时间: 00: 00: 00.10
19:45:09 SQL> select a.SCN,a.TIMESTAMP,a.SQL_REDO from v$logmnr_contents A where table_name='TSPITR_TEST' and OPERATION='DDL' order by a.SCN;
```

SCN	TIMESTAMP	SQL_REDO
1596943	2015-05-24 19:41:15	truncate table LHR.TSPITR_TEST ;

```
已用时间: 00: 00: 00.26
19:45:23 SQL>
```

可以看出执行 TRUNCATE 的时间为 2015-05-24 19:41:15，我们恢复的时候恢复到 19:41:00 。

1. 4. 4 执行 TSPITR 之前的检查

1. 4. 4. 1 检查是否自包含

```
19:45:23 SQL> BEGIN
19:47:07    2        SYS.DBMS_TTS.TRANSPORT_SET_CHECK('AA', TRUE, TRUE);
19:47:07    3    END;
19:47:07    4    /

PL/SQL 过程已成功完成。

已用时间:   00: 00: 02.86
19:47:10 SQL> SELECT * FROM SYS.TRANSPORT_SET_VIOLATIONS;

VIOLATIONS
-----
ORA-39907: 索引 LHR.TSPITR_TEST_INDEX (在表空间 USERS 中) 指向表 LHR.TSPITR_TEST (在表空间 AA 中)。

已用时间:   00: 00: 00.00
```

我们还可以这样去检查：

```
select * from SYS.TS_PITR_CHECK a WHERE a.ts1_name='AA';
```

	OBJ1_OWNER	OBJ1_NAME	OBJ1_SUBNAME	OBJ1_TYPE	TS1_NAME	OBJ2_NAME	OBJ2_SUBNAME	OBJ2_TYPE	OBJ2_OWNER	TS2_NAME	CONSTRAINT_NAME	REASON
1	LHR	TSPITR_TEST		TABLE	AA	TSPITR_TEST_INDEX		INDEX	LHR	USERS		Tables and associated indexes not fully contained in the recovery set
2	SYS	AA		TABLE	AA					-1		Sys owned tables not allowed in Recovery Set

说明，有一个索引 LHR..TSPITR_TEST_INDEX 在 users 表空间中，那么我们删除这个索引，等待恢复完成后再重建该索引。

```
19:47:29 SQL> drop index LHR.TSPITR_TEST_INDEX;

索引已删除。

已用时间:   00: 00: 00.02
19:48:46 SQL> BEGIN
19:49:11    2        SYS.DBMS_TTS.TRANSPORT_SET_CHECK('AA', TRUE, TRUE);
19:49:11    3    END;
19:49:11    4    /

PL/SQL 过程已成功完成。

已用时间:   00: 00: 02.64
19:49:14 SQL> SELECT * FROM SYS.TRANSPORT_SET_VIOLATIONS;

未选定行

已用时间:   00: 00: 00.00
19:49:20 SQL>
```

至此，自包含问题已解决。

1. 4. 4. 2 检查哪些对象执行 TSPITR 后将被删除

select * from SYS.TS_PITR_OBJECTS_TO_BE_DROPPED A WHERE a.tablespace_name='AA' and a.creation_time>='2015-05-24 19:41:00';

	OWNER		NAME		CREATION_TIME		TABLESPACE_NAME	
1	LHR	...	TSPITR_TEST2	...	2015-05-24 19:43:56	▼	AA	...
2	LHR	...	TSPITR_TEST3	...	2015-05-24 19:44:04	▼	AA	...
3	LHR	...	TSPITR_TEST4	...	2015-05-24 19:44:10	▼	AA	...

则说明若执行 TSPITR 后，AA 表空间中的表 TSPITR_TEST2、TSPITR_TEST3、TSPITR_TEST4 将被删除，那么为了保存这些对象我们将其利用 expdp 导出备份，等待 tspitr 执行完毕后再导入回来即可。

```
[oracle@rhel6_lhr ~]$ ORACLE_SID=orallg
[oracle@rhel6_lhr ~]$ expdp lhr/lhr directory=DATA_PUMP_DIR dumpfile=exptable_tspitr_lhr.dmp TABLES=TSPITR_TEST2,TSPITR_TEST3,TSPITR_TEST4 LOGFILE=exptable_tspitr_lhr.log

Export: Release 11.2.0.3.0 - Production on 星期日 5月 24 19:53:34 2015

Copyright (c) 1982, 2011, Oracle and/or its affiliates. All rights reserved.

连接到: Oracle Database 11g Enterprise Edition Release 11.2.0.3.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
启动 "LHR"."SYS_EXPORT_TABLE_01": lhr/***** directory=DATA_PUMP_DIR dumpfile=exptable_tspitr_lhr.dmp TABLES=TSPITR_TEST2,TSPITR_TEST3,TSPITR_TEST4 LOGFILE=exptable_tspitr_lhr.log
正在使用 BLOCKS 方法进行估计...
处理对象类型 TABLE_EXPORT/TABLE/TABLE_DATA
使用 BLOCKS 方法的总估计: 192 KB
处理对象类型 TABLE_EXPORT/TABLE/TABLE
. . 导出了 "LHR"."TSPITR_TEST2" 8.570 KB 14 行
. . 导出了 "LHR"."TSPITR_TEST3" 8.570 KB 14 行
. . 导出了 "LHR"."TSPITR_TEST4" 8.570 KB 14 行
已成功加载/卸载了主表 "LHR"."SYS_EXPORT_TABLE_01"
*****
LHR.SYS_EXPORT_TABLE_01 的转储文件集为:
/u01/app/oracle/admin/orallg/dpdump/exptable_tspitr_lhr.dmp
作业 "LHR"."SYS_EXPORT_TABLE_01" 已于 19:53:39 成功完成

[oracle@rhel6_lhr ~]$

[oracle@rhel6_lhr ~]$ cd /u01/app/oracle/admin/orallg/dpdump
[oracle@rhel6_lhr dpdump]$ ll -h
total 69M
-rw-r----- 1 oracle asmadmin 116 Mar 13 14:24 dp.log
-rw-r----- 1 oracle asmadmin 144K May 24 19:53 exptable_tspitr_lhr.dmp
-rw-r--r-- 1 oracle asmadmin 1.2K May 24 19:53 exptable_tspitr_lhr.log
-rw-r--r-- 1 oracle asmadmin 4.1K May 19 15:59 imp_exptest.log
-rwxr-xr-x 1 oracle oinstall 69M May 19 15:50 lhrsql20150515.dmp
[oracle@rhel6_lhr dpdump]$
```

1.4.5 执行 TSPITR

```

RMAN> RECOVER TABLESPACE AA UNTIL time "to_date('2015-05-24 19:41:00','YYYY-MM-DD HH24:mi:ss')" auxiliary destination '/u01/auxdest';

```

启动 recover 于 2015-05-24 19:57:30
使用通道 ORA_DISK_1
RMAN-05026: 警告: 假定以下表空间集适用于指定的时间点

表空间列表要求具有 UNDO 段
表空间 SYSTEM
表空间 UNDOTBS1

使用 SID='lsll' 创建自动实例

供自动实例使用的初始化参数:
db_name=ORA11G
db_unique_name=lsll tspitr ORA11G
compatible=11.2.0.0.0
db_block_size=8192
db_files=200
sga_target=280M
processes=50
db_create_file_dest=/u01/auxdest
log_archive_dest_1='location=/u01/auxdest'
#No auxiliary parameter file used

启动自动实例 ORA11G

Oracle 实例已启动

系统全局区域总计 292278272 字节

Fixed Size	2227744 字节
Variable Size	100663776 字节
Database Buffers	184549376 字节
Redo Buffers	4837376 字节

自动实例已创建
对恢复集表空间运行 TRANSPORT_SET_CHECK
TRANSPORT_SET_CHECK 已成功完成

内存脚本的内容:
{
set requested point in time
set until time "to_date('2015-05-24 19:41:00','YYYY-MM-DD HH24:mi:ss')";
restore the controlfile
restore clone controlfile;
mount the controlfile
sql clone 'alter database mount clone database';
archive current online log
sql 'alter system archive log current';
avoid unnecessary autobackups for structural changes during TSPITR
sql 'begin dbms_backup_restore.AutoBackupFlag(FALSE); end;';
}
正在执行内存脚本

正在执行命令: SET until clause

启动 restore 于 2015-05-24 19:57:37
分配的通道: ORA_AUX_DISK_1
通道 ORA_AUX_DISK_1: SID=58 设备类型=DISK

通道 ORA_AUX_DISK_1: 正在开始还原数据文件备份集
通道 ORA_AUX_DISK_1: 正在还原控制文件
通道 ORA_AUX_DISK_1: 正在读取备份片段 /u01/app/oracle/flash_recovery_area/ORA11G/autobackup/2015_05_24/o1_mf_s_880573039_bp3fzhgy_.bkp
通道 ORA_AUX_DISK_1: 段句柄 = /u01/app/oracle/flash_recovery_area/ORA11G/autobackup/2015_05_24/o1_mf_s_880573039_bp3fzhgy_.bkp 标记 = TAG20150524T193719
通道 ORA_AUX_DISK_1: 已还原备份片段 1
通道 ORA_AUX_DISK_1: 还原完成, 用时: 00:00:01
输出文件名=/u01/auxdest/ORA11G/controlfile/o1_mf_bp3h5ks3_.ctl
完成 restore 于 2015-05-24 19:57:38

sql 语句: alter database mount clone database

sql 语句: alter system archive log current

sql 语句: begin dbms_backup_restore.AutoBackupFlag(FALSE); end;

内存脚本的内容:
{
set requested point in time
set until time "to_date('2015-05-24 19:41:00','YYYY-MM-DD HH24:mi:ss')";
plsql <<<-- tspitr_2
declare
 sqlstatement varchar2(512);
 offline_not_needed exception;
 pragma exception_init(offline_not_needed, -01539);
begin
 sqlstatement := 'alter tablespace '|| 'AA' ||' offline immediate';
 krmicd.writeMsg(6162, sqlstatement);
 krmicd.execSql(sqlstatement);
exception
 when offline_not_needed then
 null;
end; >>>;
set destinations for recovery set and auxiliary set datafiles
set newname for clone datafile 1 to new;
set newname for clone datafile 3 to new;
set newname for clone datafile 2 to new;
set newname for clone tempfile 1 to new;
set newname for datafile 6 to
 "/u01/app/oracle/oradata/orallg/aa.dbf";
switch all tempfiles
switch clone tempfile all;
restore the tablespaces in the recovery set and the auxiliary set
restore clone datafile 1, 3, 2, 6;
switch clone datafile all;
}

正在执行内存脚本

正在执行命令: SET until clause

sql 语句: alter tablespace AA offline immediate

正在执行命令: SET NEWNAME

正在执行命令: SET NEWNAME

正在执行命令: SET NEWNAME

正在执行命令: SET NEWNAME

正在执行命令: SET NEWNAME

临时文件 1 在控制文件中已重命名为 /u01/auxdest/ORA11G/datafile/o1_mf_temp_%u_.tmp

启动 restore 于 2015-05-24 19:57:43

使用通道 ORA_AUX_DISK_1

通道 ORA_AUX_DISK_1: 正在开始还原数据文件备份集
通道 ORA_AUX_DISK_1: 正在指定从备份集还原的数据文件
通道 ORA_AUX_DISK_1: 将数据文件 00001 还原到 /u01/auxdest/ORA11G/datafile/ol_mf_system_%u_.dbf
通道 ORA_AUX_DISK_1: 将数据文件 00003 还原到 /u01/auxdest/ORA11G/datafile/ol_mf_undotbs1_%u_.dbf
通道 ORA_AUX_DISK_1: 将数据文件 00002 还原到 /u01/auxdest/ORA11G/datafile/ol_mf_sysaux_%u_.dbf
通道 ORA_AUX_DISK_1: 将数据文件 00006 还原到 /u01/app/oracle/oradata/ora11g/aa.dbf
通道 ORA_AUX_DISK_1: 正在读取备份片段 /u01/app/oracle/flash_recovery_area/ORA11G/backupset/2015_05_24/ol_mf_nnndf_TAG20150524T193654_bp3fypc1_.bkp
通道 ORA_AUX_DISK_1: 段句柄 = /u01/app/oracle/flash_recovery_area/ORA11G/backupset/2015_05_24/ol_mf_nnndf_TAG20150524T193654_bp3fypc1_.bkp 标记 = TAG20150524T193654
通道 ORA_AUX_DISK_1: 已还原备份片段 1
通道 ORA_AUX_DISK_1: 还原完成, 用时: 00:01:25
完成 restore 于 2015-05-24 19:59:08

数据文件 1 已转换成数据文件副本
输入数据文件副本 RECID=5 STAMP=880574348 文件名=/u01/auxdest/ORA11G/datafile/ol_mf_system_bp3h5qps_.dbf
数据文件 3 已转换成数据文件副本
输入数据文件副本 RECID=6 STAMP=880574348 文件名=/u01/auxdest/ORA11G/datafile/ol_mf_undotbs1_bp3h5qpy_.dbf
数据文件 2 已转换成数据文件副本
输入数据文件副本 RECID=7 STAMP=880574348 文件名=/u01/auxdest/ORA11G/datafile/ol_mf_sysaux_bp3h5qpw_.dbf

内存脚本的内容:
{
set requested point in time
set until time "to_date('2015-05-24 19:41:00','YYYY-MM-DD HH24:mi:ss')";
online the datafiles restored or switched
sql clone "alter database datafile 1 online";
sql clone "alter database datafile 3 online";
sql clone "alter database datafile 2 online";
sql clone "alter database datafile 6 online";
recover and open resetlogs
recover clone database tablespace "AA", "SYSTEM", "UNDOTBS1", "SYSAUX" delete archivelog;
alter clone database open resetlogs;
}

正在执行内存脚本

正在执行命令: SET until clause

sql 语句: alter database datafile 1 online

sql 语句: alter database datafile 3 online

sql 语句: alter database datafile 2 online

sql 语句: alter database datafile 6 online

启动 recover 于 2015-05-24 19:59:09
使用通道 ORA_AUX_DISK_1

正在开始介质的恢复

线程 1 序列 63 的归档日志已作为文件 /u01/app/oracle/flash_recovery_area/ORA11G/archivelog/2015_05_24/ol_mf_1_63_bp3g2b6k_.arc 存在于磁盘上
线程 1 序列 64 的归档日志已作为文件 /u01/app/oracle/flash_recovery_area/ORA11G/archivelog/2015_05_24/ol_mf_1_64_bp3g2g9r_.arc 存在于磁盘上
线程 1 序列 65 的归档日志已作为文件 /u01/app/oracle/flash_recovery_area/ORA11G/archivelog/2015_05_24/ol_mf_1_65_bp3g6280_.arc 存在于磁盘上
线程 1 序列 66 的归档日志已作为文件 /u01/app/oracle/flash_recovery_area/ORA11G/archivelog/2015_05_24/ol_mf_1_66_bp3h5qcs_.arc 存在于磁盘上
归档日志文件名=/u01/app/oracle/flash_recovery_area/ORA11G/archivelog/2015_05_24/ol_mf_1_63_bp3g2b6k_.arc 线程=1 序列=63
归档日志文件名=/u01/app/oracle/flash_recovery_area/ORA11G/archivelog/2015_05_24/ol_mf_1_64_bp3g2g9r_.arc 线程=1 序列=64
归档日志文件名=/u01/app/oracle/flash_recovery_area/ORA11G/archivelog/2015_05_24/ol_mf_1_65_bp3g6280_.arc 线程=1 序列=65
归档日志文件名=/u01/app/oracle/flash_recovery_area/ORA11G/archivelog/2015_05_24/ol_mf_1_66_bp3h5qcs_.arc 线程=1 序列=66
介质恢复完成, 用时: 00:00:01
完成 recover 于 2015-05-24 19:59:10

数据库已打开

内存脚本的内容:
{
make read only the tablespace that will be exported
sql clone 'alter tablespace AA read only';
create directory for datapump import

```
sql "create or replace directory TSPITR_DIROBJ_DPDIR as ''
/u01/auxdest''";
# create directory for datapump export
sql clone "create or replace directory TSPITR_DIROBJ_DPDIR as ''
/u01/auxdest''";
}
```

正在执行内存脚本

```
sql 语句: alter tablespace AA read only
```

```
sql 语句: create or replace directory TSPITR_DIROBJ_DPDIR as ''/u01/auxdest''
```

```
sql 语句: create or replace directory TSPITR_DIROBJ_DPDIR as ''/u01/auxdest''
```

正在执行元数据导出...

```
EXPDP> 启动 "SYS"."TSPITR_EXP_1s11":
EXPDP> 处理对象类型 TRANSPORTABLE_EXPORT/PLUGTS_BLK
EXPDP> 处理对象类型 TRANSPORTABLE_EXPORT/TABLE
EXPDP> 处理对象类型 TRANSPORTABLE_EXPORT/TABLE_STATISTICS
EXPDP> 处理对象类型 TRANSPORTABLE_EXPORT/POST_INSTANCE/PLUGTS_BLK
EXPDP> 已成功加载/卸载了主表 "SYS"."TSPITR_EXP_1s11"
EXPDP> *****
EXPDP> SYS.TSPITR_EXP_1s11 的转储文件集为:
EXPDP> /u01/auxdest/tspitr_1s11_94503.dmp
EXPDP> *****
EXPDP> 可传输表空间 AA 所需的数据文件:
EXPDP> /u01/app/oracle/oradata/orallg/aa.dbf
EXPDP> 作业 "SYS"."TSPITR_EXP_1s11" 已于 19:59:58 成功完成
```

导出完毕

内存脚本的内容:

```
{
# shutdown clone before import
shutdown clone immediate
# drop target tablespaces before importing them back
sql 'drop tablespace AA including contents keep datafiles';
}
```

正在执行内存脚本

数据库已关闭
数据库已卸装
Oracle 实例已关闭

```
sql 语句: drop tablespace AA including contents keep datafiles
```

正在执行元数据导入...

```
IMPDP> 已成功加载/卸载了主表 "SYS"."TSPITR_IMP_1s11"
IMPDP> 启动 "SYS"."TSPITR_IMP_1s11":
IMPDP> 处理对象类型 TRANSPORTABLE_EXPORT/PLUGTS_BLK
IMPDP> 处理对象类型 TRANSPORTABLE_EXPORT/TABLE
IMPDP> 处理对象类型 TRANSPORTABLE_EXPORT/TABLE_STATISTICS
IMPDP> 处理对象类型 TRANSPORTABLE_EXPORT/POST_INSTANCE/PLUGTS_BLK
IMPDP> 作业 "SYS"."TSPITR_IMP_1s11" 已于 20:00:15 成功完成
```

导入完毕

内存脚本的内容:

```
{
# make read write and offline the imported tablespaces
sql 'alter tablespace AA read write';
sql 'alter tablespace AA offline';
# enable autobackups after TSPITR is finished
sql 'begin dbms_backup_restore.AutoBackupFlag(TRUE); end;';
}
```

正在执行内存脚本


```
sql 语句: alter tablespace AA read write

sql 语句: alter tablespace AA offline

sql 语句: begin dbms_backup_restore.AutoBackupFlag(TRUE); end;
```

```
删除自动实例
自动实例已删除
已删除辅助实例文件 /u01/auxdest/ORA11G/datafile/ol_mf_temp_bp3h8mj7_.tmp
已删除辅助实例文件 /u01/auxdest/ORA11G/onlinelog/ol_mf_3_bp3h8k01_.log
已删除辅助实例文件 /u01/auxdest/ORA11G/onlinelog/ol_mf_2_bp3h8hk7_.log
已删除辅助实例文件 /u01/auxdest/ORA11G/onlinelog/ol_mf_1_bp3h8goh_.log
已删除辅助实例文件 /u01/auxdest/ORA11G/datafile/ol_mf_sysaux_bp3h5qpw_.dbf
已删除辅助实例文件 /u01/auxdest/ORA11G/datafile/ol_mf_undotbs1_bp3h5qpy_.dbf
已删除辅助实例文件 /u01/auxdest/ORA11G/datafile/ol_mf_system_bp3h5qps_.dbf
已删除辅助实例文件 /u01/auxdest/ORA11G/controlfile/ol_mf_bp3h5ks3_.ctl
完成 recover 于 2015-05-24 20:00:17
```

```
RMAN>
```

1. 4. 6 online 表空间并且导入丢失的对象

执行完恢复之后，表空间处于 offline 状态，需要将表空间 online。

```
19:49:20 SQL> select * from LHR.TSPITR_TEST;
select * from LHR.TSPITR_TEST
*
第 1 行出现错误:
ORA-00376: 此时无法读取文件 6
ORA-01110: 数据文件 6: '/u01/app/oracle/oradata/orallg/aa.dbf'
```

```
已用时间: 00: 00: 00.00
20:01:49 SQL> alter tablespace aa online;
```

表空间已更改。

```
已用时间: 00: 00: 00.05
20:02:14 SQL> select * from LHR.TSPITR_TEST ;
```

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	1980-12-17 00:00:00	800		20
7499	ALLEN	SALESMAN	7698	1981-02-20 00:00:00	1600	300	30
7521	WARD	SALESMAN	7698	1981-02-22 00:00:00	1250	500	30
7566	JONES	MANAGER	7839	1981-04-02 00:00:00	2975		20
7654	MARTIN	SALESMAN	7698	1981-09-28 00:00:00	1250	1400	30
7698	BLAKE	MANAGER	7839	1981-05-01 00:00:00	2850		30
7782	CLARK	MANAGER	7839	1981-06-09 00:00:00	2450		10
7788	SCOTT	ANALYST	7566	1987-04-19 00:00:00	3000		20
7839	KING	PRESIDENT		1981-11-17 00:00:00	5000		10
7844	TURNER	SALESMAN	7698	1981-09-08 00:00:00	1500	0	30
7876	ADAMS	CLERK	7788	1987-05-23 00:00:00	1100		20
7900	JAMES	CLERK	7698	1981-12-03 00:00:00	950		30
7902	FORD	ANALYST	7566	1981-12-03 00:00:00	3000		20
7934	MILLER	CLERK	7782	1982-01-23 00:00:00	1300		10

已选择 14 行。

已用时间: 00: 00: 00.04
20:02:53 SQL> select * from LHR.TSPITR_TEST2 ;
select * from LHR.TSPITR_TEST2
*
第 1 行出现错误:
ORA-00942: 表或视图不存在

已用时间: 00: 00: 00.00
20:03:59 SQL>

数据已经恢复，可是 LHR.TSPITR_TEST2 丢失，下边我们来导入备份的数据：

[oracle@rhel6_lhr dpdump]\$ impdp lhr/lhr directory=DATA_PUMP_DIR dumpfile=exptable_tspitr_lhr.dmp TABLES=TSPITR_TEST2,TSPITR_TEST3,TSPITR_TEST4 LOGFILE=imptable_tspitr_lhr.log

Import: Release 11.2.0.3.0 - Production on 星期日 5月 24 20:05:31 2015

Copyright (c) 1982, 2011, Oracle and/or its affiliates. All rights reserved.

连接到: Oracle Database 11g Enterprise Edition Release 11.2.0.3.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
已成功加载/卸载了主表 "LHR"."SYS_IMPORT_TABLE_01"
启动 "LHR"."SYS_IMPORT_TABLE_01": lhr/***** directory=DATA_PUMP_DIR dumpfile=exptable_tspitr_lhr.dmp TABLES=TSPITR_TEST2,TSPITR_TEST3,TSPITR_TEST4 LOGFILE=imptable_tspitr_lhr.log
处理对象类型 TABLE_EXPORT/TABLE/TABLE
处理对象类型 TABLE_EXPORT/TABLE/TABLE_DATA
.. 导入了 "LHR"."TSPITR_TEST2" 8.570 KB 14 行
.. 导入了 "LHR"."TSPITR_TEST3" 8.570 KB 14 行
.. 导入了 "LHR"."TSPITR_TEST4" 8.570 KB 14 行
作业 "LHR"."SYS_IMPORT_TABLE_01" 已于 20:05:34 成功完成

[oracle@rhel6_lhr dpdump]\$

20:03:59 SQL> select * from LHR.TSPITR_TEST2 ;

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	1980-12-17 00:00:00	800		20
7499	ALLEN	SALESMAN	7698	1981-02-20 00:00:00	1600	300	30
7521	WARD	SALESMAN	7698	1981-02-22 00:00:00	1250	500	30
7566	JONES	MANAGER	7839	1981-04-02 00:00:00	2975		20
7654	MARTIN	SALESMAN	7698	1981-09-28 00:00:00	1250	1400	30
7698	BLAKE	MANAGER	7839	1981-05-01 00:00:00	2850		30
7782	CLARK	MANAGER	7839	1981-06-09 00:00:00	2450		10
7788	SCOTT	ANALYST	7566	1987-04-19 00:00:00	3000		20
7839	KING	PRESIDENT		1981-11-17 00:00:00	5000		10
7844	TURNER	SALESMAN	7698	1981-09-08 00:00:00	1500	0	30
7876	ADAMS	CLERK	7788	1987-05-23 00:00:00	1100		20
7900	JAMES	CLERK	7698	1981-12-03 00:00:00	950		30
7902	FORD	ANALYST	7566	1981-12-03 00:00:00	3000		20
7934	MILLER	CLERK	7782	1982-01-23 00:00:00	1300		10

已选择 14 行。

已用时间: 00: 00: 00.01
20:05:55 SQL>

可以看到数据已经恢复。

1.5 与 TSPITR 有关的 OCP 试题部分

Which options must you configure while performing an automated Tablespace Point-in-Time Recovery (TSPITR) by using Recovery Manager (RMAN)? 使用恢复管理器 (RMAN) 执行自动表空间时间点恢复 (TSPITR) 的同时，你必须配置哪些选项？

- A. New channels for restore and recovery tasks
- B. New name for the data files of the tablespace
- C. Auxiliary name for the data files of the tablespace
- D. Auxiliary destinations for an auxiliary set of data files 用于辅助数据文件集的辅助目的地

http://docs.oracle.com/cd/B19306_01/backup.102/b14191/rcmtspit.htm

Answer: D

353.True or false: tablespace point-in-time recovery is possible only with RMAN.

真的还是假的：表空间时间点只能用 RMAN 恢复

- A. True
 - B. False
- Answer: B

Tablespace point-in-time recovery has been done in Oracle for some time, using various means and methods. RMAN simply automates and simplifies the process for you. 在 Oracle 表空间的时间点恢复使用各种手段和方法已经有一段时间了。 RMAN 仅为你自动化和简化过程。

354.Which command is used to begin a tablespace point-in-time recovery?

- A. Restore tablespace
- B. Recover tablespace
- C. Tablespace recover
- D. Recover to time
- E. recover datafile

Answer: B

Why should you back up a duplicated tablespace after a TSPITR is complete?

- A. The tablespace cannot be duplicated or restored to any point in time after the duplication.
- B. The tablespace cannot be duplicated or restored to the point in time before the duplication.
- C. The entire database cannot be restored after a TSPITR, so a backup is required.
- D. You cannot bring the tablespace online until its been backed up.

E. There is no requirement to do so, as RMAN will back up the tablespace after the TSPITR.

Answer: B

In what state are the datafiles of a tablespace after a TSPITR has been successfully completed?

[A. The datafiles have an ONLINE status.](#)

B. The datafiles have an OFFLINE status.

C. The datafiles have an ONLINE status and are in hot backup mode prepared for an online backup.

D. The datafiles have an OFFLINE status and are in hot backup mode for an online backup.

E. The datafiles are in STANDBY mode.

Answer: A

该题答案有误，应该选择 B。

The recovery set tablespaces are left offline for you to back up and then bring back online. These last steps follow Oracle's recommendation and best practice of backing up recovered tablespaces as soon as TSPITR completes.

Because of a logical corruption in the EMPLOYES tables, you want to perform Tablespace Point-in-Time Recovery (TSPITR) to recover the table. Before you started the TSPITR process, you queried the TS_PITR_CHECK view and you realized that the table has a referential constraint with DEPARTMENTS that exists in another tablespace, MASTERTBS.

Which two actions will permit the TSPITR to work? (Choose two.)

A.Taking the MASTERTBS tablespace offline

[B.Dropping the relationship between the tables](#)

[C.Adding the MASTERTBS tablespace to the recovery set](#)

D.Putting the MASTERTBS tablespace in read-only mode

Answer:BC

What view would you use to determine if a given tablespace is fully self-contained for the execution of a tablespace point-in-time recovery?你会用什么视图以确

定是否一个给定的表空间是完全独立的用于执行一个表空间时间点恢复？

A. TS_CHECK

B. TPITR_CHECK

C. TS_PITR_CHECK (tablespace point-in-time recovery)

D. CHECK_TSPITR

E. PITR_TS_CHECK

Answer: C

The TS_PITR_CHECK view is used to determine if a given tablespace (or tablespaces) can be independently transported or if there are other dependencies that will require the transport of additional tablespaces. TS_PITR_CHECK 视图是用来确定是否一个给定的表空间（或表）可以独立地传输或是否有其他的依赖关系，这种关系将需要额外的表空间的传输。

Which of the following restrictions are not true with respect to tablespace point-in-time recovery? (Choose all that apply .)关于表空间时间点恢复，下列哪些限制不是真的？

[A. The target database must be in NOARCHIVELOG mode. 目标数据库必须在非归档模式。](#)

[B. No backup is required of the database before you perform a TSPITR.当执行一个表空间时间点恢复之前，不需要数据库备份](#)

C. You must have all archived redo logs generated since the last backup up to the point to which you want to restore the transport set. 您必须拥有所有归档重做日志，该日志生成于自上次备

份到你想还原传输集的点。

D. If you rename a tablespace, you can not perform a TSPITR to any point in time before that rename operation occurred. 如果你重命名一个表空间 ,你不能执行 TSPITR 到在重命名操作发生之前任何一点。

E. If you have tables in tablespace_1 that have associated constraints in tablespace_2, then you must transport both tablespaces. 如果你有表在表空间 1 ,而表空间 1 与表空间 2 有相关约束 ,那么你必须传输两个表空间。

Answer: A,B

If there is a constraint between two objects in two different tablespaces, you must perform a TSPITR between the two tablespaces. As an alternative, you could disable or drop the constraint. You may not be able to reenale the constraint with validation after the TSPITR, however. 如果有两个不同的表空间中的两个对象之间的约束 ,你必须执行两个表之间的 TSPITR 作为替代 ,你可以禁用或删除该约束。然而您可能无法 TSPITR 后重新启用验证约束。

After you perform a TSPITR, you should back up the tablespace/datafile. If you do not, you will not be able to do a TSPITR to any point in time before the original TSPITR 您执行表空间时间点恢复后 ,你应该备份的表空间/数据文件。如果你不这样做 ,在原表空间时间点恢复之前 ,你就不能进行一个表空间时间点恢复到任何时间点。

After a TSPITR each datafile associated with the TSPITR will be offline. Oracle recommends you back up the datafile before bringing it online.

Because of a logical corruption in your production database, you wanted to perform Tablespace Point in Time Recovery (TSPITR). But before you start the recovery, you queried the TS_PITR_OBJECTS_TO_BE_DROPPED view and realized that there are a large number of objects that would be dropped when you start the recovery by using this method. 由于生产数据库逻辑损坏 ,你想要执行表空间时间点恢复 (TSPITR)。但在开始恢复之前 ,您查询了表空间时间点恢复删除对象视图 ,并意识到当你开始使用此方法恢复时有大量的对象会被丢弃。

You want to preserve these objects. Which option must you use to perform TSPITR and preserve the object?

要保留这些对象。你必须使用哪一个选项来执行表空间时间点恢复及保留对象 ?

[A. Perform Export before TSPITR and Import after TSPITR](#)

[表空间时间点恢复前执行导出和表空间时间点恢复后执行导入](#)

B. Move objects to another schema that has the same tablespace assigned

C. Perform Incomplete Recovery before TSPITR with the Log Sequence Number (LSN)

D. Perform Incomplete Recovery before TSPITR with the System Change Number (SCN)

Answer: A

(参考: http://docs.oracle.com/cd/E11882_01/backup.112/e10642/rcmtspit.htm#BRADV89795)

When you perform RMAN TSPITR on a tablespace, objects created after the target recovery time are lost. You can preserve such objects after they are identified by exporting them before TSPITR with the Data Pump Export utility and reimporting them afterward with Data Pump Import.

To determine which objects are lost in TSPITR, query the TS_PITR_OBJECTS_TO_BE_DROPPED view on the primary database.)

In Recovery Manager (RMAN), you are taking image copies of the data files of your production database and rolling them forward at regular intervals. You attempt to restart your database. After a regular maintenance task, you realize that one of the data files that belongs to the USERS tablespace is damaged and you need to recover the data file by using the image copy. Because a media failure caused the data file to be damaged, you want to place the data file in a different location while restoring the file. Which option must you consider for this task?

在恢复管理器(RMAN) , 你正在为你的生产数据库的数据文件制作映像副本，并定期向前滚动。您尝试重新启动数据库。定期维护任务后，你会发现一个属于用户表空间的数据文件被损坏，你需要通过使用映像副本去恢复数据文件。由于介质故障造成数据文件损坏，你想要把数据文件放置在不同的位置，同时恢复文件。你必须为这个任务考虑哪个选项？

- A. using only the RMAN SWITCH command to set the new location for the data file
- B. placing the database in the MOUNT state for the restore and recovery operations.

C. using an RMAN RUN block with the SET NEWNAME and then the SWITCH command. 使用带有设置新名字然后切换命令的 RMAN 运行块。Copy 了一份，取个新名字，然后切换过去。

- D. configuring two channels: one for the restore operation and the other for the recovery operation

Answer: C

In your production database, users report that they are unable to generate reports on an important table because it does not contain any data. While investigating the reason, you realize that another user executed the TRUNCATE TABLE command, which accidentally caused the data to be lost. Now you want to recover the lost data of the table without affecting objects in other

schemas. 在生产数据库中，用户报告他们无法在一个重要的表上生成报告，因为它不包含任何数据。虽然调查的原因，你会发现另一个用户执行删减表命令，造成数据意外丢失。

现在你要恢复表中丢失的数据，而不会影响其他方案中的对象。

Which method must you use to recover the lost data?

你必须使用哪种方法来恢复丢失的数据？

- A. Complete Recovery with online redo log
- B. Complete Recovery with archived redo log

C. Tablespace Point-in-Time Recovery (TSPITR) 表空间时间点恢复 (TSPITR)

- D. Incomplete Recovery with system change number (SCN)

Answer: C

表空间时间点恢复或简称 TSPITR，是一种仅在部分数据库上执行不完整恢复的技术。作为一条基本规则，不完整恢复必须应用到整个数据库：如第 16 章所述，必须还原所有的数据文件并且一起向前滚动。TSPITR 依据目标数据库的表空间的一个子集来创建辅助数据库，仅在此子集上执行不完整的恢复，然后使用辅助数据库的表空间来替代目标数据库中的表空间。最终结果看起来好像仅恢复和还原了该子集，保持目标数据库的其余部分是最新的。手动完成此任务将非常繁琐。

1.6 总结

TSPITR 虽然很简单，但是有些小的细节还是应该需要注意一下的，比如我实验的时候就犯了一个错误，导致这个实验花了很久才完成。

1、备份的时候需要注意的地方:备份完成后需要做切换日志的操作，否则做 truncate 操作和备份有可能处于同一个 scn，会导致最后报错，找不见备份文件。

<http://www.itpub.net/thread-1922807-1-1.html>

```
[oracle@rhel6_lhr dpdump]$ echo $ORACLE_SID
orallg
[oracle@rhel6_lhr dpdump]$ rman target /
```

恢复管理器: Release 11.2.0.3.0 - Production on 星期日 5月 24 17:10:51 2015

Copyright (c) 1982, 2011, Oracle and/or its affiliates. All rights reserved.

已连接到目标数据库: ORA11G (DBID=4270446895)

```
RMAN> RECOVER TABLESPACE AA
2> UNTIL time "to_date('2015-05-24 15:35:00','YYYY-MM-DD HH24:mi:ss')"
3> AUXILIARY DESTINATION '/u01/auxdest';
```

启动 recover 于 2015-05-24 17:10:59

使用目标数据库控制文件替代恢复目录

分配的通道: ORA_DISK_1

通道 ORA_DISK_1: SID=31 设备类型=DISK

RMAN-05026: 警告: 假定以下表空间集适用于指定的时间点

表空间列表要求具有 UNDO 段

表空间 SYSTEM

表空间 UNDOTBS1

使用 SID='pExF' 创建自动实例

供自动实例使用的初始化参数:

db_name=ORA11G

db_unique_name=pExF_tspitr_ORA11G

compatible=11.2.0.0.0

db_block_size=8192

db_files=200

sga_target=280M

processes=50

db_create_file_dest=/u01/auxdest

log_archive_dest_1='location=/u01/auxdest'

#No auxiliary parameter file used

启动自动实例 ORA11G

Oracle 实例已启动

系统全局区域总计	292278272 字节
Fixed Size	2227744 字节
Variable Size	100663776 字节
Database Buffers	184549376 字节
Redo Buffers	4837376 字节

自动实例已创建
对恢复集表空间运行 TRANSPORT_SET_CHECK
TRANSPORT_SET_CHECK 已成功完成

```
内存脚本的内容：
{
# set requested point in time
set until    time "to_date('2015-05-24 15:35:00','YYYY-MM-DD HH24:mi:ss')";
# restore the controlfile
restore clone controlfile;
# mount the controlfile
sql clone 'alter database mount clone database';
# archive current online log
sql 'alter system archive log current';
# avoid unnecessary autobackups for structural changes during TSPITR
sql 'begin dbms_backup_restore.AutoBackupFlag(FALSE); end;';
}
```

正在执行内存脚本

正在执行命令: SET until clause

启动 restore 于 2015-05-24 17:11:03
分配的通道: ORA_AUX_DISK_1
通道 ORA_AUX_DISK_1: SID=58 设备类型=DISK

通道 ORA_AUX_DISK_1: 正在开始还原数据文件备份集
通道 ORA_AUX_DISK_1: 正在还原控制文件
通道 ORA_AUX_DISK_1: 正在读取备份片段 /u01/app/oracle/flash_recovery_area/ORA11G/backupset/2015_05_24/o1_mf_ncsnf_TAG20150524T151723_bp2yx059_.bkp
通道 ORA_AUX_DISK_1: 段句柄 = /u01/app/oracle/flash_recovery_area/ORA11G/backupset/2015_05_24/o1_mf_ncsnf_TAG20150524T151723_bp2yx059_.bkp 标记 = TAG20150524T151723
通道 ORA_AUX_DISK_1: 已还原备份片段 1
通道 ORA_AUX_DISK_1: 还原完成, 用时: 00:00:01
输出文件名=/u01/auxdest/ORA11G/controlfile/o1_mf_bp35f83l_.ctl
完成 restore 于 2015-05-24 17:11:05

sql 语句: alter database mount clone database

sql 语句: alter system archive log current

sql 语句: begin dbms_backup_restore.AutoBackupFlag(FALSE); end;

内存脚本的内容:

```
{
# set requested point in time
set until      time "to_date('2015-05-24 15:35:00','YYYY-MM-DD HH24:mi:ss')";
plsql <<<-- tspitr_2
declare
    sqlstatement          varchar2(512);
    offline_not_needed exception;
    pragma exception_init(offline_not_needed, -01539);
begin
    sqlstatement := 'alter tablespace '||      'AA' ||' offline immediate';
    krmicd.writeMsg(6162, sqlstatement);
    krmicd.execSql(sqlstatement);
exception
    when offline_not_needed then
        null;
end; >>>;
# set destinations for recovery set and auxiliary set datafiles
set newname for clone datafile      1 to new;
set newname for clone datafile      3 to new;
set newname for clone datafile      2 to new;
set newname for clone tempfile      1 to new;
set newname for datafile      6 to
"/u01/app/oracle/oradata/orallg/aa.dbf";
# switch all tempfiles
switch clone tempfile all;
# restore the tablespaces in the recovery set and the auxiliary set
restore clone datafile      1, 3, 2, 6;
switch clone datafile all;
}
```

正在执行内存脚本

正在执行命令: SET until clause

sql 语句: alter tablespace AA offline immediate

正在执行命令: SET NEWNAME

正在执行命令: SET NEWNAME

正在执行命令: SET NEWNAME

正在执行命令: SET NEWNAME

正在执行命令：SET NEWNAME

临时文件 1 在控制文件中已重命名为 /u01/auxdest/ORAl1G/datafile/o1_mf_temp_%u_.tmp

启动 restore 于 2015-05-24 17:11:09
使用通道 ORA_AUX_DISK_1

无法删除一个或多个辅助集数据文件

删除自动实例
关闭自动实例
数据库已卸装
Oracle 实例已关闭
自动实例已删除
已删除辅助实例文件 /u01/auxdest/ORAl1G/controlfile/o1_mf_bp35f83l_.ctl
RMAN-00571: =====
RMAN-00569: ===== ERROR MESSAGE STACK FOLLOWS =====
RMAN-00571: =====
RMAN-03002: recover 命令（在 05/24/2015 17:11:14 上）失败
RMAN-03015: 在存储的脚本 Memory Script 中出现错误
RMAN-06026: 有些目标没有找到 - 终止还原
RMAN-06023: 没有找到数据文件 3 的副本来还原
RMAN-06023: 没有找到数据文件 1 的副本来还原

RMAN> list backupset of datafile 1,3,2,6;

备份集列表
=====

BS 关键字	类型	LV	大小	设备类型	经过时间	完成时间
14	Full		1.17G	DISK	00:02:30	2015-05-24 15:19:53
BP 关键字: 14 状态: AVAILABLE 已压缩: NO 标记: TAG20150524T151723						
段名:/u01/app/oracle/flash_recovery_area/ORAl1G/backupset/2015_05_24/o1_mf_nnndf_TAG20150524T151723_bp2yrbww_.bkp						
备份集 14 中的数据文件列表						
文件	LV	类型	Ckp	SCN	Ckp 时间	名称
1		Full		1515679	2015-05-24 15:17:23	/u01/app/oracle/oradata/oral1g/system01.dbf
2		Full		1515679	2015-05-24 15:17:23	/u01/app/oracle/oradata/oral1g/sysaux01.dbf
3		Full		1515679	2015-05-24 15:17:23	/u01/app/oracle/oradata/oral1g/undotbs01.dbf
6		Full		1515679	2015-05-24 15:17:23	/u01/app/oracle/oradata/oral1g/aa.dbf

RMAN>

1.7 RMAN 系列参考文章

RMAN 备份恢复系列	
【推荐】 【RMAN】rm -rf 误操作的恢复过程	http://blog.itpub.net/26736162/viewspace-1623938/
【推荐】 【RMAN】利用备份片还原数据库(中)-附加	http://blog.itpub.net/26736162/viewspace-1621938/
【推荐】 【RMAN】利用备份片还原数据库(下)	http://blog.itpub.net/26736162/viewspace-1621672/
【推荐】 【RMAN】利用备份片还原数据库(中)	http://blog.itpub.net/26736162/viewspace-1621661/
【推荐】 【RMAN】利用备份片还原数据库(上)	http://blog.itpub.net/26736162/viewspace-1621581/
【推荐】 【RMAN】RMAN 跨版本恢复(下)	http://blog.itpub.net/26736162/viewspace-1562583/
【推荐】 Oracle 组件 系列 小结	http://blog.itpub.net/26736162/viewspace-1562441/
【推荐】 【RMAN】RMAN 跨版本恢复(中)	http://blog.itpub.net/26736162/viewspace-1561352/
【推荐】 【RMAN】RMAN 跨版本恢复(上)	http://blog.itpub.net/26736162/viewspace-1561185/
【推荐】 关于在不同版本和平台之间进行还原或复制的常见问题	http://blog.itpub.net/26736162/viewspace-1549041/
【推荐】 undo 表空间文件丢失恢复（4）--无备份无 recover 的情况下恢复	http://blog.itpub.net/26736162/viewspace-1458787/
【推荐】 undo 表空间文件丢失恢复（3）--无备份无 redo 的情况下恢复	http://blog.itpub.net/26736162/viewspace-1458750/
【推荐】 undo 表空间文件丢失恢复（2）--无备份有 redo 的情况下恢复	http://blog.itpub.net/26736162/viewspace-1458663/
【推荐】 undo 表空间文件丢失恢复（1）--有备份	http://blog.itpub.net/26736162/viewspace-1458654/
【推荐】 ORACLE 数据泵之 NETWORK_LINK	http://blog.itpub.net/26736162/viewspace-1432591/
【推荐】 oracle 控制文件在缺失归档日志的情况下的恢复	http://blog.itpub.net/26736162/viewspace-1426552/
【推荐】 ORACLE 只读数据文件备份与恢复	http://blog.itpub.net/26736162/viewspace-1425283/
【推荐】 热备下的测试库搭建	http://blog.itpub.net/26736162/viewspace-1405324/
【推荐】 oracle 异构平台迁移之传输表空间一例	http://blog.itpub.net/26736162/viewspace-1391913/
【推荐】 oracle 传输表空间一例	http://blog.itpub.net/26736162/viewspace-1375260/
【推荐】 利用 rman 来实现 linux 平台数据库复制到 windows 平台数据库	http://blog.itpub.net/26736162/viewspace-1352436/
【推荐】 直接复制数据文件实现 linux 平台数据库复制到 windows 平台数据库	http://blog.itpub.net/26736162/viewspace-1352243/
【推荐】 使用 OEM 复制数据库	http://blog.itpub.net/26736162/viewspace-1224865/
【推荐】 采用 DUPLICATE 把 asm 数据库复制到文件系统	http://blog.itpub.net/26736162/viewspace-1224861/
Duplicating a Database Without Recovery Catalog or Target Connection	http://blog.itpub.net/26736162/viewspace-1223253/
【推荐】 Duplicating an Active Database	http://blog.itpub.net/26736162/viewspace-1223247/

1.8 about me

.....

本文作者：小麦苗，只专注于数据库的技术，更注重技术的运用

ITPUB BLOG：<http://blog.itpub.net/26736162>

本文地址：

本文pdf版：<http://yunpan.cn/QCwUAI9bn7g7w> 提取码：af2d

QQ：642808185 若加 QQ 请注明你所正在读的文章标题

创作时间地点：2015-05-24 09:00~ 2015-05-14 22:00 于唐镇金唐公寓宿舍

<版权所有，文章允许转载，但须以链接方式注明源地址，否则追究法律责任!>

.....