

```
SQL> select name,status from v$datafile
2 /
NAME                                STATUS
-----
C:\ORACLE\ORADATA\ORA9I\SYSTEM01.DBF  SYSTEM
C:\ORACLE\ORADATA\ORA9I\UNDOTBS01.DBF  ONLINE
C:\ORACLE\ORADATA\ORA9I\USERS01.DBF    ONLINE
C:\ORACLE\ORADATA\ORA9I\USERS02.DBF    ONLINE
SQL> select count(*) from eygle;
COUNT(*)
-----
12646
```

以下是告警日志文件中的信息输出：

```
Wed Jan 11 17:06:05 2012
alter database open
ORA-1113 signalled during: alter database open...
Wed Jan 11 17:06:10 2012
ALTER DATABASE RECOVER datafile 4
Media Recovery Start
Wed Jan 11 17:06:10 2012
Recovery of Online Redo Log: Thread 1 Group 3 Seq 78 Reading mem 0
  Mem# 0 errs 0: C:\ORACLE\ORADATA\ORA9I\REDO03.LOG
Media Recovery Complete
Completed: ALTER DATABASE RECOVER datafile 4
Wed Jan 11 17:06:13 2012
alter database open
Wed Jan 11 17:06:13 2012
LGWR: Primary database is in CLUSTER CONSISTENT mode
Wed Jan 11 17:06:13 2012
ARCH: Evaluating archive log 1 thread 1 sequence 76
ARCH: Beginning to archive log 1 thread 1 sequence 76
Creating archive destination LOG_ARCHIVE_DEST_1: 'C:\ORACLE\9.2.0\RDBMS\ARC00076.001'
ARCH: Completed archiving log 1 thread 1 sequence 76
```

```
Wed Jan 11 17:06:13 2012
LGWR: Primary database is in CLUSTER CONSISTENT mode
Thread 1 advanced to log sequence 79
Thread 1 opened at log sequence 79
  Current log# 1 seq# 79 mem# 0: C:\ORACLE\ORADATA\ORA9I\RED001.LOG
Successful open of redo thread 1
Wed Jan 11 17:06:13 2012
SMON: enabling cache recovery
Wed Jan 11 17:06:14 2012
Successfully onlined Undo Tablespace 1.
Dictionary check beginning
Tablespace 'TEMP' #2 found in data dictionary,
but not in the controlfile. Adding to controlfile.
Dictionary check complete
Wed Jan 11 17:06:14 2012
SMON: enabling tx recovery
Wed Jan 11 17:06:15 2012
Database Characterset is ZHS16GBK
replication_dependency_tracking turned off (no async multimaster replication found)
Completed: alter database open
```

注意这种方法是不得已情况下采取的非常手段，通常不被官方支持，如果跳过的日志中涵盖的大量事务，数据将不可避免的出现不一致的问题，一般建议用户通过导出重建数据库。

空间与文件离线

离线表空间加载修复

与前面两个案例极其相似，我在 2012 年又遇到了一则类似的案例，这一次是某银行用户，数据同样极其重要。

灾难描述

这是又一次由于文件离线保护而导致的数据灾难：

1. 用户的归档空间满，数据库崩溃
2. 用户删除了归档日志，释放空间
3. 用户恢复数据库运行
4. 发现有两个数据文件离线，数据量大约 40G
5. 由于归档日志删除，数据文件无法加载
6. 无备份，数据无法追补，灾难形成

这则故障与前面一则故障类似，恢复方法也是类似的。

案例警示

这个案例给我们的警示有：

1. **备份重于一切，备份是数据库应用的第一要义**

再次重申：唯一能让公司主管从夜里警醒的就是，数据备份。

如果每天晚上睡觉前，你想一想，哪个重要的业务数据库没有备份，假定当晚磁盘损坏，系统崩溃，文件丢失，你将如何应对？

我想每一个 IT 主管都无法在没有备份保障机制下安然入睡。

备份对于数据库来说重于一切。

2. 不要輕易的删除任何一个归档日志

哪怕是在空间紧张的情况下，也不要輕易删除任何一个归档日志文件，压缩或者转移会是临时释放空间的办法。

由于归档日志对于数据库的恢复不可缺少，所以，在你不能确定数据库完全无故障隐患之前，最好不要草率删除日志。我们已经遇到了众多与日志缺失有关的案例，所以我们严正的提醒大家：**不要輕易的删除任何一个归档日志。**

3. 日志监控与检查是防范问题的根本

有了备份的保障，再加上日常监控与检查才能够让数据库高枕无忧。

这里所说的监控和检查其实可以简化到 Oracle 数据库的告警日志（alert 日志）检查。Oracle 会将数据库的异常全部记录在告警日志文件当中，实时或准时时监控或检查告警日志，发现其中的问题，并做出处理和响应，是防范问题的基础与根本。

这个案例，如果用户能够检查日志，做好基本工作，那就可以防范灾难的出现，而如果有良好的备份，则灾难出现后也就不会手足无措。

技术回放

用户数据量大约 3T，是一个非常重要的核心数据库：

```
SYS@bank>select sum(bytes)/1024/1024 from v$datafile;  
SUM(BYTES)/1024/1024
```

```
-----
```

```
3415000
```

首先由于磁盘空间满，导致 Oracle 数据库对于文件的读写失败，有两个文件被离线，这是 Oracle 数据库的一种保护机制：

```
Fri Jan 13 19:22:21 2012
```

```
KCF: write/open error block=0xf1fa6 online=1
```

```
file=73 /dev/rbank_gdm05
```

```
error=27063 txt: 'IBM AIX RISC System/6000 Error: 22: Invalid argument
```

```

Additional information: -1
Additional information: 557056'
Automatic datafile offline due to write error on
file 73: /dev/rbank_gdm05
Fri Jan 13 19:22:53 2012
KCF: write/open error block=0xf91cd online=1
      file=74 /dev/rbank_gdm06
      error=27063 txt: 'IBM AIX RISC System/6000 Error: 5: I/O error
Additional information: -1
Additional information: 32768'
Automatic datafile offline due to write error on
file 74: /dev/rbank_gdm06
Fri Jan 13 19:22:53 2012
KCF: write/open error block=0xf9145 online=1
      file=74 /dev/rbank_gdm06
      error=27063 txt: 'IBM AIX RISC System/6000 Error: 22: Invalid argument
Additional information: -1
Additional information: 32768'
Automatic datafile offline due to write error on
file 74: /dev/rbank_gdm06

```

接下来控制文件的写操作失败,数据库崩溃,控制文件的故障表现在于队列无法获取,队列 CF 代表 Control File:

```

Fri Jan 13 19:38:12 2012
Errors in file /oracle/admin/bank/bdump/bank_arc1_438808.trc:
ORA-00494: enqueue [CF] held for too long (more than 900 seconds) by 'inst 1, osid
475952'
Fri Jan 13 19:38:14 2012
System State dumped to trace file /oracle/admin/bank/bdump/bank_arc1_438808.trc
Killing enqueue blocker (pid=475952) on resource CF-00000000-00000000
  by killing session 3302.1
Fri Jan 13 19:38:19 2012
Errors in file /oracle/admin/bank/bdump/bank_arc0_229686.trc:
ORA-00494: enqueue [CF] held for too long (more than 900 seconds) by 'inst 1, osid

```

```

475952'
Fri Jan 13 19:38:20 2012
System State dumped to trace file /oracle/admin/bank/bdump/bank_arc0_229686.trc
Killing enqueue blocker (pid=475952) on resource CF-00000000-00000000
  by killing session 3302.1
Fri Jan 13 19:43:17 2012
Errors in file /oracle/admin/bank/bdump/bank_arc1_438808.trc:
ORA-00494: enqueue [CF] held for too long (more than 900 seconds) by 'inst 1, osid
475952'
Fri Jan 13 19:43:19 2012
System State dumped to trace file /oracle/admin/bank/bdump/bank_arc1_438808.trc
Killing enqueue blocker (pid=475952) on resource CF-00000000-00000000
  by terminating the process
ARC1: terminating instance due to error 2103
Instance terminated by ARC1, pid = 438808

```

客户清理日志，释放空间后启动数据库：

```

Fri Jan 13 20:45:08 2012
ALTER DATABASE OPEN
Fri Jan 13 20:45:08 2012
Beginning crash recovery of 1 threads
  parallel recovery started with 15 processes
Fri Jan 13 20:45:09 2012
Started redo scan

```

注意此处，有 40479 个数据块需要恢复，这是相当多的内容：

```

Fri Jan 13 20:45:26 2012
Completed redo scan
  2561915 redo blocks read, 40479 data blocks need recovery

```

以下所有的日志组都需要用于恢复，起始的日志序号为 89206，起始的日志块号为 300488，以下显示了这些日志的恢复过程：

```

Fri Jan 13 20:45:54 2012
Started redo application at

```

```

Thread 1: logseq 89206, block 300488
Fri Jan 13 20:45:54 2012
Recovery of Online Redo Log: Thread 1 Group 1 Seq 89206 Reading mem 0
  Mem# 0: /dev/rbank_redo1_01
  Mem# 1: /dev/rbank_redo1_11
Fri Jan 13 20:46:03 2012
Recovery of Online Redo Log: Thread 1 Group 2 Seq 89207 Reading mem 0
  Mem# 0: /dev/rbank_redo1_02
  Mem# 1: /dev/rbank_redo1_12
Fri Jan 13 20:46:23 2012
Recovery of Online Redo Log: Thread 1 Group 3 Seq 89208 Reading mem 0
  Mem# 0: /dev/rbank_redo1_03
  Mem# 1: /dev/rbank_redo1_13
Fri Jan 13 20:46:41 2012
Completed redo application

```

完成恢复，读取了 2561915 个日志块，40447 个数据块写出。

```

Fri Jan 13 20:46:41 2012
Completed crash recovery at
Thread 1: logseq 89208, block 928298, scn 3386095894
40479 data blocks read, 40447 data blocks written, 2561915 redo blocks read

```

数据库启动之后，日志继续切换应用，注意这里的日志序列从 89209 开始：

```

Fri Jan 13 20:47:04 2012
Thread 1 advanced to log sequence 89209 (thread open)
Thread 1 opened at log sequence 89209
  Current log# 1 seq# 89209 mem# 0: /dev/rbank_redo1_01
  Current log# 1 seq# 89209 mem# 1: /dev/rbank_redo1_11
Successful open of redo thread 1
Fri Jan 13 20:47:05 2012
MTTR advisory is disabled because FAST_START_MTTR_TARGET is not set
Fri Jan 13 20:47:05 2012
SMON: enabling cache recovery
Fri Jan 13 20:47:07 2012

```

```

Successfully online Undo Tablespace 1.
Fri Jan 13 20:47:07 2012
SMON: enabling tx recovery
Fri Jan 13 20:47:07 2012
Database Characterset is ZHS16GBK
Opening with internal Resource Manager plan
where NUMA PG = 1, CPUs = 16
replication_dependency_tracking turned off (no async multimaster replication found)
Starting background process QMNC
QMNC started with pid=35, OS id=540710
Fri Jan 13 20:47:19 2012
db_recovery_file_dest_size of 2048 MB is 10.33% used. This is a
user-specified limit on the amount of space that will be used by this
database for recovery-related files, and does not reflect the amount of
space available in the underlying filesystem or ASM diskgroup.
Fri Jan 13 20:47:19 2012
Completed: ALTER DATABASE OPEN
Fri Jan 13 21:07:03 2012
ALTER SYSTEM ARCHIVE LOG
Fri Jan 13 21:07:03 2012
Thread 1 cannot allocate new log, sequence 89210
Private strand flush not complete
    Current log# 1 seq# 89209 mem# 0: /dev/rbank_redo1_01
    Current log# 1 seq# 89209 mem# 1: /dev/rbank_redo1_11
Fri Jan 13 21:07:05 2012
Thread 1 advanced to log sequence 89210 (LGWR switch)
    Current log# 2 seq# 89210 mem# 0: /dev/rbank_redo1_02
    Current log# 2 seq# 89210 mem# 1: /dev/rbank_redo1_12

```

注意这里提示闪回归档区 100%使用率，空间用完了：

```

Fri Jan 13 21:07:39 2012
Errors in file /oracle/admin/bank/udump/bank_ora_332386.trc:
ORA-19815: WARNING: db_recovery_file_dest_size of 2147483648 bytes is 100.00% used,
and has 0 remaining bytes available.

```

Fri Jan 13 21:07:39 2012

You have following choices to free up space from flash recovery area:

1. Consider changing RMAN RETENTION POLICY. If you are using Data Guard, then consider changing RMAN ARCHIVELOG DELETION POLICY.
2. Back up files to tertiary device such as tape using RMAN BACKUP RECOVERY AREA command.
3. Add disk space and increase db_recovery_file_dest_size parameter to reflect the new space.
4. Delete unnecessary files using RMAN DELETE command. If an operating system command was used to delete files, then use RMAN CROSSCHECK and DELETE EXPIRED commands.

Fri Jan 13 21:08:03 2012

ALTER SYSTEM ARCHIVE LOG

Fri Jan 13 21:08:03 2012

Thread 1 cannot allocate new log, sequence 89211

Private strand flush not complete

Current log# 2 seq# 89210 mem# 0: /dev/rbank_redo1_02

Current log# 2 seq# 89210 mem# 1: /dev/rbank_redo1_12

Fri Jan 13 21:08:05 2012

Thread 1 advanced to log sequence 89211 (LGWR switch)

Current log# 3 seq# 89211 mem# 0: /dev/rbank_redo1_03

Current log# 3 seq# 89211 mem# 1: /dev/rbank_redo1_13

Fri Jan 13 21:08:13 2012

Errors in file /oracle/admin/bank/udump/bank_ora_536702.trc:

ORA-19815: WARNING: db_recovery_file_dest_size of 2147483648 bytes is 100.00% used, and has 0 remaining bytes available.

Fri Jan 13 21:08:13 2012

You have following choices to free up space from flash recovery area:

1. Consider changing RMAN RETENTION POLICY. If you are using Data Guard, then consider changing RMAN ARCHIVELOG DELETION POLICY.

2. Back up files to tertiary device such as tape using RMAN
BACKUP RECOVERY AREA command.
3. Add disk space and increase db_recovery_file_dest_size parameter to
reflect the new space.
4. Delete unnecessary files using RMAN DELETE command. If an operating
system command was used to delete files, then use RMAN CROSSCHECK and
DELETE EXPIRED commands.

Fri Jan 13 21:12:02 2012

ALTER SYSTEM ARCHIVE LOG

Fri Jan 13 21:12:02 2012

Thread 1 cannot allocate new log, sequence 89212

Checkpoint not complete

Current log# 3 seq# 89211 mem# 0: /dev/rbank_redo1_03

Current log# 3 seq# 89211 mem# 1: /dev/rbank_redo1_13

Thread 1 cannot allocate new log, sequence 89212

Private strand flush not complete

Current log# 3 seq# 89211 mem# 0: /dev/rbank_redo1_03

Current log# 3 seq# 89211 mem# 1: /dev/rbank_redo1_13

Fri Jan 13 21:12:05 2012

Thread 1 advanced to log sequence 89212 (LGWR switch)

Current log# 1 seq# 89212 mem# 0: /dev/rbank_redo1_01

Current log# 1 seq# 89212 mem# 1: /dev/rbank_redo1_11

注意此时空间被释放出来

Fri Jan 13 21:12:07 2012

db_recovery_file_dest_size of 2048 MB is 0.06% used. This is a
user-specified limit on the amount of space that will be used by this
database for recovery-related files, and does not reflect the amount of
space available in the underlying filesystem or ASM diskgroup.

然后用户才发现文件离线，之后这两个文件无法 Online，故障出现了：

Fri Jan 13 21:23:01 2012

alter database recover datafile 73

```
Fri Jan 13 21:23:01 2012
Media Recovery Start
  parallel recovery started with 15 processes
ORA-279 signalled during:
alter database recover datafile 73
...
Fri Jan 13 21:28:13 2012
alter database datafile 73 online

Fri Jan 13 21:28:13 2012
ORA-1113 signalled during:
alter database datafile 73 online
...
Fri Jan 13 21:28:29 2012
alter database datafile 74 online
Fri Jan 13 21:28:30 2012
ORA-1113 signalled during:
alter database datafile 74 online
...
Fri Jan 13 21:28:41 2012
alter database recover datafile 74
Fri Jan 13 21:28:42 2012
Media Recovery Start
ORA-275 signalled during:
alter database recover datafile 74
...
Fri Jan 13 21:29:09 2012
alter database datafile 74 online
Fri Jan 13 21:29:09 2012
ORA-1113 signalled during:
alter database datafile 74 online
...
```

查询 v\$datafile 视图，可以看到这两个文件处于 RECOVER 状态：

```
SYS@bank>select file#,status from v$datafile where status='RECOVER';
```

```
FILE# STATUS
-----
73 RECOVER
74 RECOVER
```

恢复过程

通过控制文件转储，可以开始分析一下文件的检查点等相关信息：

```
SYS@bank>alter database backup controlfile to trace;
```

```
Database altered.
```

```
SYS@bank>alter session set events 'immediate trace name file_hdrs level 12';
```

```
Session altered.
```

以下摘录三个文件的内容，这三个文件属于同一个表空间：

```
72 /dev/rbank_gdm04
73 /dev/rbank_gdm05
74 /dev/rbank_gdm06
```

以下是来自文件头和控制文件的信息：

```
DATA FILE #72:
```

```
(name #80) /dev/rbank_gdm04
```

```
creation size=640000 block size=32768 status=0xe head=80 tail=80 dup=1
```

```
tablespace 27, index=27 krfil=72 prev_file=71
```

```
unrecoverable scn: 0x0000.c4664adc 12/27/2011 11:48:51
```

控制文件中记录的文件检查点信息：

```
Checkpoint cnt:68648 scn: 0x0000.c9d832b3 01/13/2012 23:19:03
```

```
Stop scn: 0x0000.c9d832b3 01/13/2012 23:19:03
```

```
Creation Checkpointed at scn: 0x0000.50bf07ae 04/21/2010 09:45:20
```

```
thread:1 rba:(0x404b.e8de9.10)
```

```
enabled threads: 01000000 00000000 00000000 00000000 00000000 00000000
```

```

Offline scn: 0x0000.00000000 prev_range: 0
Online Checkpointed at scn: 0x0000.00000000
thread:0 rba:(0x0.0.0)
enabled threads: 00000000 00000000 00000000 00000000 00000000 00000000
Hot Backup end marker scn: 0x0000.00000000
aux_file is NOT DEFINED
V10 STYLE FILE HEADER:
    Compatibility Vsn = 169870080=0xa200300
    Db ID=2256689008=0x86825770, Db Name='BANK'
    Activation ID=0=0x0
    Control Seq=5695898=0x56e99a, File size=1600000=0x186a00
    File Number=72, Blksiz=32768, File Type=3 DATA
Tablespace #27 - BANK2 rel_fn:72
Creation at scn: 0x0000.50bf07ae 04/21/2010 09:45:20
Backup taken at scn: 0x0000.00000000 01/01/1988 00:00:00 thread:0
reset logs count:0x2997f230 scn: 0x0000.00000001 reset logs terminal rcv data:0x0
scn: 0x0000.00000000
prev reset logs count:0x0 scn: 0x0000.00000000 prev reset logs terminal rcv data:0x0
scn: 0x0000.00000000
recovered at 01/13/2012 20:45:09
status:0x0 root dba:0x00000000 chkpt cnt: 68648 ctl cnt:68647
begin-hot-backup file size: 0

```

数据文件头上记录的检查点 SCN，可以看到文件头与控制文件一致，这说明该文件一致。时间点是 23: 19: 03

```

Checkpointed at scn: 0x0000.c9d832b3 01/13/2012 23:19:03
thread:1 rba:(0x15c7c.1a4d8.10)
enabled threads: 01000000 00000000 00000000 00000000 00000000 00000000
Backup Checkpointed at scn: 0x0000.00000000
thread:0 rba:(0x0.0.0)
enabled threads: 00000000 00000000 00000000 00000000 00000000 00000000

```

```

DATA FILE #73:
(name #81) /dev/rbank_gdm05

```

```
creation size=640000 block size=32768 status=0x1c head=81 tail=81 dup=1
tablespace 27, index=27 krfil=73 prev_file=72
unrecoverable scn: 0x0000.c4664adf 12/27/2011 11:48:52
```

对于 73 号文件，其控制文件记录该文件的检查点停留在 19: 20: 21

```
Checkpoint cnt:68641 scn: 0x0000.c9d2c793 01/13/2012 19:20:21
Stop scn: 0x0000.c9d337f8 01/13/2012 19:22:52
Creation Checkpointed at scn: 0x0000.50bf08bb 04/21/2010 09:49:06
thread:1 rba:(0x404b.e910d.10)
enabled threads: 01000000 00000000 00000000 00000000 00000000 00000000
Offline scn: 0x0000.00000000 prev_range: 0
Online Checkpointed at scn: 0x0000.00000000
thread:0 rba:(0x0.0.0)
enabled threads: 00000000 00000000 00000000 00000000 00000000 00000000
Hot Backup end marker scn: 0x0000.00000000
aux_file is NOT DEFINED
V10 STYLE FILE HEADER:
    Compatibility Vsn = 169870080=0xa200300
    Db ID=2256689008=0x86825770, Db Name='BANK'
    Activation ID=0=0x0
    Control Seq=5695894=0x56e996, File size=1600000=0x186a00
    File Number=73, Blksiz=32768, File Type=3 DATA
Tablespace #27 - BANK2 rel_fn:73
Creation at scn: 0x0000.50bf08bb 04/21/2010 09:49:06
Backup taken at scn: 0x0000.00000000 01/01/1988 00:00:00 thread:0
reset logs count:0x2997f230 scn: 0x0000.00000001 reset logs terminal rcv data:0x0
scn: 0x0000.00000000
prev reset logs count:0x0 scn: 0x0000.00000000 prev reset logs terminal rcv data:0x0
scn: 0x0000.00000000
recovered at 01/13/2012 21:23:02
status:0x4 root dba:0x00000000 chkpt cnt: 68641 ctl cnt:68640
begin-hot-backup file size: 0
```

数据文件头记录的信息和控制文件一致，停留在 19:20:21。

```
Checkpointed at scn: 0x0000.c9d2c793 01/13/2012 19:20:21
```

接下来这里的 RBA 信息中止处，恢复该文件的日志文件序号是 89205 (15c75)，这正是数据库崩溃前最后一个日志序列号。

数据库启动之后恢复的序列从 89206 开始，这之后的恢复与 73 号文件无关，因为该文件已经离线。

```
thread:1 rba:(0x15c75.2.10)
enabled threads: 01000000 00000000 00000000 00000000 00000000 00000000
Backup Checkpointed at scn: 0x0000.00000000
thread:0 rba:(0x0.0.0)
enabled threads: 00000000 00000000 00000000 00000000 00000000 00000000
```

```
DATA FILE #74:
```

```
(name #82) /dev/rbank_gdm06
creation size=640000 block size=32768 status=0x1c head=82 tail=82 dup=1
tablespace 27, index=27 krfil=74 prev_file=73
unrecoverable scn: 0x0000.c4664ae2 12/27/2011 11:48:52
```

74 号文件和 73 号文件相同，检查点停留在数据库崩溃之前

```
Checkpoint cnt:68641 scn: 0x0000.c9d2c793 01/13/2012 19:20:21
Stop scn: 0x0000.c9d3b918 01/13/2012 20:47:04
Creation Checkpointed at scn: 0x0000.50bf0a62 04/21/2010 09:52:55
thread:1 rba:(0x404b.e9549.10)
enabled threads: 01000000 00000000 00000000 00000000 00000000 00000000
Offline scn: 0x0000.00000000 prev_range: 0
Online Checkpointed at scn: 0x0000.00000000
thread:0 rba:(0x0.0.0)
enabled threads: 00000000 00000000 00000000 00000000 00000000 00000000
Hot Backup end marker scn: 0x0000.00000000
aux_file is NOT DEFINED
V10 STYLE FILE HEADER:
Compatibility Vsn = 169870080=0xa200300
Db ID=2256689008=0x86825770, Db Name='BANK'
Activation ID=0=0x0
Control Seq=5695583=0x56e85f, File size=1600000=0x186a00
```

```

File Number=74, Blksiz=32768, File Type=3 DATA
Tablespace #27 - BANK2 rel_fn:74
Creation at scn: 0x0000.50bf0a62 04/21/2010 09:52:55
Backup taken at scn: 0x0000.00000000 01/01/1988 00:00:00 thread:0
reset logs count:0x2997f230 scn: 0x0000.00000001 reset logs terminal rcv data:0x0
scn: 0x0000.00000000
prev reset logs count:0x0 scn: 0x0000.00000000 prev reset logs terminal rcv data:0x0
scn: 0x0000.00000000
recovered at 01/01/1988 00:00:00
status:0x4 root dba:0x00000000 chkpt cnt: 68641 ctl cnt:68640
begin-hot-backup file size: 0

```

文件头记录的信息和之前一致:

```

Checkpointed at scn: 0x0000.c9d2c793 01/13/2012 19:20:21
thread:1 rba:(0x15c75.2.10)
enabled threads: 01000000 00000000 00000000 00000000 00000000 00000000
Backup Checkpointed at scn: 0x0000.00000000
thread:0 rba:(0x0.0.0)
enabled threads: 00000000 00000000 00000000 00000000 00000000 00000000

```

这种情况我们只能尽量挽救数据，由于文件离线，归档日志丢失，无法进行恢复，只能通过修改数据文件头信息的方式，使数据库跳过缺失的文件，强制加载数据文件。

通常如果恢复之后出现错误，则需要导出数据，重建数据库；如果幸运，则不会出现任何错误，数据库可以继续运行。

最简单的操作，我们可以将 72 号文件的头块复制到 73 号和 74 号文件，修改相应的文件号之后，可以打开数据库，其修改过程与上一则案例相同：

```

copy file 72 block 1 to file 73 block 1;
copy file 72 block 1 to file 74 block 1;

```

○○○○

之后重建控制文件后，即可打开数据库：

```

SYS@bank>startup nomount;
ORACLE instance started.

```

```

Total System Global Area 1.8874E+10 bytes
Fixed Size                2117456 bytes
Variable Size             704643248 bytes
Database Buffers          1.8153E+10 bytes
Redo Buffers              14659584 bytes
SYS@bank>@cr
Control file created.
SYS@bank>recover datafile 73;
ORA-00283: recovery session canceled due to errors
ORA-00264: no recovery required
SYS@bank>recover datafile 74;
ORA-00283: recovery session canceled due to errors
ORA-00264: no recovery required
SYS@bank>alter database open;
Database altered.
SYS@bank>select file#,status from v$datafile where status='RECOVER';
0 rows selected.

```

查询数据文件上的数据，能够成功返回结果：

```

SYS@bank>col segment_name for a40
SYS@bank>select owner,segment_name,segment_type
           2 from dba_extents where file_id=73 and rownum < 2;

```

OWNER	SEGMENT_NAME	SEGMENT_TYPE
BODANK	F_CB_KDA_AUTD_H	TABLE

```

SYS@bank>select count(*) from BODANK.F_CB_KDA_AUTD_H;
COUNT(*)
-----
1141005

```

增加临时表空间：

```

SYS@bank>ALTER TABLESPACE TEMP ADD TEMPFILE '/dev/rbank_temp02' REUSE;
Tablespace altered.
SYS@bank>ALTER TABLESPACE TEMP ADD TEMPFILE '/dev/rbank_temp01' REUSE;

```

Tablespace altered.

这样就完美的完成了这则数据灾难的恢复工作，这次故障恢复之所以顺利，是因为用户在之后快速的发现并处理，及时发现问题是控制灾难影响的关键。

技术提示

关于归档空间的设置

我们注意到很多用户在设置了归档模式之后，却没有针对性的归档备份清理机制，也没有良好的空间规划，结果归档模式为用户带来了麻烦和灾难。

虽然归档模式是为了数据安全而设定的，但是如果用户无法掌握关于归档的维护，则这一设定是得不偿失的。

尤其在 Oracle 10g 引入了闪回恢复区之后，db_recovery_file_dest_size 的设置让很多客户遭受了惨痛的教训。

在本案例中，客户的存储空间并非不足，但是缺省的 db_recovery_file_dest_size 设置仅有 2G，客户未能主动去修改和调整这个参数：

```
db_recovery_file_dest    = /oracle/flash_recovery_area
db_recovery_file_dest_size= 2147483648
```

以下是告警日志中的信息摘录，这 2G 的归档空间在高峰期可能会很快耗尽：

```
db_recovery_file_dest_size= 2147483648
db_recovery_file_dest_size of 2048 MB is 10.33% used. This is a
ORA-19815: WARNING: db_recovery_file_dest_size of 2147483648 bytes is 100.00% used,
and has 0 remaining bytes available.
```

而且可能更为折磨的是，如果仅只通过手工删除归档日志，Oracle 并不认可这种空间释放，必须通过 Oracle 的手段清理，才能让数据库认识到这种变化。

以下是一个案例，当 db_recovery_file_dest_size 100% 用尽时，告警日志中会出现以下处理提示：

```
ORA-19815: WARNING: db_recovery_file_dest_size of 2147483648 bytes is 100.00% used,
and has 0 remaining bytes available.
```

```
*****
```

You have the following choices to free up space from
flash recovery area:

1. Consider changing your RMAN retention policy.
If you are using dataguard, then consider changing your
RMAN archivelog deletion policy.
2. Backup files to tertiary device such as tape using the
RMAN command BACKUP RECOVERY AREA.
3. Add disk space and increase the db_recovery_file_dest_size
parameter to reflect the new space.
4. Delete unnecessary files using the RMAN DELETE command.
If an OS command was used to delete files, then use
RMAN CROSSCHECK and DELETE EXPIRED commands.

```
*****
```

此时，手工释放空间，甚至切换闪回区到一个全新的磁盘，仍然无法解决问题，在数据库内部记录的信息不会变更：

```
SYS AS SYSDBA>set linesize 120
```

```
SYS AS SYSDBA>SELECT substr(name, 1, 30) name, space_limit AS quota,
```

```
2      space_used      AS used,
```

```
3      space_reclaimable AS reclaimable,
```

```
4      number_of_files  AS files
```

```
5  FROM  v$recovery_file_dest;
```

NAME	QUOTA	USED	RECLAIMABLE	FILES
-----	-----	-----	-----	-----
/data5/flash_recovery_area	2147483648	2144863232	0	227

可以看到，闪回区仍然记录了 227 个文件，USED 空间并未释放，释放的方式，可以通过 RMAN 进行 crosscheck，删除过期或废弃的文件：

```
RMAN> crosscheck archivelog all;
```

```
using target database controlfile instead of recovery catalog
```

```
allocated channel: ORA_DISK_1
```

```
channel ORA_DISK_1: sid=144 devtype=DISK
```

```
validation failed for archived log
```

```

archive log filename=/opt/oracle/flash_recovery_area/EYGLE/
archivelog/2004_05_17/o1_mf_1_790_0bjq36ps_.arc recid=1 stamp=526401126
validation failed for archived log
archive log filename=/opt/oracle/flash_recovery_area/EYGLE/
archivelog/2004_05_17/o1_mf_1_791_0bkbcy7x_.arc recid=2 stamp=526420862
validation failed for archived log
archive log filename=/opt/oracle/flash_recovery_area/EYGLE/
archivelog/2004_05_17/o1_mf_1_792_0bklds4d_.arc recid=3 stamp=526428057
.....
archive log filename=/opt/oracle/flash_recovery_area/EYGLE/
archivelog/2004_07_16/o1_mf_1_1014_0hh3zsrp_.arc recid=225 stamp=531678074
validation failed for archived log
archive log filename=/opt/oracle/flash_recovery_area/EYGLE/
archivelog/2004_07_16/o1_mf_1_1015_0hh40qyp_.arc recid=226 stamp=531678104
validation failed for archived log
archive log filename=/opt/oracle/flash_recovery_area/EYGLE/
archivelog/2004_07_16/o1_mf_1_1016_0hh41jqj_.arc recid=227 stamp=531678129
Crosschecked 227 objects
RMAN> delete noprompt expired archivelog all;
deleted archive log
archive log filename=/opt/oracle/flash_recovery_area/EYGLE/
archivelog/2004_05_17/o1_mf_1_790_0bjq36ps_.arc recid=1 stamp=526401126
deleted archive log
archive log filename=/opt/oracle/flash_recovery_area/EYGLE/
archivelog/2004_05_17/o1_mf_1_791_0bkbcy7x_.arc recid=2 stamp=526420862
deleted archive log
.....
archive log filename=/opt/oracle/flash_recovery_area/EYGLE/
archivelog/2004_07_16/o1_mf_1_1014_0hh3zsrp_.arc recid=225 stamp=531678074
deleted archive log
archive log filename=/opt/oracle/flash_recovery_area/EYGLE/
archivelog/2004_07_16/o1_mf_1_1015_0hh40qyp_.arc recid=226 stamp=531678104
deleted archive log

```

```
archive log filename=/opt/oracle/flash_recovery_area/EYGLE/
archivelog/2004_07_16/o1_mf_1_1016_0hh41jqj.arc recid=227 stamp=531678129
Deleted 227 EXPIRED objects
```

此时空间才会得以释放:

```
SYS AS SYSDBA on 28-MAR-05 >SELECT substr(name, 1, 30) name, space_limit AS quota,
2         space_used      AS used,
3         space_reclaimable AS reclaimable,
4         number_of_files  AS files
5   FROM   v$recovery_file_dest;

NAME                                QUOTA      USED RECLAIMABLE   FILES
-----
/data5/flash_recovery_area          2147483648  9959424         0         1
```

如果磁盘空间充足，最快速的解决方案是通过命令扩展闪回区的空间设定，以下是一个案例处理过程的参数修改:

```
SQL> show parameter recov

NAME                                TYPE        VALUE
-----
db_recovery_file_dest               string       /msflsh
db_recovery_file_dest_size           big integer  50G
recovery_parallelism                 integer      0

SQL> alter system set db_recovery_file_dest_size=65G scope=both;
System altered

SQL> show parameter recov

NAME                                TYPE        VALUE
-----
db_recovery_file_dest               string       /msflsh
db_recovery_file_dest_size           big integer  65G
recovery_parallelism                 integer      0
```

此外，如果归档日志可以全部清除，不需备份，则可以通过类似如下方式的指定日期直接删除:

```
RMAN> DELETE NOPROMPT ARCHIVELOG UNTIL TIME 'SYSDATE-2';
using target database control file instead of recovery catalog
```

allocated channel: ORA_DISK_1

channel ORA_DISK_1: SID=27 device type=DISK

List of Archived Log Copies for database with db_unique_name ORCL11G

=====

Key	Thrd	Seq	S	Low	Time
7	1	101	A	19-JAN-12	
				Name:	/archivelog/2012_01_19/o1_mf_1_101_7khsv8o5_.arc
8	1	102	A	19-JAN-12	
				Name:	/archivelog/2012_01_19/o1_mf_1_102_7khsv9rr_.arc
9	1	103	A	19-JAN-12	
				Name:	/archivelog/2012_01_19/o1_mf_1_103_7khs82o_.arc
10	1	104	A	19-JAN-12	
				Name:	/archivelog/2012_01_19/o1_mf_1_104_7kht29ns_.arc
11	1	105	A	19-JAN-12	
				Name:	/archivelog/2012_01_21/o1_mf_1_105_7kmvyxlx_.arc
12	1	106	A	21-JAN-12	
				Name:	/archivelog/2012_01_23/o1_mf_1_106_7krdp0qd_.arc

deleted archived log

archived log file name=/2012_01_19/o1_mf_1_101_7khsv8o5_.arc RECID=7 STAMP=772998632

deleted archived log

archived log file name=/2012_01_19/o1_mf_1_102_7khsv9rr_.arc RECID=8 STAMP=772998633

deleted archived log

archived log file name=/2012_01_19/o1_mf_1_103_7khs82o_.arc RECID=9 STAMP=772998664

deleted archived log

archived log file name=/2012_01_19/o1_mf_1_104_7kht29ns_.arc RECID=10 STAMP=772998857

deleted archived log

archived log file name=/2012_01_21/o1_mf_1_105_7kmvyxlx_.arc RECID=11 STAMP=773131877

deleted archived log

archived log file name=/2012_01_23/o1_mf_1_106_7krdp0qd_.arc RECID=12 STAMP=773280072

Deleted 11 objects

关于检查点的一致性调整

文件离线，如果恢复在线状态，主要是 SCN 与检查点以及 RBA 在发挥作用，如果归档日志不可用，则通过 BBED 方式跳过日志是可以尝试的。

直接修改文件检查点和 RBA 信息，也可以达到同样的效果，以下是一个测试说明，供参考。

首先将文件离线：

```
SQL> col name for a50
```

```
SQL> select file#,name from v$datafile;
```

```
FILE# NAME
```

```
-----
1 /u01/app/oracle/oradata/ORCL/system01.dbf
2 /u01/app/oracle/oradata/ORCL/undotbs01.dbf
3 /u01/app/oracle/oradata/ORCL/sysaux01.dbf
4 /u01/app/oracle/oradata/ORCL/users01.dbf
5 /u01/app/oracle/oradata/ORCL/marven01.dbf
6 /u01/app/oracle/oradata/ORCL/xxx.dbf
7 /u01/app/oracle/oradata/ORCL/eygle01.dbf
```

```
7 rows selected.
```

```
SQL> alter database datafile 7 offline drop;
```

```
Database altered.
```

该文件状态变更为 RECOVER：

```
SQL> select file#,status from v$datafile;
```

```
FILE# STATUS
```

```
-----
1 SYSTEM
2 ONLINE
3 ONLINE
4 ONLINE
5 ONLINE
6 ONLINE
```

7 RECOVER

7 rows selected.

切换日志，覆盖就的 REDO 日志，当再次执行恢复时，需要的日志将不可用：

```
SQL> alter system switch logfile;
```

```
System altered.
```

```
SQL> /
```

```
System altered.
```

```
SQL> /
```

```
System altered.
```

```
SQL> /
```

```
System altered.
```

```
SQL> recover datafile 7;
```

```
ORA-00279: change 4282927 generated at 01/16/2012 16:43:36 needed for thread 1
```

```
ORA-00289: suggestion :
```

```
/u01/app/oracle/flash_recovery_area/ORCL/archivelog/2012_01_16/o1_mf_1_204_%u_.a  
rc
```

```
ORA-00280: change 4282927 for thread 1 is in sequence #204
```

```
Specify log: {<RET>=suggested | filename | AUTO | CANCEL}
```

```
auto
```

```
ORA-00308: cannot open archived log
```

```
'/u01/app/oracle/flash_recovery_area/ORCL/archivelog/2012_01_16/o1_mf_1_204_%u_.  
arc'
```

```
ORA-27037: unable to obtain file status
```

```
Linux-x86_64 Error: 2: No such file or directory
```

```
Additional information: 3
```

```
ORA-00308: cannot open archived log
```

```
'/u01/app/oracle/flash_recovery_area/ORCL/archivelog/2012_01_16/o1_mf_1_204_%u_.  
arc'
```