

# 9i新特点：SPFILE的使用

盖国强(ITPUB: Eygle [eygle@itpub.net](mailto:eygle@itpub.net))

## 简介：

本文是关于 Oracle9i spfile 的一篇文章，这是 Oracle9i 的一个新特性，虽然是很小的一个改进，但是却体现了 Oracle 在安全性和可靠性上的一个提高，通过 rman 备份或者控制文件的自动备份，Oracle 把 spfile 纳入到整个备份恢复体系当中。

本文对 spfile 的使用和修改等作了详细描述，对于其备份恢复也给出了相应的例子，希望大家能够从这篇文章中受益，从 Oracle 的这个新特性中获得收益。

在 Oracle9i 中，一个新的文件被引入 -spfile, spfile 用于服务器端，管理初始化参数。

在 9i 以前，Oracle 使用 pfile 存储初始化参数设置，这些参数在实例启动时被读取，任何修改需要重起实例才能生效；使用 spfile 你可以使用 ALTER SYSTEM 或者 ALTER SESSION 来动态修改那些可动态修改的参数，所有更改可以立即生效，你可以选择使更改只应用于当前实例还是同时应用到 spfile。这就使得所有对 spfile 的修改都可以在命令行完成，我们可以彻底告别手工修改初始化参数文件，这就大大减少了人为错误的发生。

SPFILE 是一个二进制文件，可以使用 RMAN 进行备份，这样实际上 Oracle 把参数文件也纳入了备份恢复管理。

除了第一次启动数据库需要 PFILE（然后可以根据 PFILE 创建 SPFILE），我们可以不再需要 PFILE，ORACLE 强烈推荐使用 spfile，应用其新特性来存储和维护初始化参数设置。

## 一. 创建 SPFILE

缺省的，ORACLE 使用 PFILE 启动数据库，SPFILE 必须由 PFILE 创建，新创建的 SPFILE 在下一次启动数据库时生效，CREATE SPFILE 需要 SYSDBA 或者 SYSOPER 的权限：

语法如下：

```
CREATE SPFILE[='SPFILE-NAME'] FROM PFILE[='PFILE-NAME']
```

例：

```
SQL> create spfile from pfile;
```

缺省的，spfile 创建到系统缺省目录

(Unix: \$ORACLE\_HOME/dbs; NT: \$ORACLE\_HOME\database)

如果 SPFILE 已经存在，那么创建会返回以下错误：

```
SQL> create spfile from pfile;
create spfile from pfile
*
ERROR 位于第 1 行:
ORA-32002: 无法创建已由例程使用的 SPFILE
```

这也可以用来判断当前是否使用了 SPFILE 文件。

然而意外的时，Oracle 并没有向其他文件一样，在运行期间保持锁定，让我们作以下试验：

```
SQL> host rename SPFILEEYGLEN.ORA SPFILEEYGLEN.ORA.BAK

SQL> alter system set db_cache_size=24M scope=both;
系统已更改。

SQL> host dir *.ora
驱动器 E 中的卷是 Doc
卷的序列号是 980C-8EFF

E:\Oracle\Ora9iR2\database 的目录

2003-02-10 14:35                2,048 PWDeyglen.ORA
          1 个文件                2,048 字节
          0 个目录   150,347,776 可用字节

SQL> alter system set db_cache_size=24M scope=spfile;
alter system set db_cache_size=24M scope=spfile
*
ERROR 位于第 1 行:
ORA-27041: 无法打开文件
OSD-04002: 无法打开文件
O/S-Error: (OS 2) 系统找不到指定的文件。
```

```
SQL> host rename SPFILEEYGLEN.ORA.BAK SPFILEEYGLEN.ORA

SQL> alter system set db_cache_size=24M scope=spfile;
系统已更改。
SQL>
```

估计 Oracle 以后会想办法来锁定这个文件。

## 二. 使用 SPFILE

重新启动数据库,使用 startup 命令,Oracle 将会按照以下顺序在缺省目录中搜索参数文件:

- a. spfile\${ORACLE\_SID}.ora

缺省目录 UNIX: \${ORACLE\_HOME}/dbs/  
NT: \${ORACLE\_HOME}\database

- b. spfile.ora

缺省目录 UNIX: \${ORACLE\_HOME}/dbs/  
NT: \${ORACLE\_HOME}\database

- c. init\${ORACLE\_SID}.ora

缺省目录 UNIX: \${ORACLE\_HOME}/dbs/  
NT: \${ORACLE\_HOME}\database or  
\${ORACLE\_HOME}\admin\db\_name\pfile\

创建了 spfile,重新启动数据库,Oracle 会按顺序搜索以上目录,spfile 就会自动生效。

### 三. 使用 pfile/spfile 启动数据库

如果你想使用 pfile 启动数据库，你可以在启动时指定 pfile 或者删除 spfile.

```
SQL> startup pfile='E:\Oracle\admin\eyglen\pfile\init.ora';
```

你不能以同样的方式指定 spfile，但是可以创建一个包含 spfile 参数的 pfile 文件，指向 spfile.

SPFILE 是一个自 Oracle9i 引入的初始化参数，类似于 IFILE 参数。SPFILE 参数用于定义非缺省路径的 spfile 文件。

你可以在 PFILE 链接到 SPFILE 文件，同时在 PFILE 中定义其他参数，如果参数重复设置，后读取的参数将取代先前的设置。

PFILE 参数的使用，例如：

这是我们使用 SPFILE 启动的情况，

```
SQL> startup
ORACLE 例程已经启动。

Total System Global Area  135338868 bytes
Fixed Size                  453492 bytes
Variable Size              109051904 bytes
Database Buffers           25165824 bytes
Redo Buffers                667648 bytes
数据库装载完毕。
数据库已经打开。
SQL> show parameter log_archive_start

NAME                                TYPE
-----
VALUE
-----
log_archive_start                    boolean
TRUE
SQL> show parameter spfile

NAME                                TYPE
-----
VALUE
```

```
-----  
spfile                                string  
%ORACLE_HOME%\DATABASE\SPFILE%  
ORACLE_SID%.ORA  
SQL>
```

我们修改 PFILE 文件内容如下：

```
#Pfile link to SPFILE  
  
SPFILE= 'E:\Oracle\Ora9iR2\database\SPFILEEYGLEN.ORA'  
  
log_archive_start = false
```

可以预见这个 log\_archive\_start 参数设置将会代替 SPFILE 中的设置：

```
SQL> startup pfile='e:\initeyglen.ora'  
  
ORACLE 例程已经启动。  
  
Total System Global Area 135338868 bytes  
Fixed Size                453492 bytes  
Variable Size             109051904 bytes  
Database Buffers         25165824 bytes  
Redo Buffers              667648 bytes
```

数据库装载完毕。

数据库已经打开。

```
SQL> show parameter spfile
```

NAME	TYPE
-----	
VALUE	
-----	
spfile	string
E:\Oracle\Ora9iR2\database\SPF	
ILEEYGLEN.ORA	

```
SQL> show parameter log_archive_start
```

NAME	TYPE
------	------

```
-----  
VALUE
```

```
-----  
log_archive_start          boolean
```

```
FALSE
```

然后我们可以使用 ALTER SYSTEM 方式将修改固定到 SPFILE.

```
SQL> alter system set log_archive_start=false scope=spfile;
```

```
系统已更改。
```

所以你也可以通过如上方式在启动时修改初始化参数。比我们在本文最后介绍的导入导出方法要简便的多。

#### 四. 修改参数

可以通过 ALTER SYSTEM 或者导入导出更改 SPFILE 的内容。

ALTER SYSTEM 增加了一个新选项：SCOPE。SCOPE 参数有三个可选值：

MEMORY ,SPFILE , BOTH

MEMORY: 只改变当前实例运行

SPFILE: 只改变 SPFILE 的设置

BOTH: 改变实例及 SPFILE

##### 1. SCOPE=MEMORY

```
SQL> show parameter timed_statistics
```

NAME	TYPE
------	------

-----  
VALUE

-----  
timed\_statistics                   boolean

TRUE

```
SQL> ALTER SYSTEM SET timed_statistics=FALSE SCOPE=MEMORY;
```

系统已更改。

```
SQL> show parameter timed_statistics
```

NAME	TYPE
------	------

-----  
VALUE

-----  
timed\_statistics                   boolean

FALSE

```
SQL> shutdown immediate
```

数据库已经关闭。

已经卸载数据库。

ORACLE 例程已经关闭。

```
SQL> startup
```

ORACLE 例程已经启动。

Total System Global Area 135338868 bytes

Fixed Size 453492 bytes

```
Variable Size          109051904 bytes
Database Buffers      25165824 bytes
Redo Buffers          667648 bytes
```

数据库装载完毕。  
数据库已经打开。

```
SQL> show parameter timed_statistics
```

NAME	TYPE
-----	-----
VALUE	
-----	
timed_statistics	boolean
TRUE	

## 2. SCOPE=SPFILE

```
SQL> show parameter timed_statistics
```

NAME	TYPE
-----	-----
VALUE	
-----	
timed_statistics	boolean
TRUE	

```
SQL> ALTER SYSTEM SET timed_statistics=FALSE SCOPE=SPFILE;
```

系统已更改。

```
SQL> show parameter timed_statistics
```

NAME	TYPE
-----	-----
VALUE	
-----	
timed_statistics	boolean
TRUE	

```
SQL> shutdown immediate
```

数据库已经关闭。  
已经卸载数据库。

ORACLE 例程已经关闭。

```
SQL> startup
```

ORACLE 例程已经启动。

```
Total System Global Area 135338868 bytes
Fixed Size                  453492 bytes
```



```
Variable Size          109051904 bytes
Database Buffers      25165824 bytes
Redo Buffers          667648 bytes
```

数据库装载完毕。  
数据库已经打开。

```
SQL> show parameter timed_statistics
```

NAME	TYPE
-----	-----
VALUE	
-----	
timed_statistics	boolean
FALSE	

```
SQL>
```

### 3 . SCOPE = BOTH

使用 BOTH 选项实际上等同于不带参数的 ALTER SYSTEM 语句。

注意：如果修改静态参数，那么需要指定 SPFILE 参数，否则将会报错。

```
SQL> show parameter timed_statistics
```

NAME	TYPE
-----	-----
VALUE	
-----	
timed_statistics	boolean
FALSE	

```
SQL> ALTER SYSTEM SET timed_statistics=TRUE SCOPE=BOTH;
```

系统已更改。

```
SQL> show parameter timed_statistics
```

NAME	TYPE
-----	-----
VALUE	
-----	
timed_statistics	boolean

TRUE

SQL> shutdown immediate

数据库已经关闭。

已经卸载数据库。

ORACLE 例程已经关闭。

SQL> startup

ORACLE 例程已经启动。

Total System Global Area 135338868 bytes

Fixed Size 453492 bytes

Variable Size 109051904 bytes

Database Buffers 25165824 bytes

Redo Buffers 667648 bytes

数据库装载完毕。

数据库已经打开。

SQL> show parameter timed\_statistics

NAME	TYPE
------	------

-----  
VALUE  
-----

timed_statistics	boolean
------------------	---------

TRUE

SQL> ALTER SYSTEM SET sql\_trace=FALSE SCOPE=BOTH;

ALTER SYSTEM SET sql\_trace=FALSE SCOPE=BOTH

\*

ERROR 位于第 1 行:

ORA-02095: 无法修改指定的初始化参数

```
SQL> ALTER SYSTEM SET sql_trace=FALSE SCOPE=SPFILE;
```

系统已更改。

4 . 你也可以在数据库 shutdown 时创建和修改 spfile,例如 :

```
SQL> show sga
```

Total System Global Area 135338868 bytes

Fixed Size 453492 bytes

Variable Size 109051904 bytes

Database Buffers 25165824 bytes

Redo Buffers 667648 bytes

```
SQL> shutdown immediate
```

数据库已经关闭。

已经卸载数据库。

ORACLE 例程已经关闭。

```
SQL> create pfile from spfile;
```

文件已创建。

```
SQL> create spfile from pfile;
```

文件已创建。

```
SQL>
```

## 五. 是否使用了 spfile

判断是否使用了 SPFILE，可以使用以下方法：

1. 查询 v\$parameter 动态视图，如果以下查询返回空值，那么你在使用 pfile.

```
SQL> SELECT name,value FROM v$parameter WHERE name='spfile';

NAME
-----
VALUE
-----
spfile
%ORACLE_HOME%\DATABASE\SPFILE%ORACLE_SID%.ORA
```

2. 或者你可以使用 SHOW 命令来显示参数设置，如果以下结果 value 列返回空值，那么说明你在使用 pfile:

```
SQL> SHOW PARAMETER spfile

NAME                                TYPE
-----                                -----
VALUE
-----
spfile                                string
%ORACLE_HOME%\DATABASE\SPFILE%
ORACLE_SID%.ORA
```

3. 查询 v\$spparameter 视图

如果以下查询返回 0 值，表示你在使用 pfile, 否则表明你使用的是 spfile:

```
SQL> SELECT COUNT(*) FROM v$spparameter WHERE value IS NOT NULL;

COUNT(*)
```

或者使用以下查询，如果 true 值返回非 0 值，那么说明我们使用的是 spfile.

```
SQL> select isspecified, count(*) from v$spparameter group
  2  by isspecified;

ISSPECIFIED    COUNT(*)
-----
FALSE          226
TRUE           33
```

更为直接的：

```
SQL> select decode(count(*), 1, 'spfile', 'pfile' ) USED
  2  from v$spparameter
  3  where rownum=1 and isspecified='TRUE'
  4  /

USED
-----
spfile
```

## 六. SPFILE 的备份与恢复

在本文开篇我们提到，Oracle 把 Spfile 也纳入到 Rman 的备份恢复策略当中，如果你配置了控制文件自动备份(autoback)，那么 Oracle 会在数据库发生重大变化(如增减表空间)时自动进行控制文件及 Spfile 文件的备份。

下面我们来看一下这个过程：

### a. 设置控制文件自动备份：

```
[oracle@jumper oracle]$ rman target /

Recovery Manager: Release 9.2.0.3.0 - Production

Copyright (c) 1995, 2002, Oracle Corporation. All rights reserved.

connected to target database: HSJF (DBID=1052178311)

RMAN> CONFIGURE CONTROLFILE AUTOBACKUP ON;

using target database controlfile instead of recovery catalog
old RMAN configuration parameters:
CONFIGURE CONTROLFILE AUTOBACKUP OFF;
new RMAN configuration parameters:
CONFIGURE CONTROLFILE AUTOBACKUP ON;
new RMAN configuration parameters are successfully stored

RMAN> exit
```

这个设置可以在数据库中通过如下方式查询得到：

```
[oracle@jumper bdump]$ sqlplus "/ as sysdba"

SQL*Plus: Release 9.2.0.3.0 - Production on Sat Jan 17 01:08:05 2004

Copyright (c) 1982, 2002, Oracle Corporation. All rights reserved.

Connected to:
Oracle9i Enterprise Edition Release 9.2.0.3.0 - Production
With the Partitioning, OLAP and Oracle Data Mining options
JServer Release 9.2.0.3.0 - Production
```

```
SQL> select * from v$rman_configuration;
CONF# NAME VALUE
-----
1 CONTROLFILE AUTOBACKUP ON
```

#### b. 记录数据库变化

```
SQL> create tablespace eygle
2 datafile '/data1/oracle/oradata/eygle01.dbf'
3 size 5M;

Tablespace created.
```

如果新建一个表空间，这时候检查 alert<sid>.log 文件，你可以在其中发现这样的备份信息:

```
Sat Jan 17 00:55:57 2004
Starting control autobackup
Control autobackup written to DISK device
handle
'/opt/oracle/product/9.2.0/dbs/c-1052178311-20040117-00'
Completed: create tablespace eygle
datafile '/data1/oracle/oradata/eygle01.dbf'
```

如果使用 rman 进行备份，在提示中你可以看到如下信息:

```
RMAN> configure controlfile autobackup on;

old RMAN configuration parameters:
CONFIGURE CONTROLFILE AUTOBACKUP OFF;
new RMAN configuration parameters:
CONFIGURE CONTROLFILE AUTOBACKUP ON;
new RMAN configuration parameters are successfully stored

RMAN> run
2> {
3>     allocate channel ch1 type disk
format='e:\oracle\orabak\penny%t.arc';
4> backup archivelog all delete all input;
```

```

5> release channel ch1;
6> }

allocated channel: ch1
channel ch1: sid=13 devtype=DISK

Starting backup at 02-DEC-03
current log archived
channel ch1: starting archive log backupset
channel ch1: specifying archive log(s) in backup set
input archive log thread=1 sequence=63 recid=168 stamp=511712617
input archive log thread=1 sequence=64 recid=169 stamp=511712620
input archive log thread=1 sequence=65 recid=170 stamp=511712626
input archive log thread=1 sequence=66 recid=171 stamp=511712690
channel ch1: starting piece 1 at 02-DEC-03
channel ch1: finished piece 1 at 02-DEC-03
piece handle=E:\ORACLE\ORABAK\PENNY511712693.ARC comment=NONE
channel ch1: backup set complete, elapsed time: 00:00:03
channel ch1: deleting archive log(s)
archive    log    filename=E:\ORACLE\ORADATA\PENNY\ARCHIVE\1_63.DBF
recid=168 stamp=511712617
archive    log    filename=E:\ORACLE\ORADATA\PENNY\ARCHIVE\1_64.DBF
recid=169 stamp=511712620
archive    log    filename=E:\ORACLE\ORADATA\PENNY\ARCHIVE\1_65.DBF
recid=170 stamp=511712626
archive    log    filename=E:\ORACLE\ORADATA\PENNY\ARCHIVE\1_66.DBF
recid=171 stamp=511712690
Finished backup at 02-DEC-03

Starting Control File and SPFILE Autobackup at 02-DEC-03
piece      handle=E:\ORACLE\ORA92\DATABASE\C-3627775766-20031202-01
comment=NONE
Finished Control File and SPFILE Autobackup at 02-DEC-03

released channel: ch1

```

我们简单看一下自动备份的控制文件及 spfile 文件的格式及命名规则:

**c-#####-YYYYMMDD-QQ**

c -----控制文件

**#####**-----DBID

**YYYYMMDD**-----时间戳

**QQ**-----序号 00-FF , 16 进制表示



### c. 使用自动备份恢复 spfile 文件

```
[oracle@jumper bdump]$ rman target /  
  
Recovery Manager: Release 9.2.0.3.0 - Production  
  
Copyright (c) 1995, 2002, Oracle Corporation. All rights reserved.  
  
connected to target database: HSJF (DBID=1052178311)  
  
RMAN> restore spfile to '/tmp/spfileeygle.ora' from autobackup;  
  
Starting restore at 17-JAN-04  
  
using target database controlfile instead of recovery catalog  
allocated channel: ORA_DISK_1  
channel ORA_DISK_1: sid=18 devtype=DISK  
channel ORA_DISK_1: looking for autobackup on day: 20040117  
channel ORA_DISK_1: autobackup found: c-1052178311-20040117-01  
channel ORA_DISK_1: SPFILE restore from autobackup complete  
Finished restore at 17-JAN-04  
  
RMAN> exit  
  
Recovery Manager complete.  
[oracle@jumper bdump]$ ls -l /tmp/spfileeygle.ora  
-rw-r----- 1 oracle dba 3584 1月 17 09:34 /tmp/spfileeygle.ora
```

你同样可以通过这种方法恢复控制文件，示例如下：

```
[oracle@jumper bdump]$ rman target /  
  
Recovery Manager: Release 9.2.0.3.0 - Production  
  
Copyright (c) 1995, 2002, Oracle Corporation. All rights reserved.  
  
connected to target database: HSJF (DBID=1052178311)  
  
RMAN> restore controlfile to '/tmp/control01.ctl' from autobackup;
```

```

Starting restore at 17-JAN-04

using target database controlfile instead of recovery catalog
allocated channel: ORA_DISK_1
channel ORA_DISK_1: sid=10 devtype=DISK
channel ORA_DISK_1: looking for autobackup on day: 20040117
channel ORA_DISK_1: autobackup found: c-1052178311-20040117-02
channel ORA_DISK_1: controlfile restore from autobackup complete
Finished restore at 17-JAN-04

RMAN> exit

Recovery Manager complete.
[oracle@jumper bdump]$ ls -l /tmp/control*
-rw-r-----      1 oracle      dba              1892352    1 月  17  09:44
/tmp/control01.ctl

```

Oracle9i 自动备份控制文件的功能给我们带来了极大的收益,通过自动备份,在数据库出现紧急状况的时候,你可能可以从这个自动备份中获得更为有效及时的控制文件.

缺省的,这个自动备份功能是关闭的,你可以用我们上面提到的方法打开该功能.

## 七. 设置 Events 事件

Events 事件是 Oracle 的重要诊断工具及问题解决办法,很多时候需要通过 Events 设置来屏蔽或者更改 Oracle 的行为,下面我们来看一下怎样修改 spfile,增加 Events 事件设置:

```

SQL> alter system set event='10841 trace name context forever'
scope=spfile;

System altered.

SQL> startup force;
ORACLE instance started.

Total System Global Area 101782380 bytes
Fixed Size                451436 bytes

```

```

Variable Size          75497472 bytes
Database Buffers      25165824 bytes
Redo Buffers          667648 bytes
Database mounted.
Database opened.
SQL> show parameter event

NAME                                TYPE        VALUE
-----                                -
event                                string      10841 trace name
context forever

```

顺便提一句，10841 事件是用于解决 Oracle9i 中 JDBC Thin Driver 问题的一个方法，如果你的 alert.log 文件中出现以下错误提示：

```

Wed Jan  7 17:17:08 2004
Errors                                in                                file
/opt/oracle/admin/phsdb/udump/phsdb_ora_1775.trc:
ORA-00600: internal error code, arguments: [ttcgcshnd-1], [0], [],
[], [], [], [], []
Wed Jan  7 17:17:18 2004
Errors                                in                                file
/opt/oracle/admin/phsdb/udump/phsdb_ora_1777.trc:
ORA-00600: internal error code, arguments: [ttcgcshnd-1], [0], [],
[], [], [], [], []
Wed Jan  7 17:17:24 2004
Errors                                in                                file
/opt/oracle/admin/phsdb/udump/phsdb_ora_1783.trc:
ORA-00600: internal error code, arguments: [ttcgcshnd-1], [0], [],
[], [], [], [], []
Wed Jan  7 17:17:31 2004
Errors                                in                                file
/opt/oracle/admin/phsdb/udump/phsdb_ora_1785.trc:
ORA-00600: internal error code, arguments: [ttcgcshnd-1], [0], [],
[], [], [], [], []
Wed Jan  7 17:17:39 2004
Errors                                in                                file
/opt/oracle/admin/phsdb/udump/phsdb_ora_1777.trc:
ORA-00600: internal error code, arguments: [ttcgcshnd-1], [0], [],
[], [], [], [], []
Wed Jan  7 17:17:45 2004

```

```

Errors                                in                                file
/opt/oracle/admin/phsdb/udump/phsdb_ora_1783.trc:
ORA-00600: internal error code, arguments: [ttcgcshnd-1], [0], [],
[], [], [], [], []
Wed Jan  7 17:17:52 2004
Errors                                in                                file
/opt/oracle/admin/phsdb/udump/phsdb_ora_1787.trc:
ORA-00600: internal error code, arguments: [ttcgcshnd-1], [0], [],
[], [], [], [], []
Wed Jan  7 17:18:11 2004
Errors                                in                                file
/opt/oracle/admin/phsdb/udump/phsdb_ora_1791.trc:
ORA-00600: internal error code, arguments: [ttcgcshnd-1], [0], [],
[], [], [], [], []
Wed Jan  7 17:18:19 2004
Errors                                in                                file
/opt/oracle/admin/phsdb/udump/phsdb_ora_1785.trc:
ORA-00600: internal error code, arguments: [ttcgcshnd-1], [0], [],
[], [], [], [], []

```

那么，很不幸，你很可能是遇到了 bug: 1725012  
 通过设置以上事件，可以屏蔽和解决这个 ORA-00600 错误。  
 具体你可以参考 Metalink 相关文档。

## 八. 导出 SPFILE 文件

SPFILE 文件可以导出为文本文件，使用导出、创建过程你可以向 SPFILE 中添加参数。

```
SQL> create pfile='e:\initeyglen.ora' from spfile;
```

文件已创建。

```
SQL> shutdown immediate
```

数据库已经关闭。

已经卸载数据库。

ORACLE 例程已经关闭。

Initeyglen.ora 文件的内容

```
*.aq_tm_processes=1
```

```
*.background_dump_dest='e:\oracle\admin\eyglen\bdump'  
*.compatible='9.2.0.0.0'  
*.control_files='e:\oracle\oradata\eyglen\control01.ctl',  
                'e:\oracle\oradata\eyglen\control02.ctl',  
                'e:\oracle\oradata\eyglen\control03.ctl'  
*.core_dump_dest='e:\oracle\admin\eyglen\cdump'  
*.db_block_size=8192  
*.db_cache_size=25165824  
*.db_domain=''  
*.db_file_multiblock_read_count=16  
*.db_name='eyglen'  
*.dispatchers='(PROTOCOL=TCP) (SERVICE=eyglenXDB)'  
*.fast_start_mttr_target=300  
*.hash_join_enabled=TRUE  
*.instance_name='eyglen'  
*.java_pool_size=33554432  
*.job_queue_processes=10  
*.large_pool_size=8388608  
*.open_cursors=300  
*.pga_aggregate_target=25165824  
*.processes=150  
*.query_rewrite_enabled='FALSE'  
*.remote_login_passwordfile='EXCLUSIVE'  
*.shared_pool_size=50331648  
*.sort_area_size=524288  
*.sql_trace=FALSE  
*.star_transformation_enabled='FALSE'  
*.timed_statistics=TRUE  
*.undo_management='AUTO'  
*.undo_retention=10800  
*.undo_tablespace='UNDOTBS1'  
*.user_dump_dest='e:\oracle\admin\eyglen\udump'
```

然后我们可以使用这个 pfile，或者手动修改其中的参数以启动数据库。

我们修改这个 pfile，增加一行

```
*.log_archive_start=true
```

使用这个 PFILE 启动数据库

```
SQL> startup pfile='e:\initeyglen.ora'
```

ORACLE 例程已经启动。

```
Total System Global Area 135338868 bytes
```

```
Fixed Size 453492 bytes
```

```
Variable Size 109051904 bytes
```

```
Database Buffers 25165824 bytes
```

```
Redo Buffers 667648 bytes
```

数据库装载完毕。

数据库已经打开。

```
SQL> show parameter log_archive_start
```

NAME	TYPE
-----	-----
VALUE	
-----	
log_archive_start	boolean

TRUE

```
SQL>
```

然后我们可以使用新的 PFILE 创建 SPFILE

```
SQL> create spfile from pfile='e:\initeyglen.ora';
```

文件已创建。

重新启动数据库，新的 SPFILE 生效。

```
SQL> startup
```

ORACLE 例程已经启动。

```
Total System Global Area 135338868 bytes
```

```
Fixed Size 453492 bytes
```

```
Variable Size 109051904 bytes
```

```
Database Buffers 25165824 bytes
```

```
Redo Buffers 667648 bytes
```

数据库装载完毕。

数据库已经打开。

```
SQL> show parameter spfile
```

```
NAME                                TYPE
-----
VALUE
-----
spfile                               string
%ORACLE_HOME%\DATABASE\SPFILE%
ORACLE_SID%.ORA
SQL> show parameter log_archive_start

NAME                                TYPE
-----
VALUE
-----
log_archive_start                    boolean
TRUE
SQL>
```

## 九. 关于 920 的 PFILE

在 \$ORACLE\_BASE\admin\db\_name\spfile 下，你很可能可以看到一个类似这样 [init.ora.192003215317]名字的文件，这就是初始化参数文件，只是跟上了时间戳。

对于 Oracle920,缺省的就使用 spfile 启动，但是这个 spfile 不是凭空而来，而是根据这个文件创建而来，你可以去掉这个长后缀，就是标准的 pfile 文件了。

如果手动创建数据库，可以看到以下的过程：

```
SQL> create spfile='e:\oracle\ora10G\database\spfilesunny.ora'  
FROM pfile='E:\oracle\admin\sunny\scripts\init.ora';  
  
文件已创建。
```

如果你想要使用 pfile 启动数据库，那么你可以把 spfile 改名，NT 缺省位于 \$ORACLE\_HOME\database 下，这样 Oracle 就不会使用这个 spfile 了。

你使用 create pfile from spfile 命令创建的 pfile，也应该在这个目录下

以下是一个完整的例子：

```
E:\Oracle\Ora9iR2\database>dir *.ora  
驱动器 E 中的卷是 Doc  
卷的序列号是 980C-8EFF  
  
E:\Oracle\Ora9iR2\database 的目录  
  
2003-02-26  10:49                1,028 INITeyglen.ORA  
2003-02-10  14:35                2,048 PWDeyglen.ORA  
2003-02-26  11:05                3,584 SPFILEEYGLN.ORA  
                3 个文件                6,660 字节  
                0 个目录            937,455,616 可用字节  
  
E:\Oracle\Ora9iR2\database>sqlplus /nolog  
  
SQL*Plus: Release 9.2.0.1.0 - Production on 星期三 2月 26 11:16:29 2003  
  
Copyright (c) 1982, 2002, Oracle Corporation.  All rights reserved.  
  
SQL> connect / as sysdba  
已连接到空闲例程。  
SQL> startup  
ORACLE 例程已经启动。
```



```
Total System Global Area 135338868 bytes
Fixed Size                  453492 bytes
Variable Size               109051904 bytes
Database Buffers           25165824 bytes
Redo Buffers                667648 bytes
```

数据库装载完毕。

数据库已经打开。

```
SQL> show parameter spfile
```

NAME	TYPE
-----	-----
VALUE	
-----	

```
spfile                                string
```

```
%ORACLE_HOME%\DATABASE\SPFILE%
```

```
ORACLE_SID%.ORA
```

```
SQL> shutdown immediate
```

数据库已经关闭。

已经卸载数据库。

ORACLE 例程已经关闭。

```
SQL> host rename SPFILEEYGLEN.ORA SPFILEEYGLEN.ORA.bak
```

```
SQL> host dir *.ora
```

驱动器 E 中的卷是 Doc

卷的序列号是 980C-8EFF

E:\Oracle\Ora9iR2\database 的目录

2003-02-26 10:49	1,028	INITeyglen.ORA
2003-02-10 14:35	2,048	PWDeyglen.ORA
2 个文件	3,076	字节
0 个目录	937,435,136	可用字节

```
SQL> startup
```

ORACLE 例程已经启动。

```
Total System Global Area 135338868 bytes
Fixed Size                  453492 bytes
Variable Size               109051904 bytes
Database Buffers           25165824 bytes
Redo Buffers                667648 bytes
```

数据库装载完毕。

数据库已经打开。

```
SQL> show parameter spfile
```

NAME	TYPE
------	------

-----	-----
VALUE	
-----	

spfile	string
--------	--------

```
SQL>
```

```
SQL> host rename SPFILEEYGLEN.ORA.bak SPFILEEYGLEN.ORA
```

```
SQL> shutdown immediate
```

数据库已经关闭。

已经卸载数据库。

ORACLE 例程已经关闭。

```
SQL> startup
```

ORACLE 例程已经启动。

Total System Global Area 135338868 bytes

Fixed Size 453492 bytes

Variable Size 109051904 bytes

Database Buffers 25165824 bytes

Redo Buffers 667648 bytes

数据库装载完毕。

数据库已经打开。

```
SQL> show parameter spfile
```

NAME	TYPE
------	------

-----	-----
VALUE	
-----	

spfile	string
--------	--------

%ORACLE\_HOME%\DATABASE\SPFILE%

ORACLE\_SID%.ORA

```
SQL>
```

## 一〇. 案例诊断

这是实际生产系统中关于 spfile 的一个案例问题的解决。

情况说明:

系统：SUN Solaris8

数据库版本:9203

问题描述：工程人员报告，数据库在重新启动时无法正常启动.检查发现 UNDO 表空间丢失.

问题诊断及解决过程:

#### 1. 登陆系统检查 alert.log 文件

```
SunOS 5.8

login: root
Password:
Last login: Thu Apr  1 11:39:16 from 10.123.7.162
Sun Microsystems Inc.    SunOS 5.8          Generic Patch    October 2001
You have new mail.
# su - oracle
bash-2.03$ cd $ORACLE_BASE/admin/*/bdump
bash-2.03$ vi *.log

"alert_gzhs.log" 7438 lines, 283262 characters
Sat Feb  7 20:30:06 2004
Starting ORACLE instance (normal)
LICENSE_MAX_SESSION = 0
LICENSE_SESSIONS_WARNING = 0
SCN scheme 3
Using log_archive_dest parameter default value
LICENSE_MAX_USERS = 0
SYS auditing is disabled
Starting up ORACLE RDBMS Version: 9.2.0.3.0.
System parameters with non-default values:
  processes                = 150
  timed_statistics         = TRUE
  shared_pool_size        = 1157627904
  large_pool_size         = 16777216
  java_pool_size          = 637534208
  control_files            = /u01/oradata/gzhs/control01.ctl,
                           /u02/oradata/gzhs/control02.ctl,
                           /u03/oradata/gzhs/control03.ctl
  db_block_size           = 8192
  db_cache_size           = 2516582400
  compatible               = 9.2.0.0.0
  log_archive_start       = TRUE
  log_archive_dest_1      = LOCATION=/u06/oradata/gzhs/arch
  log_archive_format      = %t_%s.dbf
  db_file_multiblock_read_count= 16
  fast_start_mttr_target  = 300
  undo_management          = AUTO
```

```
undo_tablespace          = UNDOTBS1
undo_retention            = 10800
remote_login_passwordfile= EXCLUSIVE
db_domain                =
instance_name            = gzhs
dispatchers              = (PROTOCOL=TCP) (SERVICE=gzhsXDB)
job_queue_processes      = 10
hash_join_enabled        = TRUE
background_dump_dest     = /oracle/admin/gzhs/bdump
user_dump_dest           = /oracle/admin/gzhs/udump
core_dump_dest           = /oracle/admin/gzhs/cdump
sort_area_size           = 524288
db_name                  = gzhs
open_cursors              = 300
star_transformation_enabled= FALSE
query_rewrite_enabled    = FALSE
pga_aggregate_target     = 838860800
aq_tm_processes          = 1
PMON started with pid=2
DBW0 started with pid=3
LGWR started with pid=4
CKPT started with pid=5
SMON started with pid=6
"alert_gzhs.log" 7438 lines, 283262 characters
USER: terminating instance due to error 30012
Instance terminated by USER, pid = 26433
ORA-1092 signalled during: ALTER DATABASE OPEN...
Thu Apr  1 11:11:08 2004
Starting ORACLE instance (normal)
LICENSE_MAX_SESSION = 0
LICENSE_SESSIONS_WARNING = 0
SCN scheme 3
Using log_archive_dest parameter default value
LICENSE_MAX_USERS = 0
SYS auditing is disabled
Starting up ORACLE RDBMS Version: 9.2.0.3.0.
System parameters with non-default values:
  processes              = 150
  timed_statistics       = TRUE
  shared_pool_size       = 1157627904
  large_pool_size        = 16777216
  java_pool_size         = 637534208
  control_files          = /u01/oradata/gzhs/control01.ctl,
/u02/oradata/gzhs/control02.ctl, /u03/oradata/gzhs/control03.ctl
```

```

db_block_size           = 8192
db_cache_size           = 2516582400
compatible               = 9.2.0.0.0
log_archive_start       = TRUE
log_archive_dest_1      = LOCATION=/u06/oradata/gzhs/arch
log_archive_format      = %t_%s.dbf
db_file_multiblock_read_count= 16
fast_start_mttr_target  = 300
undo_management          = AUTO
undo_tablespace         = UNDOTBS1
undo_retention          = 10800
remote_login_passwordfile= EXCLUSIVE
db_domain               =
instance_name           = gzhs
dispatchers             = (PROTOCOL=TCP) (SERVICE=gzhsXDB)
job_queue_processes     = 10
hash_join_enabled       = TRUE
background_dump_dest    = /oracle/admin/gzhs/bdump
user_dump_dest          = /oracle/admin/gzhs/udump
core_dump_dest          = /oracle/admin/gzhs/cdump
sort_area_size          = 524288
db_name                 = gzhs
open_cursors            = 300
star_transformation_enabled= FALSE
query_rewrite_enabled   = FALSE
pga_aggregate_target    = 838860800
aq_tm_processes         = 1
PMON started with pid=2
DBW0 started with pid=3
LGWR started with pid=4
CKPT started with pid=5
SMON started with pid=6
RECO started with pid=7
CJQ0 started with pid=8
Thu Apr  1 11:11:13 2004
starting up 1 shared server(s) ...
QMN0 started with pid=9
Thu Apr  1 11:11:13 2004
starting          up          1          dispatcher(s)          for          network          address
'(ADDRESS=(PARTIAL=YES)(PROTOCOL=TCP))'...
ARCH: STARTING ARCH PROCESSES
ARC0 started with pid=12
ARC0: Archival started
ARC1 started with pid=13

```

```
Thu Apr  1 11:11:13 2004
ARCH: STARTING ARCH PROCESSES COMPLETE
Thu Apr  1 11:11:13 2004
ARC0: Thread not mounted
Thu Apr  1 11:11:13 2004
ARC1: Archival started
ARC1: Thread not mounted
Thu Apr  1 11:11:14 2004
ALTER DATABASE  MOUNT
Thu Apr  1 11:11:18 2004
Successful mount of redo thread 1, with mount id 1088380178.
Thu Apr  1 11:11:18 2004
Database mounted in Exclusive Mode.
Completed: ALTER DATABASE  MOUNT
Thu Apr  1 11:11:27 2004
alter database open
Thu Apr  1 11:11:27 2004
Beginning crash recovery of 1 threads
Thu Apr  1 11:11:27 2004
Started first pass scan
Thu Apr  1 11:11:28 2004
Completed first pass scan
  1 redo blocks read, 0 data blocks need recovery
Thu Apr  1 11:11:28 2004
Started recovery at
  Thread 1: logseq 177, block 2, scn 0.33104793
Recovery of Online Redo Log: Thread 1 Group 3 Seq 177 Reading mem 0
  Mem# 0 errs 0: /u01/oradata/gzhs/redo03.log
Thu Apr  1 11:11:28 2004
Completed redo application
Thu Apr  1 11:11:28 2004
Ended recovery at
  Thread 1: logseq 177, block 3, scn 0.33124794
  0 data blocks read, 0 data blocks written, 1 redo blocks read
Crash recovery completed successfully
Thu Apr  1 11:11:28 2004
LGWR: Primary database is in CLUSTER CONSISTENT mode
Thread 1 advanced to log sequence 178
Thread 1 opened at log sequence 178
  Current log# 1 seq# 178 mem# 0: /u01/oradata/gzhs/redo01.log
Successful open of redo thread 1.
Thu Apr  1 11:11:28 2004
ARC0: Evaluating archive  log 3 thread 1 sequence 177
Thu Apr  1 11:11:28 2004
```

```
ARC0: Beginning to archive log 3 thread 1 sequence 177
Creating          archive          destination          LOG_ARCHIVE_DEST_1:
'/u06/oradata/gzhs/arch/1_177.dbf'
Thu Apr  1 11:11:28 2004
SMON: enabling cache recovery
ARC0: Completed archiving  log 3 thread 1 sequence 177
Thu Apr  1 11:11:28 2004
Errors in file /oracle/admin/gzhs/udump/gzhs_ora_27781.trc:
ORA-30012:          \263\267\317\373\261\355\277\325\274\344          'UNDOTBS1'
\262\273\264\346\324\332\273\362\300\340\320\315\262\273\325\375\310\
267
Thu Apr  1 11:11:28 2004
Error 30012 happened during db open, shutting down database
USER: terminating instance due to error 30012
Instance terminated by USER, pid = 27781
ORA-1092 signalled during: alter database open...
:q

在警报日志末尾显示了数据库在 Open 状态因为错误而异常终止。
```

## 2. 尝试重新启动数据库

```
bash-2.03$ sqlplus "/ as sysdba"

SQL*Plus: Release 9.2.0.3.0 - Production on 星期四 4月 1 11:43:52 2004

Copyright (c) 1982, 2002, Oracle Corporation. All rights reserved.

已连接到空闲例程。

SQL> startup
ORACLE 例程已经启动。

Total System Global Area 4364148184 bytes
Fixed Size 736728 bytes
Variable Size 1845493760 bytes
Database Buffers 2516582400 bytes
Redo Buffers 1335296 bytes
数据库装载完毕。
ORA-01092: ORACLE 例程终止。强行断开连接
```

问题重现.

### 3. 检查数据文件

```
bash-2.03$ cd /u01/ oradata/gzhs
bash-2.03$ ls -l
total 55702458
-rw-r----- 1 oracle dba 1073750016 Apr 1 11:44 UNDOTBS2.dbf
-rw-r----- 1 oracle dba 1073750016 Apr 1 11:44
WAP12_BILLINGDETAIL.dbf
-rw-r----- 1 oracle dba 1073750016 Apr 1 11:44 WAP12_MAIN.dbf
-rw-r----- 1 oracle dba 2097160192 Apr 1 11:44 WAP12_MAIN10.dbf
-rw-r----- 1 oracle dba 2097160192 Apr 1 11:44 WAP12_MAIN11.dbf
-rw-r----- 1 oracle dba 2097160192 Apr 1 11:44 WAP12_MAIN2.dbf
-rw-r----- 1 oracle dba 2097160192 Apr 1 11:44 WAP12_MAIN3.dbf
-rw-r----- 1 oracle dba 2097160192 Apr 1 11:44 WAP12_MAIN4.dbf
-rw-r----- 1 oracle dba 2097160192 Apr 1 11:44 WAP12_MAIN5.dbf
-rw-r----- 1 oracle dba 2097160192 Apr 1 11:44 WAP12_MAIN6.dbf
-rw-r----- 1 oracle dba 2097160192 Apr 1 11:44 WAP12_MAIN7.dbf
-rw-r----- 1 oracle dba 2097160192 Apr 1 11:44 WAP12_MAIN8.dbf
-rw-r----- 1 oracle dba 2097160192 Apr 1 11:44 WAP12_MAIN9.dbf
-rw-r----- 1 oracle dba 1073750016 Apr 1 11:44 WAP12_MVIEW.dbf
-rw-r----- 1 oracle dba 1073750016 Mar 24 17:15 WAP12_TEMP1.dbf
-rw-r----- 1 oracle dba 1810432 Apr 1 11:44 control01.ctl
-rw-r----- 1 oracle dba 104865792 Apr 1 11:44 cwmlite01.dbf
-rw-r----- 1 oracle dba 104865792 Apr 1 11:44 drsys01.dbf
-rw-r----- 1 oracle dba 144842752 Apr 1 11:44 example01.dbf
-rw-r----- 1 oracle dba 104865792 Apr 1 11:44 indx01.dbf
-rw-r--r-- 1 oracle dba 1113 Mar 31 22:51 mkqio.dat
-rw-r----- 1 oracle dba 104865792 Apr 1 11:44 odm01.dbf
-rw-r----- 1 oracle dba 268435968 Apr 1 11:44 redo01.log
-rw-r----- 1 oracle dba 268435968 Apr 1 11:44 redo02.log
-rw-r----- 1 oracle dba 268435968 Apr 1 11:44 redo03.log
-rw-r----- 1 oracle dba 524296192 Apr 1 11:44 system01.dbf
-rw-r----- 1 oracle dba 1048584192 Feb 7 21:10 temp01.dbf
-rw-r----- 1 oracle dba 104865792 Apr 1 11:44 tools01.dbf
-rw-r----- 1 oracle dba 209723392 Apr 1 11:44 users01.dbf
-rw-r----- 1 oracle dba 209723392 Apr 1 11:44 xdb01.dbf
```

发现存在文件 **UNDOTBS2.dbf**

### 4. mount 数据库，检查系统参数

```
bash-2.03$ sqlplus "/ as sysdba"

SQL*Plus: Release 9.2.0.3.0 - Production on 星期四 4月 1 11:46:20 2004

Copyright (c) 1982, 2002, Oracle Corporation. All rights reserved.
```



已连接到空闲例程。

SQL>

SQL>

SQL> startup mount;

ORACLE 例程已经启动。

Total System Global Area 4364148184 bytes

Fixed Size 736728 bytes

Variable Size 1845493760 bytes

Database Buffers 2516582400 bytes

Redo Buffers 1335296 bytes

数据库装载完毕。

SQL> select name from v\$datafile;

NAME

-----  
/u01/oradata/gzhs/system01.dbf  
/u01/oradata/gzhs/cwmlite01.dbf  
/u01/oradata/gzhs/drsys01.dbf  
/u01/oradata/gzhs/example01.dbf  
/u01/oradata/gzhs/indx01.dbf  
/u01/oradata/gzhs/odm01.dbf  
/u01/oradata/gzhs/tools01.dbf  
/u01/oradata/gzhs/users01.dbf  
/u01/oradata/gzhs/xdm01.dbf  
/u01/oradata/gzhs/WAP12\_MAIN.dbf  
/u01/oradata/gzhs/WAP12\_BILLINGDETAIL.dbf

NAME

-----  
/u01/oradata/gzhs/WAP12\_MVIEW.dbf  
/u01/oradata/gzhs/WAP12\_MAIN2.dbf  
/u01/oradata/gzhs/WAP12\_MAIN3.dbf  
/u01/oradata/gzhs/WAP12\_MAIN4.dbf  
/u01/oradata/gzhs/WAP12\_MAIN5.dbf  
/u01/oradata/gzhs/WAP12\_MAIN6.dbf  
/u01/oradata/gzhs/WAP12\_MAIN7.dbf  
/u01/oradata/gzhs/WAP12\_MAIN8.dbf  
/u01/oradata/gzhs/WAP12\_MAIN9.dbf  
/u01/oradata/gzhs/WAP12\_MAIN10.dbf  
/u01/oradata/gzhs/WAP12\_MAIN11.dbf

```

NAME
-----
/u01/oradata/gzhs/UNDOTBS2.dbf

已选择 23 行。

SQL>
SQL> show parameter undo

NAME                                TYPE                                VALUE
-----                                -                                -
undo_management                      string                              AUTO
undo_retention                       integer                             10800
undo_suppress_errors                 boolean                             FALSE
undo_tablespace                      string                              UNDOTBS1
SQL> show parameter spfile

NAME                                TYPE                                VALUE
-----                                -                                -
spfile                               string

```

发现系统没有使用 spfile,而初始化参数设置的 undo 表空间为 UNDOTBS1

### 5. 检查参数文件

```

bash-2.03$ cd $ORACLE_HOME/dbs
bash-2.03$ ls
init.ora          initgzhs.ora      initgzhs.ora.old  orapwgzhs
initdw.ora        initgzhs.ora.hurray lkGZHS             snapcf_gzhs.f
bash-2.03$ vi initgzhs.ora
"initgzhs.ora" [Incomplete last line] 105 lines, 3087 characters
#####
# Copyright (c) 1991, 2001, 2002 by Oracle Corporation
#####

#####
# Archive
#####
log_archive_dest_1='LOCATION=/u06/oradata/gzhs/arch'
log_archive_format=%t_%s.dbf
log_archive_start=true

#####
# Cache and I/O
#####

```

```
db_block_size=8192
db_cache_size=2516582400
db_file_multiblock_read_count=16

#####
# Cursors and Library Cache
#####
open_cursors=300

#####
# Database Identification
#####
db_domain=""
db_name=gzhs

#####
# Diagnostics and Statistics
#####
background_dump_dest=/oracle/admin/gzhs/bdump
core_dump_dest=/oracle/admin/gzhs/cdump
timed_statistics=TRUE
user_dump_dest=/oracle/admin/gzhs/udump

#####
# File Configuration
#####
control_files=("/u01/oradata/gzhs/control01.ctl",           "/u02/oradata/gzhs/control02.ctl",
"/u03/oradata/gzhs/control03.ctl")

#####
# Instance Identification
#####
instance_name=gzhs

#####
# Job Queues
#####
job_queue_processes=10

#####
# MTS
#####
dispatchers="(PROTOCOL=TCP) (SERVICE=gzhsXDB)"
```

```
#####  
# Miscellaneous  
#####  
aq_tm_processes=1  
compatible=9.2.0.0  
  
#####  
# Optimizer  
#####  
hash_join_enabled=TRUE  
query_rewrite_enabled=FALSE  
star_transformation_enabled=FALSE  
  
#####  
# Pools  
#####  
java_pool_size=629145600  
large_pool_size=8388608  
shared_pool_size=1153433600  
  
#####  
# Processes and Sessions  
#####  
processes=150  
  
#####  
# Redo Log and Recovery  
#####  
fast_start_mtr_target=300  
  
#####  
# Security and Auditing  
#####  
remote_login_passwordfile=EXCLUSIVE  
  
#####  
# Sort, Hash Joins, Bitmap Indexes  
#####  
pga_aggregate_target=838860800  
sort_area_size=524288  
  
#####  
# System Managed Undo and Rollback Segments  
#####
```

```
undo_management=AUTO
undo_retention=10800
undo_tablespace=UNDOTBS1

:q!
```

这个设置是极其可疑的.

6. 再次检查 alert 文件  
查找对于 UNDO 表空间的操作

第一部分, 创建数据库时的信息:

```
Sat Feb 7 20:30:12 2004
CREATE DATABASE gzhz
MAXINSTANCES 1
MAXLOGHISTORY 1
MAXLOGFILES 5
MAXLOGMEMBERS 3
MAXDATAFILES 100
DATAFILE '/u01/oradata/gzhz/system01.dbf' SIZE 500M REUSE AUTOEXTEND ON
NEXT 10240K MAXSIZE UNLIMITED
EXTENT MANAGEMENT LOCAL
DEFAULT TEMPORARY TABLESPACE TEMP TEMPFILE
'/u01/oradata/gzhz/temp01.dbf' SIZE 1000M REUSE AUTOEXTEND ON NEXT
250M MAXSIZE UNLIMITED
UNDO TABLESPACE "UNDOTBS1" DATAFILE
'/u01/oradata/gzhz/undotbs01.dbf' SIZE 1000M REUSE AUTOEXTEND ON NEXT
100M MAXSIZE UNLIMITED
CHARACTER SET ZHS16GBK
NATIONAL CHARACTER SET AL16UTF16
LOGFILE GROUP 1 ('/u01/oradata/gzhz/redo01.log') SIZE 256M,
GROUP 2 ('/u01/oradata/gzhz/redo02.log') SIZE 256M,
GROUP 3 ('/u01/oradata/gzhz/redo03.log') SIZE 256M
```

第二部分,发现创建 UNDOTBS2 的记录信息:

```
Wed Mar 24 20:20:58 2004
/* OracleOEM */ CREATE UNDO TABLESPACE "UNDOTBS2" DATAFILE
'/u01/oradata/gzhz/UNDOTBS2.dbf' SIZE 1024M AUTOEXTEND ON NEXT
100M MAXSIZE UNLIMITED
Wed Mar 24 20:22:37 2004
Created Undo Segment _SYSSMU11$
Created Undo Segment _SYSSMU12$
Created Undo Segment _SYSSMU13$
```

```
Created Undo Segment _SYSSMU14$
Created Undo Segment _SYSSMU15$
Created Undo Segment _SYSSMU16$
Created Undo Segment _SYSSMU17$
Created Undo Segment _SYSSMU18$
Created Undo Segment _SYSSMU19$
Created Undo Segment _SYSSMU20$
Completed: /* OracleOEM */ CREATE UNDO TABLESPACE "UNDOTBS2"
Wed Mar 24 20:24:25 2004
Undo Segment 11 Onlined
Undo Segment 12 Onlined
Undo Segment 13 Onlined
Undo Segment 14 Onlined
Undo Segment 15 Onlined
Undo Segment 16 Onlined
Undo Segment 17 Onlined
Undo Segment 18 Onlined
Undo Segment 19 Onlined
Undo Segment 20 Onlined
Successfully onlined Undo Tablespace 15.
Undo Segment 1 Offlined
Undo Segment 2 Offlined
Undo Segment 3 Offlined
Undo Segment 4 Offlined
Undo Segment 5 Offlined
Undo Segment 6 Offlined
Undo Segment 7 Offlined
Undo Segment 8 Offlined
Undo Segment 9 Offlined
Undo Segment 10 Offlined
Undo Tablespace 1 successfully switched out.
```

### 第三部分，新的 UNDO 表空间被应用

```
Wed Mar 24 20:24:25 2004
ALTER SYSTEM SET undo_tablespace='UNDOTBS2' SCOPE=MEMORY;
```

我们发现问题就在这里,创建了新的 UNDO 表空间以后,因为使用的是 pfile 文件,修改的只对当前实例生效,操作人员忘记了修改 pfile 文件.

如果使用 spfile,缺省的修改范围是 both,会同时修改 spfile 文件,就可以避免以上问题的出现.

#### 第四部分，删除了 UNDOTBS1 的信息

```
Wed Mar 24 20:25:01 2004
/* OracleOEM */ DROP TABLESPACE "UNDOTBS1" INCLUDING CONTENTS
AND DATAFILES CASCADE CONSTRAINTS
Wed Mar 24 20:25:03 2004
Deleted file /u01/oradata/gzhs/undotbs01.dbf
Completed: /* OracleOEM */ DROP TABLESPACE "UNDOTBS1" INCLUDI
```

这样再次重新启动数据库的时候，问题出现了，pfile 中定义的 UNDOTBS1 找不到了，而且操作实在很久以前，没人能回忆起来，甚至无法得知是什么人的操作。

#### 7. 更改 pfile，启动数据库

```
bash-2.03$ vi initgzhs.ora
"initgzhs.ora" [Incomplete last line] 105 lines, 3087 characters
#####
# Copyright (c) 1991, 2001, 2002 by Oracle Corporation
#####

#####
# Archive
#####
log_archive_dest_1='LOCATION=/u06/oradata/gzhs/arch'
log_archive_format=%t_%s.dbf
log_archive_start=true

#####
# Cache and I/O
#####
db_block_size=8192
db_cache_size=2516582400
db_file_multiblock_read_count=16

#####
# Cursors and Library Cache
#####
open_cursors=300

#####
# Database Identification
#####
db_domain=""
db_name=gzhs

#####
```

```
# Diagnostics and Statistics
#####
background_dump_dest=/oracle/admin/gzhs/bdump
core_dump_dest=/oracle/admin/gzhs/cdump
timed_statistics=TRUE
user_dump_dest=/oracle/admin/gzhs/udump

#####
# File Configuration
#####
control_files=("/u01/oradata/gzhs/control01.ctl",           "/u02/oradata/gzhs/control02.ctl",
"/u03/oradata/gzhs/control03.ctl")

#####
# Instance Identification
#####
instance_name=gzhs

#####
"initgzhs.ora" [Incomplete last line] 105 lines, 3087 characters

#####
# Redo Log and Recovery
#####
fast_start_mtr_target=300

#####
# Security and Auditing
#####
remote_login_passwordfile=EXCLUSIVE

#####
# Sort, Hash Joins, Bitmap Indexes
#####
pga_aggregate_target=838860800
sort_area_size=524288

#####
# System Managed Undo and Rollback Segments
#####
undo_management=AUTO
undo_retention=10800
undo_tablespace=UNDOTBS2
```



```
~  
~  
~  
~  
"initgzhs.ora" 105 lines, 3088 characters
```

## 8. 启动数据库

```
bash-2.03$ sqlplus "/ as sysdba"  
  
SQL*Plus: Release 9.2.0.3.0 - Production on 星期四 4月 1 11:55:11 2004  
  
Copyright (c) 1982, 2002, Oracle Corporation. All rights reserved.  
  
连接到:  
Oracle9i Enterprise Edition Release 9.2.0.3.0 - 64bit Production  
With the Partitioning, OLAP and Oracle Data Mining options  
JServer Release 9.2.0.3.0 - Production  
  
SQL> select * from v$version;  
  
BANNER  
-----  
Oracle9i Enterprise Edition Release 9.2.0.3.0 - 64bit Production  
PL/SQL Release 9.2.0.3.0 - Production  
CORE 9.2.0.3.0 Production  
TNS for Solaris: Version 9.2.0.3.0 - Production  
NLSRTL Version 9.2.0.3.0 - Production  
  
SQL> exit  
从 Oracle9i Enterprise Edition Release 9.2.0.3.0 - 64bit Production  
With the Partitioning, OLAP and Oracle Data Mining options  
JServer Release 9.2.0.3.0 - Production 中断开  
bash-2.03$
```

在这里我们可以看到，使用 spfile 可以免去手工修改 pfile 文件的麻烦，减少了犯错的可能。