

```
DBVERIFY - Verification starting : FILE = system01.dbf
```

```
DBVERIFY - Verification complete
```

```
Total Pages Examined          : 48640
```

```
Total Pages Processed (Data)  : 26289
```

```
Total Pages Failing (Data)    : 0
```

```
Total Pages Processed (Index): 4211
```

```
Total Pages Failing (Index)  : 0
```

```
Total Pages Processed (Other): 15284
```

```
Total Pages Empty            : 2856
```

```
Total Pages Marked Corrupt    : 0
```

```
Total Pages Influx           : 0
```

为了简化恢复，抛弃了一些不必要的数据库文件：

```
SQL> archive log list;
```

```
Database log mode          Archive Mode
Automatic archival        Enabled
Archive destination        /opt/oracle/admin/MEDIA/arch_opt
Oldest online log sequence 19922
Next log sequence to archive 19925
Current log sequence       19925
```

```
SQL> alter database datafile '/opt/oracle/oradata/MEDIA/indx01.dbf' offline;
```

```
Database altered.
```

```
SQL> alter database datafile '/opt/oracle/oradata/MEDIA/drsys01.dbf' offline;
```

```
Database altered.
```

在不完全恢复尝试打开时，数据库会提示，这些文件在 Resetlogs 之后会被丢弃，我们先手工将文件 DROP：

```
SQL> alter database open resetlogs;
```

```
alter database open resetlogs
```

```
*
```

```
ERROR at line 1:
```

```
ORA-01245: offline file 6 will be lost if RESETLOGS is done
```

```
ORA-01110: data file 6: '/opt/oracle/oradata/MEDIA/indx01.dbf'
```

```
SQL> alter database datafile '/opt/oracle/oradata/MEDIA/indx01.dbf' offline drop;
Database altered.
```

对于从历史备份中恢复的文件，可以通过 BBED 从其他文件的 SCN 检查点进行同步，强制修正一致性，这可以参考上一章的做法。

由于控制文件和部分数据文件不符，所以尝试用较旧的控制文件是无法打开数据库的，数据库提示文件较控制文件新，这需要重建控制文件来解决：

```
SQL> alter database open resetlogs;
alter database open resetlogs
*
ERROR at line 1:
ORA-01248: file 33 was created in the future of incomplete recovery
ORA-01110: data file 33: '/opt/oracle/oradata/MEDIA/media_16.dbf'
```

重建控制文件之后，强制打开数据库，出现实例崩溃：

```
SQL> alter database open resetlogs;
alter database open resetlogs
*
ERROR at line 1:
ORA-03113: end-of-file on communication channel
```

但是此时告警日志已经显示了熟悉的信息。除了几个抛弃的文件之外，就是 4194 的回滚错误信息：

```
Sat Apr 10 23:33:49 2010
SMON: enabling cache recovery
Sat Apr 10 23:33:49 2010
Dictionary check beginning
Tablespace 'INDX' #5 found in data dictionary,
but not in the controlfile. Adding to controlfile.
Tablespace 'DRSYS' #6 found in data dictionary,
but not in the controlfile. Adding to controlfile.
File #6 found in data dictionary but not in controlfile.
Creating OFFLINE file 'MISSING00006' in the controlfile.
This file can no longer be recovered so it must be dropped.
File #7 found in data dictionary but not in controlfile.
```

```

Creating OFFLINE file 'MISSING00007' in the controlfile.
This file can no longer be recovered so it must be dropped.
File #22 found in data dictionary but not in controlfile.
Creating OFFLINE file 'MISSING00022' in the controlfile.
This file can no longer be recovered so it must be dropped.
Dictionary check complete
Sat Apr 10 23:33:49 2010
SMON: enabling tx recovery
Sat Apr 10 23:33:50 2010
Errors in file /opt/oracle/admin/MEDIA/bdump/media_smon_3695.trc:
ORA-00600: internal error code, arguments: [4194], [53], [14], [], [], [], [], []
Recovery of Online Redo Log: Thread 1 Group 7 Seq 1 Reading mem 0
   Mem# 0 errs 0: /opt/oracle/oradata/MEDIA/redo07.log
Sat Apr 10 23:33:51 2010
Errors in file /opt/oracle/admin/MEDIA/bdump/media_smon_3695.trc:
ORA-00600: internal error code, arguments: [4194], [53], [14], [], [], [], [], []
SMON: terminating instance due to error 600
Instance terminated by SMON, pid = 3695

```

这个错误通过重建 Undo 表空间就可以消除，这是一个 Oracle 8i 的数据库，我们在参数文件中设置了如下参数：

```

_allow_resetlogs_corruption=true
_corrupted_rollback_segments='RBS0','RBS1','RBS2','RBS3','RBS4','RBS5','RBS6','RBS7','RBS8','RBS9','RBS10','
RBS11','RBS12','RBS13','RBS14','RBS15','RBS16','RBS17','RBS18','RBS19','RBS20','RBS21','RBS22','RBS23','RBS24','
RBS25','RBS26','RBS27','RBS28','RBS29'
_offline_rollback_segments='RBS0','RBS1','RBS2','RBS3','RBS4','RBS5','RBS6','RBS7','RBS8','RBS9','RBS10','RBS11','
RBS12','RBS13','RBS14','RBS15','RBS16','RBS17','RBS18','RBS19','RBS20','RBS21','RBS22','RBS23','RBS24','RBS25','
RBS26','RBS27','RBS28','RBS29'

```

然后数据库久违的可以重新启动：

```

oracle@v880 ~/admin/MEDIA/bdump$ sqlplus "/ as sysdba"
SQL*Plus: Release 8.1.7.0.0 - Production on Sat Apr 10 23:38:16 2010
(c) Copyright 2000 Oracle Corporation. All rights reserved.

```

Connected to an idle instance.

SQL> startup

ORACLE instance started.

Total System Global Area 9873338044 bytes

Fixed Size 102076 bytes

Variable Size 915955712 bytes

Database Buffers 8955625472 bytes

Redo Buffers 1654784 bytes

Database mounted.

Database opened.

SQL> select name from v\$datafile;

NAME

```
-----  
/opt/oracle/oradata/MEDIA/system01.dbf  
/opt/oracle/oradata/MEDIA/tools01.dbf  
/opt/oracle/oradata/MEDIA/rbs01.dbf  
/opt/oracle/oradata/MEDIA/temp01.dbf  
/opt/oracle/oradata/MEDIA/users01.dbf  
/opt/oracle/product/8.1.7.4/dbs/MISSING00006  
/opt/oracle/product/8.1.7.4/dbs/MISSING00007  
/opt/oracle/oradata/MEDIA/media_01.dbf  
/opt/oracle/oradata/MEDIA/media_idx_01.dbf  
/opt/oracle/oradata/MEDIA/media_02.dbf  
/opt/oracle/oradata/MEDIA/media_03.dbf  
.....
```

然后删除重建回滚表空间:

SQL> drop tablespace RBS including contents;

Tablespace dropped.

SQL> CREATE TABLESPACE rbs

2 DATAFILE '/opt/oracle/oradata/MEDIA/rbs201.dbf' SIZE 500M REUSE AUTOEXTEND ON

3 NEXT 100M MAXSIZE 4096M;

Tablespace created.

再然后数据库可以被成功打开:

```
SQL> startup
```

```
ORACLE instance started.
```

```
Total System Global Area 9873338044 bytes
```

```
Fixed Size                  102076 bytes
```

```
Variable Size              915955712 bytes
```

```
Database Buffers          8955625472 bytes
```

```
Redo Buffers               1654784 bytes
```

```
Database mounted.
```

```
Database opened.
```

```
SQL> select owner,count(*) from dba_tables group by owner;
```

```
OWNER                                COUNT(*)
-----
AURORA$JIS$UTILITY$                  14
CTXSYS                                31
MDSYS                                  15
MED                                     4
MEDIA                                 391
MID                                    1
ORDSYS                                9
OSE$HTTP$ADMIN                        3
OUTLN                                  2
PEPSI                                  19
PERFSTAT                              28
SYS                                    203
SYSTEM                                52
```

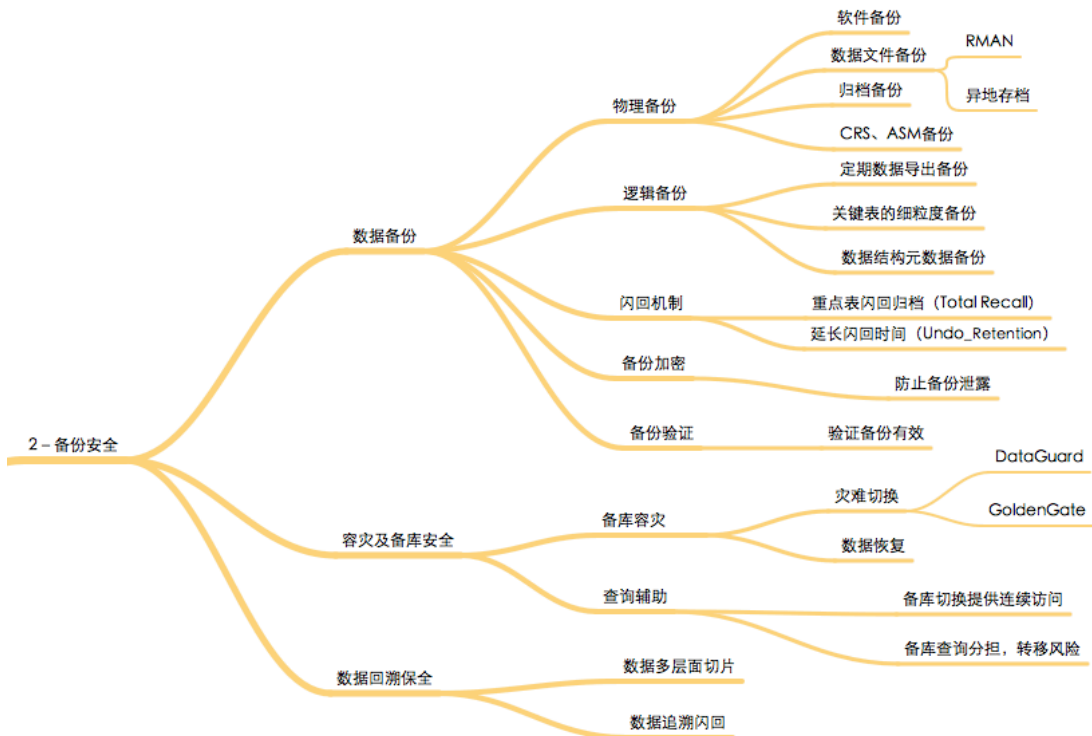
```
13 rows selected.
```

在这种情况下，我们建议用户通过导出数据，重建数据库。

千里之堤，溃于蚁穴

千丈之堤，溃于蚁穴，以蝼蚁之穴溃；百尺之室，以突隙之烟焚。

——先秦·韩非《韩非子·喻老》



有时候，防守坚固的数据库，可能因为一些微小的误操作、误处理，就走上了毁灭之路，这期间，技术往往是一个次要的环节，缺少规范、管理不善、盲目试错，最终导致了灾难的不可挽回。

本篇内容涉及管理安全方面的变更与上线管理，通过明确的变更规范和流程规范，本篇的几个数据灾难就可以避免：

以下的两个案例都是因为微小失误导致的多米诺效应，这些微小的失误最终使数据库陷入灾难。

一个字符引发的灾难

大小写字符疏忽导致的维护故障

以下是一个字符引发的灾难，不要小看任何一个字符：

灾难描述

用户是这样描述这次灾难的：

1. 进行删除表空间重建的维护，删除某表空间
2. 重建表空间后，发现创建有问题，再次尝试删除
3. 发现无法删除，提示表空间不存在，再创建提示裸设备已经使用
4. 离线、在线操作表空间数据库没问题
5. 多次操作后，出现 ORA-600 错误
6. 数据库无法正常运行，灾难形成

这个案例在《Oracle DBA 手记 2》中曾经有更为详细的描述，在本书中，我们仅仅作为一个警示收录在这里。

案例警示

经过分析处理后我们了解了案情的来龙去脉，问题的根源在于一个本应该大写的 K，被写成了小写 k，一个字符最后导致了这场数据库的大灾难。

这个案例给我们的警示有：

1. 不以规矩不成方圆，规范是技术的保障与补充。

在进行数据库维护，编写脚本时，要注重规范，统一编码，避免不必要的麻烦和风险，有时候规范比技术能力本身更重要。这些脚本必须经过测试才能够被应用到生产环境，而且，如果是非常重要的维护操作，至少申请一位审核人员对代码进行进一步的检查确认。

有的规范要靠企业和客户，有的规范来自于 DBA 自身，一个 DBA 应该养成自己的良好习惯和规范，避免让自己陷入不可预期的困境。

2. 不要在常规任务中处理超出把握的异常

如果你在工作中遇到了类似的情况，当数据库的表征超出了你的预期，那么最好停下来，仔细检查或咨询他人，避免在生产数据库上进行无把握的试错和尝试。机会喜欢光顾有准备的头脑，灾难喜欢降临在无把握的尝试。

不要在常规的工作任务中，处理毫无把握、超出预期的异常，你的尝试很有可能使得情况越来越糟糕。

3. 事先定制明确的回退方案

我们主张在重要的变更之前，准备好回退方案，如果没有，那么至少在你进行无把握的操作前，想一想如果出现不可预期的情况，你应当如何回退。

如果你的脑海中有回退的概念，我想很多灾难就不会发生，或者你根本就不会去走向不可预期。

在这个案例中，不规范的操作是导致灾难的根源。

案情解析

最初收到这个案例之后，我们首先分析告警日志文件，理清事件的来龙去脉。根据用户提供的信息，故障是因为表空间维护导致的，最初出现问题的是一个名称叫做"EMIS116K"表空间。

表空间的创建与删除

在告警日志中，这个表空间最初于 Apr 27 创建：

```
Tue Apr 27 09:05:27 2010
CREATE TABLESPACE "EMIS116K" DATAFILE '/dev/vg02/rlvdata224_8G'
SIZE 7900M AUTOEXTEND ON NEXT 23M MAXSIZE UNLIMITED LOGGING
EXTENT MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT AUTO
Tue Apr 27 09:11:05 2010
Completed: CREATE TABLESPACE "EMIS116K" DATAFILE '/dev/vg02/rlvdata224_8G'
```

这个表空间在 Jun 24 被删除，注意这里用户发出的名称“EMIS116k”，最后是一个小写的字母，这个小写的 k 会被数据库强制转换成大写，所以这个删除命令能够成功执行。但是这一个字母的微小差异，最终造成了数据库无法运行的灾难：

```
Thu Jun 24 16:37:53 2010
DROP TABLESPACE EMIS116k INCLUDING CONTENTS AND DATAFILES
Thu Jun 24 16:47:48 2010
Deleted file /dev/vg02/rlvdata224_8G
```

再然后，一个新的表空间在该文件上建立，表空间名称与原来“相同”。注意这里的相同是加了引号的，用户在这里指定的表空间名称是"EMIS116k"，这与原来的"EMIS116K"是不同的。大小写在引号之中，会存在不同的含义，引号强制了这个小写字母现在创建出来的表空间名称包含了最后一个小写字母 k：

```
Thu Jun 24 16:49:01 2010
CREATE TABLESPACE "EMIS116k" DATAFILE '/dev/vg02/rlvdata224_8G'
SIZE 7900M AUTOEXTEND ON NEXT 20M MAXSIZE UNLIMITED LOGGING
EXTENT MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT AUTO
Thu Jun 24 16:53:39 2010
Completed: CREATE TABLESPACE "EMIS116k" DATAFILE '/dev/vg02/rlvdata224_8G'
SIZE 7900M AUTOEXTEND ON NEXT 20M MAXSIZE UNLIMITED LOGGING
EXTENT MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT AUTO
```

随后用户向这个表空间创建对象，无法成功，就试图去删除这个表空间，这里仍然给出的是 EMIS116k，这个拼写经过 Oracle 的强制转换，就变成了 EMIS116K,这个名字的表空间是不存在的，于是出现了 ORA-959 错误：

```
Thu Jun 24 16:57:19 2010
DROP TABLESPACE EMIS116k INCLUDING CONTENTS AND DATAFILES
Thu Jun 24 16:57:19 2010
ORA-959 signalled during: DROP TABLESPACE EMIS116k INCLUDING CONTENTS AND DATAFILES
...
```

ORA-959 错误的含义是指定的表空间不存在：

```
ORA-00959 tablespace 'string' does not exist
Cause: A statement specified the name of a tablespace that does not exist.
Action: Enter the name of an existing tablespace. For a list of tablespace names,
query the data dictionary.
```

由于前面的操作报出的提示是表空间不存在，客户认为该表空间没有创建成功，或是已经被删除，于是接下来继续在同样的裸设备上创建新的表空间，这一次的名称是全部大写的"EMIS116K"：

```
Thu Jun 24 19:04:12 2010
CREATE TABLESPACE "EMIS116K" DATAFILE '/dev/vg02/rlvdata224_8G'
SIZE 7900M AUTOEXTEND ON NEXT 20M MAXSIZE UNLIMITED LOGGING
EXTENT MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT AUTO
```

由于上一次创建的表空间仍然存在，在同一个裸设备上创建表空间必然不会成功，这一次出现的是ORA-1537号错误：

```
Thu Jun 24 19:04:12 2010
ORA-1537 signalled during:
CREATE TABLESPACE "EMIS116K" DATAFILE '/dev/vg02/rlvdata224_8G'
SIZE 7900M AUTOEXTEND ON NEXT 20M MAXSIZE UNLIMITED LOGGING
EXTENT MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT AUTO
...
```

ORA-01537 的错误是指，指定的文件已经是数据库的一部分了，不能再被使用：

```
ORA-01537 cannot add data file 'string' - file already part of database
Cause: During CREATE or ALTER TABLESPACE, a file being added is already part of the
database.
Action: Use a different file name.
```

客户并没有认识到这样的错误，紧跟着一个创建语句再次被发出，此次的表空间名称又变成了大小写混合的"EMIS116k"：

```
Thu Jun 24 19:04:28 2010
CREATE TABLESPACE "EMIS116k" DATAFILE '/dev/vg02/rlvdata224_8G'
SIZE 7900M AUTOEXTEND ON NEXT 20M MAXSIZE UNLIMITED LOGGING
EXTENT MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT AUTO
```

这显然也不会成功，这次的错误提示是ORA-1543：

```
Thu Jun 24 19:04:28 2010
ORA-1543 signalled during:
CREATE TABLESPACE "EMIS116k" DATAFILE '/dev/vg02/rlvdata224_8G'
SIZE 7900M AUTOEXTEND ON NEXT 20M MAXSIZE UNLIMITED LOGGING
```

```
EXTENT MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT AUTO
...
```

ORA-01543 错误是指，要创建的表空间已经存在：

```
ORA-01543 tablespace 'string' already exists
```

```
Cause: An attempt was made to create a tablespace which already exists.
```

```
Action: Use a different name for the new tablespace.
```

好了，到这里打住。

我在此再次提醒大家的一点是：如果你在工作中遇到了类似的情况，当数据库的表征超出了你的预期，那么最好停下来，仔细检查或咨询他人，避免在生产数据库上进行无把握的试错和尝试。

这次的事情还要继续，接下来两个 DROP 命令被发出，但是 EMIS116K 和 EMIS116k 都不被数据库认可，两次尝试又失败了：

```
Thu Jun 24 19:05:09 2010
DROP TABLESPACE EMIS116K INCLUDING CONTENTS AND DATAFILES
Thu Jun 24 19:05:09 2010
ORA-959 signalled during: DROP TABLESPACE EMIS116K INCLUDING CONTENTS AND DATAFILES
...
Thu Jun 24 19:05:31 2010
DROP TABLESPACE EMIS116k INCLUDING CONTENTS AND DATAFILES
ORA-959 signalled during: DROP TABLESPACE EMIS116k INCLUDING CONTENTS AND DATAFILES
...
```

数据文件的离线与在线

用户开始尝试将该文件离线再 Online，试图通过这样的操作消除这个不明原因的无法 DROP 问题。这个尝试过程又引入了恢复的环节，还好一切顺利，表空间的数据文件能够被重新 Online，这样看起来数据文件是没有问题的：

```
Thu Jun 24 19:20:32 2010
alter database datafile '/dev/vg02/rlvdata224_8G' offline
Thu Jun 24 19:20:33 2010
Completed: alter database datafile '/dev/vg02/rlvdata224_8G' offline
```

```
Thu Jun 24 19:21:02 2010
alter database datafile '/dev/vg02/rlvdata224_8G' online
Thu Jun 24 19:21:02 2010
ORA-1113 signalled during: alter database datafile '/dev/vg02/rlvdata224_8G'
online...
Thu Jun 24 19:21:31 2010
ALTER DATABASE RECOVER datafile '/dev/vg02/rlvdata224_8G'
Thu Jun 24 19:21:31 2010
Media Recovery Start
  parallel recovery started with 15 processes
Thu Jun 24 19:21:35 2010
Media Recovery Complete (cwgk1)
Completed: ALTER DATABASE RECOVER datafile '/dev/vg02/rlvdata224_8G'
Thu Jun 24 19:21:45 2010
alter database datafile '/dev/vg02/rlvdata224_8G' online
Thu Jun 24 19:21:45 2010
Completed: alter database datafile '/dev/vg02/rlvdata224_8G' online
```

再回过头来继续尝试 DROP 表空间的操作，在没找到根本原因之前，这样的操作当然又失败了：

```
Thu Jun 24 19:48:18 2010
DROP TABLESPACE EMIS116k INCLUDING CONTENTS AND DATAFILES
Thu Jun 24 19:48:18 2010
ORA-959 signalled during: DROP TABLESPACE EMIS116k INCLUDING CONTENTS AND DATAFILES
...
Thu Jun 24 19:48:54 2010
DROP TABLESPACE EMIS116k INCLUDING CONTENTS AND DATAFILES
Thu Jun 24 19:48:54 2010
ORA-959 signalled during: DROP TABLESPACE EMIS116k INCLUDING CONTENTS AND DATAFILES
...
```

终于客户发现了大小写的问题，尝试将这个小写的 EMIS116k 更改为大写的 EMIS116K 名称。但是注意，不加入双引号，Oracle 认为这两者是没有区别的：

```
Thu Jun 24 19:52:52 2010
ALTER TABLESPACE EMIS116k RENAME TO EMIS116K
```

```
Thu Jun 24 19:52:52 2010
ORA-710 signalled during: ALTER TABLESPACE EMIS116k RENAME TO EMIS116K
...
```

ORA-710 是说这两个名字是一样的，无须更改。

```
ORA-00710 new tablespace name is the same as the old tablespace name
Cause : An attempt to rename a tablespace failed because the new name is the same as
the old name.
Action: No action required.
```

用户继续 DROP，仍然是 ORA-959。

```
Thu Jun 24 20:00:04 2010
DROP TABLESPACE EMIS116k
Thu Jun 24 20:00:04 2010
ORA-959 signalled during: DROP TABLESPACE EMIS116k
...
```

看着这样的过程，我们是否会有很着急的感觉？我相信在现场时，你可能会是焦急！

ORA-00600 4348 错误引发故障

终于用户发现了本质问题，发出了带着双引号的删除命令，这本身是没有问题的，可是恰恰此时数据库出了问题，一个异常的 ORA-00600 错误出现了，4348 出现：

```
Thu Jun 24 20:03:59 2010
DROP TABLESPACE "EMIS116k"
Thu Jun 24 20:03:59 2010
Errors in file /oracle/admin/cwgk/udump/cwgk1_ora_25919.trc:
ORA-00600: internal error code, arguments: [4348], [U], [0], [229], [], [], [], []
```

再然后，案情基本成立，任何 DROP 命令都提示表空间不存在了：

```
Thu Jun 24 20:04:12 2010
DROP TABLESPACE "EMIS116k"
Thu Jun 24 20:04:13 2010
ORA-959 signalled during: DROP TABLESPACE "EMIS116k"
```

```

...
Thu Jun 24 20:04:30 2010
DROP TABLESPACE "EMIS116k"
Thu Jun 24 20:04:30 2010
ORA-959 signalled during: DROP TABLESPACE "EMIS116k"
...
Thu Jun 24 20:04:40 2010
DROP TABLESPACE "EMIS116n"
Thu Jun 24 20:04:40 2010
ORA-959 signalled during: DROP TABLESPACE "EMIS116n"
...

```

接下来用户继续尝试创建其他的表空间，此时 ORA-00600 25013 错误出现：

```

Fri Jun 25 09:14:00 2010
CREATE TABLESPACE "EMIS116N2" DATAFILE '/dev/vg02/rlvdata088_16G'
SIZE 7900M AUTOEXTEND ON NEXT 20M MAXSIZE UNLIMITED LOGGING
EXTENT MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT AUTO
Fri Jun 25 09:18:40 2010
Errors in file /oracle/admin/cwgg/udump/cwggk1_ora_20031.trc:
ORA-00600: internal error code, arguments: [25013], [0], [229], [EMIS116N2],
[EMIS116k], [184], [179], []
Fri Jun 25 09:18:45 2010
Errors in file /oracle/admin/cwgg/bdump/cwggk1_pmon_4050.trc:
ORA-00600: internal error code, arguments: [25015], [229], [179], [184], [], [], [],
[]
Fri Jun 25 09:18:47 2010
Trace dumping is performing id=[cdmp_20100625091847]
Fri Jun 25 09:18:47 2010
Errors in file /oracle/admin/cwgg/bdump/cwggk1_pmon_4050.trc:
ORA-00600: internal error code, arguments: [kccocx_01], [], [], [], [], [], [], []
ORA-00600: internal error code, arguments: [25015], [229], [179], [184], [], [], [],
[]

```

伴随这些错误，有一个新的提示出现，Oracle 实例进行事务恢复时遇到 600 错误，这已经给我们一个暗示，

数据库内部事务出现了一致性的问题:

```
Fri Jun 25 09:19:01 2010
ORACLE Instance cwgk1 (pid = 30) - Error 600 encountered while recovering transaction
(203, 10).
```

此后, 后台进程异常终止, 数据库挂起, 最后被迫 Abort 关闭了数据库:

```
Fri Jun 25 09:24:06 2010
Errors in file /oracle/admin/cwgk/bdump/cwgk1_pmon_4050.trc:
ORA-00474: SMON process terminated with error
Fri Jun 25 09:24:06 2010
PMON: terminating instance due to error 474
Fri Jun 25 09:24:06 2010
Errors in file /oracle/admin/cwgk/bdump/cwgk1_lms3_4071.trc:
ORA-00474: SMON process terminated with error
Fri Jun 25 09:24:06 2010
Errors in file /oracle/admin/cwgk/bdump/cwgk1_lms1_4067.trc:
ORA-00474: SMON process terminated with error
Fri Jun 25 09:24:06 2010
Errors in file /oracle/admin/cwgk/bdump/cwgk1_lms0_4065.trc:
ORA-00474: SMON process terminated with error
Fri Jun 25 09:24:06 2010
Errors in file /oracle/admin/cwgk/bdump/cwgk1_lms2_4069.trc:
ORA-00474: SMON process terminated with error
Fri Jun 25 09:24:08 2010
Shutting down instance (abort)
License high water mark = 265
Fri Jun 25 09:24:12 2010
Instance terminated by PMON, pid = 4050
Fri Jun 25 09:24:13 2010
Instance terminated by USER, pid = 24022
```

接下来用户执行了一系列的恢复尝试, 包括重建控制文件, 强制 Resetlogs 打开数据库等, 最终导致情况失控。

我们回过头来分析一下最初的错误出现时机，这里用户执行了正确的语句，但是数据库却不再配合，报出了 ORA-00600 4348 错误，这个错误号在 Metalink 上也没有收录，错误原因未知：

```
Thu Jun 24 20:03:59 2010
DROP TABLESPACE "EMIS116k"
Thu Jun 24 20:03:59 2010
Errors in file /oracle/admin/cwgk/udump/cwgk1_ora_25919.trc:
ORA-00600: internal error code, arguments: [4348], [U], [0], [229], [], [], [], []
```

但是结合后面的出错信息：

```
Fri Jun 25 09:19:01 2010
ORACLE Instance cwgk1 (pid = 30) - Error 600 encountered while recovering transaction
(203, 10).
```

我们猜测，DROP TABLESPACE 实质上执行了后台递归操作，但是这个 DDL 事务因为异常，在失败后没有完成回滚，造成了数据字典不一致。

技术回放

要解决这个数据库故障，首先要分析 ORA-00600 4348 错误是如何出现的。

明白了所有的内部操作过程，我们又可以继续分析 ORA-00600 4348 号错误的成因了：

```
ORA-00600: internal error code, arguments: [4348], [U], [0], [229], [], [], [], []
```

我说过，猜测是一种重要的学习能力，现在我们就运用这种能力了。

通过猜测推演问题

这个错误的 4384 后的第一个代码是 U，应该是 Update 的缩写，更新的应该就是 file\$, ts\$ 字典表。这里的 229 应该就是更新的表空间号，Update 出错的可能就是被更新的记录不存在了。

那么更新的记录为什么会不存在？

这个我猜测可能是某个 DBA 考虑到是这个表空间的问题，手工执行了对于字典表的 Delete 操作，这就导致了后面的异常；当然也有可能是 Oracle 数据库自身丢失了这条记录（这种可能性不大）。

通过实验验证问题

可以验证一下这个猜测，首先通过手工删除一个表空间信息。当然我们需要严正提示：这些操作仅可在测试环境，或者说属于你自己的测试环境上进行，测试之前请先备份你的数据库。

删除可以通过 dba_tablespace 和 ts\$ 表进行，我们选择了对 ts\$ 表直接操作：

```
SQL> delete from dba_tablespace where tablespace_name='ENMO';
```

已删除 1 行。

```
SQL> select ts#,name,online$ from ts$ ;
```

TS#	NAME	ONLINE\$
0	SYSTEM	1
1	UNDOTBS1	1
2	SYSAUX	1
3	TEMP	1
4	USERS	1
5	UNDOTBS2	3
6	EYGLE	1
7	EYGLe	3

已选择 8 行。

```
SQL> rollback;
```

回退已完成。

```
SQL> select ts#,name,online$ from ts$ ;
```

TS#	NAME	ONLINE\$
0	SYSTEM	1
1	UNDOTBS1	1
2	SYSAUX	1
3	TEMP	1
4	USERS	1
5	UNDOTBS2	3
6	EYGLE	1

```

7 EYGL e                                3
8 ENMO                                    1

```

已选择 9 行。

```
SQL> delete from ts$ where ts#=8;
```

已删除 1 行。

```
SQL> commit;
```

提交完成。

删除了字典表信息之后，如果此时再来执行 `drop tablespace` 的操作，即出现了 4348 错误，后面的参数 8 是指表空间号是 8，也就是说当试图去 Update 修改 8 号表空间的状态时出现了异常，也就是发现这个表空间的信息不存在了，在正常情况下，这种情况绝不应该发生，因为数据库根本不对 `ts$` 执行 DELETE 操作：

```
SQL> drop tablespace enmo;
```

```
drop tablespace enmo
```

```
*
```

第 1 行出现错误：

```
ORA-00600: 内部错误代码, 参数: [4348], [U], [0], [8], [], [], [], []
```

告警日志中也记录了错误信息，并且产生了一个跟踪文件，这里的 8 号表空间也就是 ENMO 表空间：

```
Mon Aug 09 15:58:10 2010
```

```
drop tablespace enmo
```

```
Mon Aug 09 15:58:10 2010
```

```
Errors in file d:\oracle\admin\enmo\udump\enmo_ora_4736.trc:
```

```
ORA-00600: 内部错误代码, 参数: [4348], [U], [0], [8], [], [], [], []
```

```
Mon Aug 09 15:58:11 2010
```

```
ORA-600 signalled during: drop tablespace enmo...
```

跟踪文件中也显示全部的相关信息，绑定变量等：

```
Cursor#3(04D60CA4) state=BOUND curiob=04D7B9B4
```

```
curflg=5 fl2=0 par=04D60C24 ses=34343FE4
```

```
sqltxt(3028CDA4)=
```

```
update ts$ set name=:2,online$=:3,content$=:4,undofile#=:5,undoblock#=:6,
```

```
blocksize=:7,dflmaxext=:8,dflinit=:9,dflincr=:10,dflxtpct=:11,dflminext=:12,dflminle
```

```

n=:13,owner#=:14,scnwrp=:15,scnbas=:16,pitrscnwrp=:17,pitrscnbas=:18,dflogging=:19,bi
tmapped=:20,inc#=:21,flags=:22,plugged=:23,spare1=:24,spare2=:25 where ts#=:1
  hash=29d4143c7a377b40366bdc581b9a48d1
  parent=2FED6560 maxchild=02 plk=31B316EC ppn=n
cursor instantiation=04D7B9B4 used=1281340689
child#0(3028CC60) pcs=2FED6764
  clk=31B31BBC ci=2FED5D00 pn=00000000 ctx=2F966490
kgsccflg=0 llk[04D7B9B8,04D7B9B8] idx=0
xscflg=a0100426 fl2=5000400 fl3=2218c fl4=0
sharing failure(s)=10
Bind bytecodes
  Opcode = 5   Bind Rpi Scalar Sql In (not out) Nocopy
.....
kkscoacd
Bind#0
  oacdty=01 mxl=32(04) mxlc=00 mal=00 scl=00 pre=00
  oacflg=18 fl2=0001 frm=01 csi=852 siz=32 off=0
  kxsbbbfp=30284552 bln=32 avl=04 flg=09
  value="ENMO"
Bind#1
  oacdty=02 mxl=22(22) mxlc=00 mal=00 scl=00 pre=00
  oacflg=08 fl2=0001 frm=00 csi=00 siz=24 off=0
  kxsbbbfp=04d734f4 bln=24 avl=02 flg=05
  value=2
Bind#2~ Bind#4 value=0
Bind#5
  oacdty=02 mxl=22(22) mxlc=00 mal=00 scl=00 pre=00
  oacflg=08 fl2=0001 frm=00 csi=00 siz=24 off=0
  kxsbbbfp=04d7345c bln=24 avl=03 flg=05
  value=8192
Bind#6
  oacdty=02 mxl=22(22) mxlc=00 mal=00 scl=00 pre=00
  oacflg=08 fl2=0001 frm=00 csi=00 siz=24 off=0

```

kxsbbbf=04d73438 bln=24 avl=06 flg=05
value=2147483645

Bind#7

oacdy=02 mxl=22(22) mxlc=00 mal=00 scl=00 pre=00
oacflg=08 fl2=0001 frm=00 csi=00 siz=24 off=0
kxsbbbf=04d73414 bln=24 avl=02 flg=05
value=8

Bind#8

oacdy=02 mxl=22(22) mxlc=00 mal=00 scl=00 pre=00
oacflg=08 fl2=0001 frm=00 csi=00 siz=24 off=0
kxsbbbf=04d733f0 bln=24 avl=03 flg=05
value=128

Bind#9

oacdy=02 mxl=22(22) mxlc=00 mal=00 scl=00 pre=00
oacflg=08 fl2=0001 frm=00 csi=00 siz=24 off=0
kxsbbbf=04d733cc bln=24 avl=01 flg=05
value=0

Bind#10

oacdy=02 mxl=22(22) mxlc=00 mal=00 scl=00 pre=00
oacflg=08 fl2=0001 frm=00 csi=00 siz=24 off=0
kxsbbbf=04d733a8 bln=24 avl=02 flg=05
value=1

Bind#11

oacdy=02 mxl=22(22) mxlc=00 mal=00 scl=00 pre=00
oacflg=08 fl2=0001 frm=00 csi=00 siz=24 off=0
kxsbbbf=04d7337c bln=24 avl=02 flg=05
value=8

Bind#12~ Bind#15 value=0

Bind#16

oacdy=02 mxl=22(22) mxlc=00 mal=00 scl=00 pre=00
oacflg=08 fl2=0001 frm=00 csi=00 siz=24 off=0
kxsbbbf=04d7284c bln=24 avl=04 flg=05
value=907939

```

Bind#17
  oacdty=02 mxl=22(22) mxlc=00 mal=00 scl=00 pre=00
  oacflg=08 fl2=0001 frm=00 csi=00 siz=24 off=0
  kxsbbbfp=04d72828  bln=24  avl=02  flg=05
  value=1
Bind#18
  oacdty=02 mxl=22(22) mxlc=00 mal=00 scl=00 pre=00
  oacflg=08 fl2=0001 frm=00 csi=00 siz=24 off=0
  kxsbbbfp=04d72804  bln=24  avl=02  flg=05
  value=8
Bind#19
  oacdty=02 mxl=22(22) mxlc=00 mal=00 scl=00 pre=00
  oacflg=08 fl2=0001 frm=00 csi=00 siz=24 off=0
  kxsbbbfp=04d727e0  bln=24  avl=02  flg=05
  value=1
Bind#20
  oacdty=02 mxl=22(22) mxlc=00 mal=00 scl=00 pre=00
  oacflg=08 fl2=0001 frm=00 csi=00 siz=24 off=0
  kxsbbbfp=04d727bc  bln=24  avl=02  flg=05
  value=33
Bind#21~ Bind#23  value=0
Bind#24
  oacdty=02 mxl=22(22) mxlc=00 mal=00 scl=00 pre=00
  oacflg=08 fl2=0001 frm=00 csi=00 siz=24 off=0
  kxsbbbfp=04d73518  bln=22  avl=02  flg=05
  value=8
Frames pfr 04D713C0 siz=7188 efr 04D71338 siz=7172
Cursor frame dump
  enxt: 4.0x00000404  enxt: 3.0x000009b8  enxt: 2.0x000001f4  enxt: 1.0x00000c54
  pnxt: 2.0x00000004  pnxt: 1.0x0000000c
kxscphp  04D70064  siz=2016  inu=0  nps=624
kxscbhp  04D66510  siz=2016  inu=0  nps=1432
-----

```

```

Cursor#1(04D60C24) state=BOUND curiob=04D74B38
  curflg=4c fl2=400 par=00000000 ses=34352EC4
  child cursor: 3
  sqltxt(04AAFC04)=drop tablespace enmo
  hash=00000000000000000000000000000000
  parent=04AA2870 maxchild=01 plk=04AAFD24 ppn=y
cursor instantiation=04D74B38 used=1281340689
  child#0(04AA2264) pcs=04AA2684
  clk=04AA2334 ci=04AA2010 pn=04AA23A4 ctx=04AA1814
  kgsccflg=0 llk[04D74B3C,04D74B3C] idx=0
  xscflg=c0802276 fl2=c000400 fl3=2202008 fl4=100
  Frames pfr 04D74AFC siz=240 efr 04D74AAC siz=232
  Cursor frame dump
  enxt: 1.0x000000e8
  pnxt: 1.0x00000008
  kxscephp 04D66020 siz=1000 inu=0 nps=140
  kxscefhp 04D704C0 siz=4072 inu=0 nps=244
-----

```

而如果此时我们尝试去重新创建 ENMO 表空间,则会出现 25013 / 25015 错误,客户的情况完全得以重演。这里的 25015 错误跟随着几个参数,其中 8 指表空间号,新创建的 ENMO 表空间被重新赋予了 8 号表空间,第二个参数 6 是指表空间记录号,5 是指数据文件号:

```

SQL> create tablespace enmo datafile size 10M;
create tablespace enmo datafile size 10M
*
第 1 行出现错误:
ORA-00603: ORACLE server session terminated by fatal error
ORA-00600: internal error code, arguments: [25015], [8], [6], [5], [], [], [], []
ORA-00600: internal error code, arguments: [25015], [8], [6], [5], [], [], [], []
ORA-00600: internal error code, arguments: [25013], [0], [8], [ENMO], [ENMO], [5],
[6], []
进程 ID: 0
会话 ID: 159 序列号: 14

```