

(053第6章)Manul Recovery

1.理解restore和recovery

restore 还原：将备份文件还原到数据库（某个时间点）

recovery 恢复：利用redo log(归档日志和联机日志)进行前滚和回滚，恢复数据到当前状态

2.丢失临时数据文件的恢复

删除临时数据文件后执行需要使用临时空间的sql时报错！

```
SQL> select file_name from dba_temp_files;

FILE_NAME
-----
/u01/app/oracle/oradata/sztech1/temp01.dbf

SQL> ! rm -rf /u01/app/oracle/oradata/sztech1/temp01.dbf  删除临时数据文件

SQL> select department_id,job_id,manager_id,sum(salary) from hr.employees
2 group by grouping sets ((department_id,job_id),(job_id,manager_id));
select department_id,job_id,manager_id,sum(salary) from hr.employees
*
ERROR at line 1:
ORA-01116: error in opening database file 201
ORA-01110: data file 201: '/u01/app/oracle/oradata/sztech1/temp01.dbf'
ORA-27041: unable to open file
Linux Error: 2: No such file or directory
Additional information: 3
```

①重新启动实例后自动创建临时数据文件

通过在启动实例OPEN数据库时跟踪报警文件alert_SID.log可以看到系统自动重建了临时文件。

tail -f \$ORACLE_BASE/diag/rdbms/sztech1/sztech1/trace/alert_sztech1.log

```
[oracle@dbserver ~]$ tail -f /u01/app/oracle/diag/rdbms/sztech1/sztech1/trace/alert_sztech1.log
Thu Jul 27 15:54:31 2017
SMON: enabling cache recovery
Archived Log entry 105 added for thread 1 sequence 110 ID 0xc1ead992 dest 1:
[6895] Successfully onlined Undo Tablespace 2.
Undo initialization finished serial:0 start:23670714 end:23670824 diff:110 (1 seconds)
Verifying file header compatibility for 11g tablespace encryption..
Verifying 11g file header compatibility for tablespace encryption completed
SMON: enabling tx recovery
Re-creating tempfile /u01/app/oracle/oradata/sztech1/temp01.dbf 自动重建了临时数据文件
Database Characterset is ZHS16GBK
No Resource Manager plan active
replication_dependency_tracking turned off (no async multimaster replication found)
Starting background process QMNC
Thu Jul 27 15:54:32 2017
```

②手动处理（实例不允许重启的场景）

为临时表空间增加一个数据文件，再删除原来丢失的数据文件。

```
SQL> alter tablespace temp add tempfile '/u01/app/oracle/oradata/sztech1/temp02.dbf' size 100m;
Tablespace altered.

SQL> alter tablespace temp drop tempfile '/u01/app/oracle/oradata/sztech1/temp01.dbf';
Tablespace altered.
```

3.日志组丢失的恢复（是指整组成员全部丢失）

①日志组的状态

current :当前被后台进程LGWR使用的日志组

active: 当前不再写入但仍然被实例使用（用于实例恢复）的日志组

inactive:当前不再写入，也不用于实例恢复的日志组。

```
SQL> select group#,sequence#,members,status,archived from v$log; 查看日志组状态
```

GROUP#	SEQUENCE#	MEMBERS	STATUS	ARC
1	111	2	ACTIVE	YES
2	112	2	CURRENT	NO
3	110	2	INACTIVE	YES

```
SQL> alter system switch logfile; 切换日志组
```

system altered.

```
SQL> select group#,sequence#,members,status,archived from v$log; 再次查看日志组状态
```

GROUP#	SEQUENCE#	MEMBERS	STATUS	ARC
1	111	2	ACTIVE	YES
2	112	2	ACTIVE	YES
3	113	2	CURRENT	NO

```
SQL> alter system checkpoint; 强制检查点，日志状态变为INACTIVE
```

system altered.

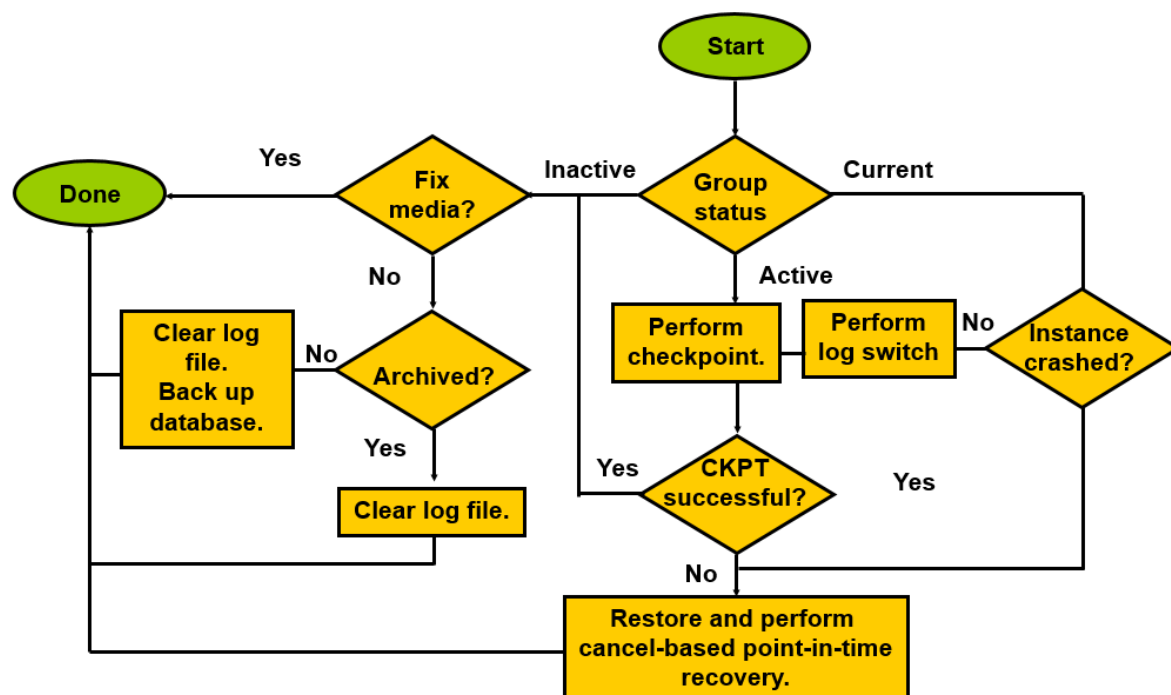
```
SQL> select group#,sequence#,members,status,archived from v$log; 再次查看日志组状态
```

GROUP#	SEQUENCE#	MEMBERS	STATUS	ARC
1	111	2	INACTIVE	YES
2	112	2	INACTIVE	YES
3	113	2	CURRENT	NO

②日志组丢失的恢复

原理图：

Recovering from the Loss of a Redo Log Group



a.丢失Inactive日志组的恢复

```
SQL> select member from v$logfile;
MEMBER
-----
/u01/app/oracle/oradata/sztech1/redo03.log
/u01/app/oracle/oradata/sztech1/redo02.log
/u01/app/oracle/oradata/sztech1/redo01.log

SQL> select group#,sequence#,members,status,archived from v$log;
  GROUP#  SEQUENCE#  MEMBERS STATUS      ARC
-----
1         114         1 CURRENT      NO
2         112         1 INACTIVE     YES 已归档
3         113         1 ACTIVE      YES

SQL> ! rm -rf /u01/app/oracle/oradata/sztech1/redo02.log 删除第2组的日志文件

SQL> ! ls -ltr /u01/app/oracle/oradata/sztech1/redo02.log
ls: cannot access /u01/app/oracle/oradata/sztech1/redo02.log: No such file or directory

SQL> alter database clear logfile group 2; 直接清除第2组系统自动创建文件
Database altered.

SQL> ! ls -ltr /u01/app/oracle/oradata/sztech1/redo02.log 检验是否创建日志文件
-rw-r----- 1 oracle oinstall 52429312 Jul 27 16:29 /u01/app/oracle/oradata/sztech1/redo02.log

SQL> select group#,sequence#,members,status,archived from v$log;
  GROUP#  SEQUENCE#  MEMBERS STATUS      ARC
-----
1         114         1 CURRENT      NO
2          0         1 UNUSED      YES 第2组的状态变成了unused,且日志序号为0
3         113         1 INACTIVE     YES

SQL> alter system switch logfile; 切换日志组
system altered.

SQL> select group#,sequence#,members,status,archived from v$log;
  GROUP#  SEQUENCE#  MEMBERS STATUS      ARC
-----
1         114         1 ACTIVE      YES
2         115         1 CURRENT      NO 可以正常切换,没有问题!
3         113         1 INACTIVE     YES
```

b.丢失active状态的日志组文件的恢复

```
SQL> select group#,sequence#,members,status,archived from v$log;
  GROUP#  SEQUENCE#  MEMBERS STATUS      ARC
-----
1         114         1 INACTIVE     YES
2         115         1 ACTIVE      YES 第2组为active状态
3         116         1 CURRENT      NO

SQL> ! rm -rf '/u01/app/oracle/oradata/sztech1/redo02.log' 删除第2组日志文件

SQL> ! ls -ltr '/u01/app/oracle/oradata/sztech1/redo02.log'
ls: cannot access /u01/app/oracle/oradata/sztech1/redo02.log: No such file or directory

SQL> alter system checkpoint; 执行检查点
system altered.

SQL> select group#,sequence#,members,status,archived from v$log; 查看第2组状态
  GROUP#  SEQUENCE#  MEMBERS STATUS      ARC
-----
1         114         1 INACTIVE     YES
2         115         1 INACTIVE     YES
3         116         1 CURRENT      NO

SQL> alter database clear logfile group 2; 清除第2组, 系统自动创建日志文件
Database altered.

SQL> ! ls -ltr '/u01/app/oracle/oradata/sztech1/redo02.log' 检查日志文件是否创建成功
-rw-r----- 1 oracle oinstall 52429312 Jul 27 16:41 /u01/app/oracle/oradata/sztech1/redo02.log

SQL> select group#,sequence#,members,status,archived from v$log;
  GROUP#  SEQUENCE#  MEMBERS STATUS      ARC
-----
1         114         1 INACTIVE     YES
2          0         1 UNUSED      YES
3         116         1 CURRENT      NO
```

c.丢失current日志组文件的恢复

```

SQL> select member from v$logfile;

MEMBER
-----
/u01/app/oracle/oradata/sztech1/redo03.log
/u01/app/oracle/oradata/sztech1/redo02.log
/u01/app/oracle/oradata/sztech1/redo01.log

SQL> select group#,sequence#,members,status,archived from v$log;

  GROUP#  SEQUENCE#  MEMBERS STATUS      ARC
-----
      1         124         1 CURRENT      NO
      2         123         1 INACTIVE     YES
      3         122         1 INACTIVE     YES

SQL> ! rm -rf /u01/app/oracle/oradata/sztech1/redo01.log 删除第1组的日志文件

SQL> ! ls -ltr /u01/app/oracle/oradata/sztech1/redo01.log
ls: cannot access /u01/app/oracle/oradata/sztech1/redo01.log: No such file or directory

SQL> alter system switch logfile; 切换日志组

system altered.

SQL> select group#,sequence#,members,status,archived from v$log;

  GROUP#  SEQUENCE#  MEMBERS STATUS      ARC
-----
      1         124         1 ACTIVE      NO
      2         123         1 INACTIVE     YES
      3         125         1 CURRENT      NO

SQL> alter system checkpoint; 执行检查点

system altered.

SQL> select group#,sequence#,members,status,archived from v$log;

  GROUP#  SEQUENCE#  MEMBERS STATUS      ARC
-----
      1         124         1 INACTIVE     NO
      2         123         1 INACTIVE     YES
      3         125         1 CURRENT      NO

SQL> alter database clear unarchived logfile group 1; 清除未归档的日志组1，必须加unarchived参数，否则
Database altered. 无法成功清除该日志组。清除后必须立即备份数据库！

SQL> ! ls -ltr /u01/app/oracle/oradata/sztech1/redo01.log
-rw-r----- 1 oracle oinstall 52429312 Jul 27 17:14 /u01/app/oracle/oradata/sztech1/redo01.log

```

--清除未归档的日志组

ALTER DATABASE CLEAR UNARCHIVED LOGFILE

--清除未归档的日志组，但如果有数据文件处于离线状态而且正好需要该组日志用于重新在线时，可能需要加UNRECOVERABLE

DATAFILE，但是这些离线数据文件永远无法重新在线！

ALTER DATABASE CLEAR UNARCHIVED LOGFILE UNRECOVERABLE DATAFILE

UNRECOVERABLE DATAFILE实例：

①离线表空间

```
SQL> alter tablespace ts_inventory offline;  
Tablespace altered.
```

②查询并删除当前日志组1

```
SQL> select group#,sequence#,archived,status,members from v$log;  
GROUP# SEQUENCE# ARC STATUS MEMBERS  
-----  
1      135 NO    CURRENT      1  
2      134 YES   INACTIVE     1  
3      133 YES   INACTIVE     1
```

```
SQL> !rm -rf /u01/app/oracle/oradata/sztech1/redo01.log
```

```
SQL> !ls -ltr /u01/app/oracle/oradata/sztech1/redo01.log
```

```
ls: cannot access /u01/app/oracle/oradata/sztech1/redo01.log: No such file or directory
```

③切换日志组

```
SQL> alter system switch logfile;  
System altered.
```

```
SQL> select group#,sequence#,archived,status,members from v$log;  
GROUP# SEQUENCE# ARC STATUS MEMBERS  
-----  
1      135 NO    ACTIVE      1  
2      134 YES   INACTIVE     1  
3      136 NO    CURRENT      1
```

④强制检查点

```
SQL> alter system checkpoint;  
System altered.
```

```
SQL> select group#,sequence#,archived,status,members from v$log;  
GROUP# SEQUENCE# ARC STATUS MEMBERS  
-----  
1      135 NO    INACTIVE     1  
2      134 YES   INACTIVE     1  
3      136 NO    CURRENT      1
```

⑤清除未归档的删除日志组1

```
SQL> alter database clear unarchived logfile group 1;  
alter database clear unarchived logfile group 1  
*  
ERROR at line 1:  
ORA-00393: log 1 of thread 1 is needed for recovery of offline datafiles  
ORA-00312: online log 1 thread 1: '/u01/app/oracle/oradata/sztech1/redo01.log'  
ORA-01110: data file 6: '/u01/app/oracle/oradata/sztech1/ts_inventory01.dbf'  
清除日志组报错，因为存在离线数据文件！
```

⑥尝试对表空间在线

```
SQL> alter tablespace ts_inventory online;  
Tablespace altered.
```

⑦在线表空间成功，再次清除未归档的删除日志组1

```
SQL> alter database clear unarchived logfile group 1;  
Database altered.
```

如果在线表空间无法成功完成，只能执行下面的命令，但是离线的表空间便再无法成功在线了！

```
alter database clear unarchived logfile group 1 unrecoverable datafile;
```

```
SQL> select group#,sequence#,archived,status,members from v$log;  
GROUP# SEQUENCE# ARC STATUS MEMBERS  
-----  
1      0 YES   UNUSED      1  
2      134 YES   INACTIVE     1  
3      136 NO    CURRENT      1
```

4.索引表空间丢失的恢复

①直接删除该表空间

②重建索引到新的表空间，然后重新收集统计信息

```
SQL> create index hr.idx_emp_lastname on hr.employees(last_name) nologging;
Index created.
SQL> alter index hr.idx_emp_lastname rebuild online;
Index altered.
SQL> create index hr.idx_emp_firstname on hr.employees(first_name) nologging parallel 4; 创建索引加并行
Index created.
SQL> alter index hr.idx_emp_firstname rebuild online; 在线重建索引。
Index altered.
```

5. 密码文件丢失的恢复

① ORACLE 支持口令认证和 OS 认证，并且 OS 认证优先。

② 在 LINUX 环境中修改 sqlnet.ora，仅使用口令认证。

在 \$ORACLE_HOME/network/admin/sqlnet.ora 中加入： sqlnet.authentication_services=(none)

③ 尝试登录数据库

```
[oracle@dbserver ~]$ sqlplus /nolog
SQL*Plus: Release 11.2.0.4.0 Production on Wed Aug 2 10:31:11 2017
Copyright (c) 1982, 2013, Oracle. All rights reserved.

SQL> conn / as sysdba 操作系统认证被忽略
ERROR:
ORA-01017: invalid username/password; logon denied

SQL> conn sjdfldsj/sdfjads1kf as sysdba 用户密码错误不允许登录
ERROR:
ORA-01017: invalid username/password; logon denied

SQL> conn sys/oracle as sysdba 需要输入正确的用户密码才能登录
Connected.
SQL> █
```

④ 口令文件删除后恢复：

```
[oracle@dbserver ~]$ rm -rf $ORACLE_HOME/dbs/orapwsztech1 删除口令文件
[oracle@dbserver ~]$ sqlplus sys/oracle as sysdba

SQL*Plus: Release 11.2.0.4.0 Production on Wed Aug 2 10:42:45 2017
Copyright (c) 1982, 2013, Oracle. All rights reserved.

ERROR:
ORA-01017: invalid username/password; logon denied 管理员登录失败

Enter user-name:
[oracle@dbserver ~]$ orapwd file=$ORACLE_HOME/dbs/orapwsztech1 password=oracle entries=5 force=y
[oracle@dbserver ~]$ sqlplus sys/oracle as sysdba

SQL*Plus: Release 11.2.0.4.0 Production on Wed Aug 2 10:43:04 2017
Copyright (c) 1982, 2013, Oracle. All rights reserved.

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.4.0 - Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options

SQL> 重建口令文件后可成功登录
```

SQL> select * from v\$pwfile_users; 查询口令文件中管理员信息

USERNAME	SYSDB	SYSOP	SYSAS
SYS	TRUE	TRUE	FALSE

⑤ 查看口令文件内容（无法看到口令明文）

```
[oracle@dbserver dbs]$ strings orapwsztech1
]\[z
ORACLE Remote Password file
INTERNAL
AB27B53EDC5FEF41
8A8F025737A9097A
[oracle@dbserver dbs]$
```

6. 完整恢复和不完整恢复（基于时间点的恢复）

完整恢复：还原最后的备份后再应用归档日志和联机日志恢复到故障点，不会丢失数据。

不完整恢复：还原备份后应用部分归档日志恢复指定的过去的某个时间点，会丢数据。

操作步骤：

还原备份->应用全部（完整恢复）或者部分归档(不完整恢复)和联机日志->自动前滚->open->自动回滚->恢复完成！

7.只读表空间的恢复

使用手工方式进行恢复，不需要将表空间修改到备份模式，直接进行恢复。

8.恢复nologging数据文件或表空间

无法恢复nologging类型的数据文件或表空间。

9.恢复所有控制文件的丢失

①查看当前控制文件信息

SQL> show parameter control_files;

NAME	TYPE	VALUE
control_files	string	/u01/app/oracle/oradata/sztech1/control01.ctl, /u01/app/oracle/fast_recovery_area/sztech1/control02.ctl

②备份当前控制文件到指定文件中

SQL> alter database backup controlfile to trace as '/home/oracle/control01.sql';

Database altered.

③删除所有的控制文件

SQL> ! rm -rf /u01/app/oracle/oradata/sztech1/control01.ctl

SQL> ! rm -rf /u01/app/oracle/fast_recovery_area/sztech1/control02.ctl

④强制关闭数据库，无法使用其它方式正常关闭数据库

SQL> shutdown abort

ORACLE instance shut down.

⑤参见之前的控制文件备份信息进行执行恢复

more /home/oracle/control01.sql

```
--
-- Set #1. NORESETLOGS case
--
-- The following commands will create a new control file and use it
-- to open the database.
-- Data used by Recovery Manager will be lost.
-- Additional logs may be required for media recovery of offline
-- Use this only if the current versions of all online logs are
-- available.
--
-- After mounting the created controlfile, the following SQL
-- statement will place the database in the appropriate
-- protection mode:
-- ALTER DATABASE SET STANDBY DATABASE TO MAXIMIZE PERFORMANCE
STARTUP NOMOUNT
CREATE CONTROLFILE REUSE DATABASE "SZTECH1" NORESETLOGS ARCHIVELOG
    MAXLOGFILES 16
    MAXLOGMEMBERS 3
    MAXDATAFILES 100
    MAXINSTANCES 8
    MAXLOGHISTORY 292
LOGFILE
  GROUP 1 '/u01/app/oracle/oradata/sztech1/redo01.log' SIZE 50M BLOCKSIZE 512,
  GROUP 2 '/u01/app/oracle/oradata/sztech1/redo02.log' SIZE 50M BLOCKSIZE 512,
  GROUP 3 '/u01/app/oracle/oradata/sztech1/redo03.log' SIZE 50M BLOCKSIZE 512
-- STANDBY LOGFILE
DATAFILE
  '/u01/app/oracle/oradata/sztech1/system01.dbf',
  '/u01/app/oracle/oradata/sztech1/sysaux01.dbf',
  '/u01/app/oracle/oradata/sztech1/undotbs01.dbf',
  '/u01/app/oracle/oradata/sztech1/users01.dbf',
  '/u01/app/oracle/oradata/sztech1/example01.dbf',
  '/u01/app/oracle/oradata/sztech1/ts_inventory01.dbf'
CHARACTER SET ZHS16GBK
;
```

⑥执行步骤如下：

```
SQL> startup nomount
ORACLE instance started.
Total System Global Area 368263168 bytes
Fixed Size          1364704 bytes
Variable Size       322964768 bytes
Database Buffers    37748736 bytes
Redo Buffers        6184960 bytes
```

```
SQL> CREATE CONTROLFILE REUSE DATABASE "SZTECH1" NORESETLOGS ARCHIVELOG
      MAXLOGFILES 16
      MAXLOGMEMBERS 3
      MAXDATAFILES 100
      MAXINSTANCES 8
      MAXLOGHISTORY 292
LOGFILE
  GROUP 1 '/u01/app/oracle/oradata/sztech1/redo01.log' SIZE 50M BLOCKSIZE 512,
  GROUP 2 '/u01/app/oracle/oradata/sztech1/redo02.log' SIZE 50M BLOCKSIZE 512,
  GROUP 3 '/u01/app/oracle/oradata/sztech1/redo03.log' SIZE 50M BLOCKSIZE 512
-- STANDBY LOGFILE
DATAFILE
  '/u01/app/oracle/oradata/sztech1/system01.dbf',
  '/u01/app/oracle/oradata/sztech1/sysaux01.dbf',
  '/u01/app/oracle/oradata/sztech1/undotbs01.dbf',
  '/u01/app/oracle/oradata/sztech1/users01.dbf',
  '/u01/app/oracle/oradata/sztech1/example01.dbf',
  '/u01/app/oracle/oradata/sztech1/ts_inventory01.dbf'
CHARACTER SET ZHS16GBK;
Control file created.
```

```
SQL> recover database;
Media recovery complete.
```

下面这一步是由于原来就设置了数据库块改变跟踪：

```
SQL> ALTER DATABASE ENABLE BLOCK CHANGE TRACKING USING FILE
      '/home/oracle/bc.dat' REUSE;
Database altered.
```

```
SQL> ALTER SYSTEM ARCHIVE LOG ALL;
System altered.
```

```
SQL> ALTER DATABASE OPEN;
Database altered.
```