

# ORA-1157 错误解决手册

## 一. 错误描述

ORA-1157, "cannot identify/lock data file %s - see DBWR trace file"

引起的原因：

因为数据文件已经在被使用了从而导致数据库的后台进程不能找到相应的数据文件或者不能锁定相应的数据文件,这样数据库将禁止访问这些数据文件而其他的数据文件则没有影响。伴随这个错误操作系统将会提示是哪个数据文件不能被识别。

ORA-01157 错误一般和ORA-01110 错误一起出现,往往还有操作系统级别上的错误,例如ORA-07360,同时一个DBWR的trace文件会在background\_dump\_dest的目录下生成。例如,在Solaris的平台上,会有如下的错误信息显示：

```
ORA-01157: cannot identify/lock data file 5 - see DBWR trace file
ORA-01110: data file 5: '/export/home/Oracle/oradata/817/users01.dbf'
然后查看DBWR的trace文件内容,会有如下的内容:
ORA-01157: cannot identify/lock data file 5 - see DBWR trace file
ORA-01110: data file 5: '/export/home /Oracle/oradata/817/users01.dbf'
ORA-27037: unable to obtain file status
SVR4 Error: 2: No such file or directory
Additional information: 3
```

下面就几个容易产生ORA-1157错误的方面详细谈谈：

## 二. 通常引起ORA-1157错误的原因和解决方法

如果你是使用Oracle9i,就用SQLPLUS代替SVRMGRL执行以下的命令。

### 1. 数据文件存在,但是Oracle认不到它

这种情况可能是在操作系统级上数据文件被重命名了或者移动到了一个新的分区或者位置,这种情况比较简单,只是需要将数据文件恢复成原始的数据文件名或者重新命名数据文件到一个新的位置/目录就可以解决问题了。

重新命名数据文件到一个新的位置/目录的方法：

#### A. 数据库是打开状态的

1) 查看那个数据文件所在的表空间还包含有哪些数据文件,执行以下查询：

```
SELECT FILE_NAME, STATUS FROM DBA_DATA_FILES
WHERE TABLESPACE_NAME = '<YOUR_TABLESPACE_NAME>';
```

2) 确定所有数据文件的状态都是可用的。

3) 把表空间变成只读表空间：

```
ALTER TABLESPACE <YOUR_TABLESPACE_NAME> READ ONLY;
```

4) 确定在数据字典中表空间是显示为只读的：

```
SELECT TABLESPACE_NAME, STATUS FROM DBA_TABLESPACES
WHERE TABLESPACE_NAME = '<YOUR_TABLESPACE_NAME>';
TABLESPACE_NAME          STATUS
```

```
-----
<YOUR_TABLESPACE_NAME>    READ ONLY
```

5) 用操作系统命令拷贝数据文件到一个新的位置,拷贝完成后把整个表空间

OFFLINE，这个时候用户是不能访问这个表空间的：

```
ALTER TABLESPACE <YOUR_TABLESPACE_NAME> OFFLINE；
```

- 6) 重命名这个数据文件到一个新的位置了，这个操作会自动的更新控制文件中的内容：

```
ALTER DATABASE RENAME FILE  
' /FULL_PATH_OF_OLD_LOCATION/AND_DATAFILE_NAME.DBF'  
TO  
' /FULL_PATH_OF_NEW_LOCATION/AND_DATAFILE_NAME.DBF'；
```

- 7) ONLINE 这个表空间：

```
ALTER TABLESPACE YOUR_TABLESPACE_NAME ONLINE；
```

- 8) 把这个表空间置成可读写的状态：

```
ALTER TABLESPACE YOUR_TABLESPACE_NAME READ WRITE；
```

- 9) 在操作系统级上删除原来旧的数据文件。

## B. 数据库是关闭状态的

- 1) 先正常关闭数据库。

- 2) 用操作系统命令拷贝数据文件到一个新的位置。

- 3) MOUNT 数据库，这样将读取控制文件，但是不会读取数据文件：

```
STARTUP MOUNT
```

- 4) 重命名这个数据文件到一个新的位置了，这个操作会自动的更新控制文件中的内容：

```
ALTER DATABASE RENAME FILE  
' /FULL_PATH_OF_OLD_LOCATION/AND_DATAFILE_NAME.DBF'  
TO  
' /FULL_PATH_OF_NEW_LOCATION/AND_DATAFILE_NAME.DBF'；
```

- 5) 打开数据库：

```
ALTER DATABASE OPEN；
```

## 2. 数据文件不存在或者对于 Oracle 来说是不可用的

数据文件被物理的移走了或者损坏导致 Oracle 不能再认识到了，例如数据文件被截断或者覆盖了，一般会出现 ORA-27046、ORA-1157 的错误提示：

ORA-27046: file size is not a multiple of logical block size

这种情况下可以有两种选择去解决问题：

### A. 重建数据文件所属的那个表空间

这种方法比较适用于 USERS、TEMP、INDEX 表空间，如果数据库是正常关闭的，也就是说回滚段中没有激活的表空间事务，也推荐使用这种方法。如果是 SYSTEM 表空间，则要重建数据库了。

具体步骤如下：

- 1) MOUNT 数据库：

```
STARTUP MOUNT PFILE='<location_of_pfile>'；
```

- 2) OFFLINE DROP 数据文件：

```
ALTER DATABASE DATAFILE '<full_path_file_name>' OFFLINE DROP；
```

- 3) 打开数据库：

```
ALTER DATABASE OPEN；
```

- 4) 删除表空间：

```
DROP TABLESPACE <tablespace_name> INCLUDING CONTENTS ;
```

5) 重建表空间 :

```
CREATE TABLESPACE <tablespace_name> DATAFILE  
'<datafile_full_path_name>' SIZE <required_size> ;
```

6) 重建表空间中所有以前存在的对象:可以使用以前创建对象的脚本或者利用最近可用的 EXPORT DUMP 来重建以前存在的对象。

## B. 用正常的恢复过程去恢复数据文件

这种方法比较适用于只读表空间或者那种不能用重建表空间的方法的 USERS 和 INDEX 表空间。如果是回滚段表空间,那必须要求数据库是正常关闭的才能使用这个方法。如果是 SYSTEM 表空间,并且备份和所有的归档日志都全的情况下,强烈建议使用这种方法去恢复的,但是如果是非归档方式,则就只能利用当前所有的联机日志进行恢复了。

在很多的情况下,重建表空间是不可能的或者是非常费时费力的,因此,从备份和利用归档日志恢复数据文件是一种比较好的方法,尤其是对于只读表空间来说,因为没有数据的写入和更改,因此直接用备份来恢复是最快最省事的。

具体步骤如下:

1) 从备份中恢复丢失或者损坏的数据文件。

2) MOUNT 数据库:

```
STARTUP MOUNT PFILE='<location_of_pfile>';
```

3) 执行以下的查询:

```
SELECT V1.GROUP#, MEMBER, SEQUENCE#,  
FIRST_CHANGE#  
FROM V$LOG V1, V$LOGFILE V2  
WHERE V1.GROUP# = V2.GROUP#;
```

这个查询将列出所有联机重做日志以及它们所代表的 SEQUENCE 和 FIRST CHANGE NUMBER.

4) 如果数据库是非归档状态下的,执行以下的查询:

```
SELECT FILE#, CHANGE# FROM V$RECOVER_FILE ;
```

如果 CHANGE#大于最小的联机重做日志文件的 FIRST\_CHANGE#,那么数据文件可以被恢复,记住恢复数据文件的时候要应用所有的联机重做日志文件,然后到第 5 步。

如果 CHANGE#小于最小的联机重做日志文件的 FIRST\_CHANGE#,那么这个数据文件将不能被恢复了,那么只能从最近的数据库全备份恢复或者重建这个数据文件所属的表空间了。

5) 恢复数据文件:

```
RECOVER DATAFILE '<full_path_file_name>';
```

6) 确认所有的归档日志都被应用了直至出现"Media recovery complete"的提示信息,如果 Oracle 提示有一个不存在的归档日志文件,那么就可能要应用所有的联机重做日志文件来恢复直至出现"Media recovery complete"的提示信息。

7) 打开数据库:

```
ALTER DATABASE OPEN ;
```

### 3. 数据库临时表空间的数据文件的丢失

当数据库的临时表空间的数据文件丢失也会引起 ORA-01157 的错误。因为数据库对临时表空间的数据文件不会发生检查点，所以这个时候数据库照样能够打开。这种情况下的解决方法是逻辑上删除临时表空间的数据文件，并且重新增加一个新的临时表空间的数据文件。

例如：

```
SELECT * FROM DBA_OBJECTS ORDER BY OBJECT_NAME ;
select * from dba_objects order by object_name ;
* ERROR at line 1:
ORA-01157: cannot identify/lock data file 5 - see DBWR trace file
ORA-01110: data file 5: '/Oracle/oradata/temp01.dbf'
ALTER DATABASE TEMPFILE '/Oracle/oradata/temp01.dbf' DROP ;
SELECT TABLESPACE_NAME,FILE_NAME FROM DBA_TEMP_FILES ;
ALTER TABLESPACE TEMP ADD TEMPFILE '/Oracle/oradata/temp01.dbf' SIZE
100M ;
```

### 三. 由于操作系统的问题或者第三方软件的问题导致 ORA-01157 错误

1. 当使用 vxfsstat 去访问快速 I/O 或者其它的应用，会获得"Cannot open file" 的错误，而 Oracle 会返回如下的错误：

```
ORA-01157: cannot identify data file 1 - file not found
ORA-01110: data file 1: '<filename>'
```

这个时候用户应该去联系Veritas的技术支持，技术支持网站网址为 <http://support.veritas.com/>。

2. 在 HP-UNIX 的机器上，如果系统核心参数 nlock 设置不是足够大的时候，这样可能会使 Oracle 不能锁定所需要的数据文件而导致错误：

```
ORA-27086: skgflk: unable to lock file - already in use
或者错误：
```

```
ORA-01157: cannot identify/lock data file 4 - see DBWR trace file
ORA-01110: data file 4: '/Oracle/oradata/user01.dbf'
```

```
ORA-27041: unable to open file
HP-UX Error: 23: File table overflow
Additional information: 2
```

或者错误：

```
ORA-07445: exception encountered: core dump [%s] [%s] [%s] [%s] [%s] [%s]
ORA-01110: data file %s: '%s'
```

```
ORA-01242: data file suffered media failure: database in NOARCHIVELOG mode
ORA-01115: IO error reading block from file %s (block # %s)
```

```
ORA-27041: unable to open file
HP-UX Error: 23: File table overflow
Additional information: 3
```

解决这个问题的方法是增大相关的核心参数：（建议以下的配置）

```
nproc      4096      Max Number of Processes
nfile      63488     Max Number of Open Files
nlocks     4096      Max Number of File Locks
```

3. 如果 Oracle 需要的数据文件被其他进程锁定的条件下也会导致这个错误。

例如：备份软件将可能锁定要备份的数据文件

在 WINDOWS 上可能会有如下的错误：

ORA-01157: signalled during alter database open

ORA-01157: can not identify datafile <datafile id>

ORA-01110: datafile <datafile id> path and filename of datafile

ORA-27047: Unable to read header of file <datafile id>

OSD-04006: Read file failure

Error 33: process can not access file

操作系统错误 33 是一个 error\_lock\_violation，表明一部分数据文件被 WINDOWS 的其他进程锁定了。

或者错误：

ORA-1157 - cannot identify datafile <name> - file not found

ORA-1110 - datafile <name>: <str>

ORA-9202 - sfifi: error identifying file

OSD-4006(OS 203) - The System could not find the environment option that was entered

在 ALERT 文件中将会同时出现以下的错误：

ORA-1115 - IO error reading block from file %s (block # %s)

ORA-1110 - datafile <name>: <str>

ORA-9206 - sfrfb: error reading from file

OSD-4006(OS 203) - The System Could not find the environment option that was entered

或者错误：

ORA-1242 - data file suffered media failure: database in NOARCHIVELOG mode

ORA-1114 - IO error writing block to file <name> block # <num>

ORA-9205 - sfqio: error reading or writing to disk

OSD-4016(OS 33) - The process cannot access the file because another process has locked a portion of the file

另外还可能会出现以下错误：

KCF: write/open error dba=0x703473d block=0x3473d online=1  
file=7 E:\Oracle\data\grec\crecind2.dbf

error=9211 txt: 'OSD-4008 : WriteFile error (OS 203) - The System Could not find the environment option that was entered

某些情况下 ALERT 文件中会出现：

Instance terminating due to error 1110.

Instance terminated by <background process> PID=XXX

或者：

<background process> TERMINATING INSTANCE DUE TO ERROR 472

ORA 472 - PMON process terminated with error

在 WINDOWS 的事件查看器中可以看到以下事件：

23 Error ReadFile() failure

25 Error WriteFile() failure

如果这是个冷备份,那就要等冷备份完成后启动数据库或者结束冷备份启动数据库。对于备份软件,最好都配置成不要锁定打开的数据文件的备份方式。这种情况的解决方法是手工的清除在数据文件上的锁:

- 1) 运行 `ps -ef | grep <SID>`, 查出在数据文件上已经存在的进程。
- 2) 运行 `kill -9 进程 ID`

4. 使用 WINDOWS 的 FILE MANAGER 拷贝 Oracle 的数据文件的时候也会引起 ORA-01157 的错误, 例如文件名大于通常用的 8.3 格式, 如果文件名大于 8 个字符或者你的扩展名大于 3 个字符就会引起这个错误。要避免这个错误, 在 WINDOWS 下拷贝文件不要用 FILE MANAGER, 最好使用浏览器去拷贝文件, 如果已经使用 FILE MANAGER, 那么对于长文件名的文件会自动加上一个 ~, 这样要重新命名拷贝的文件为原来的文件名字。

5. 使用网络应用工具也可能会引起 ORA-01157 的错误。

在一些网络工具的使用操作中要求对数据文件进行加锁, 如果由于实例错误或者主机的问题可能会导致这些锁会一直的存在, 这种情况下需要系统管理员手工的去释放这些锁。

6. 如果 Oracle 的数据文件被一个其他的用户恢复也可能引起 ORA-01157 的错误。在 Oracle 的数据文件被恢复之后, Oracle 数据库认不到恢复后的数据文件, 因此错误 ORA-1157 (cannot identify datafile - file not found) 就可能发生:

- 数据文件在操作系统上是否存在
- `SELECT * FROM V$DATAFILE` 查看数据文件的正确路径
- `ALTER SYSTEM CHECK DATAFILE` 是否成功
- 使用 `BACKUP CONTROLFILE TO TRACE` 查看数据文件的正确路径

一般出现这种问题有可能是操作系统上的权限问题, 首先查看数据文件的权限, 当数据文件被其他用户恢复的时候可能权限就变了, 可能 Oracle 用户就不能访问了, 这样就要对恢复后的数据文件修改权限和属主。

7. ULIMIT 设置的值不够大也可能会引起 ORA-01157 的错误。

在 DBWR 的跟踪文件中会有 ORA-1157 和 ORA-27092 的错误:

```
ORA-01157: cannot identify/lock data file N - see DBWR trace file
ORA-01110: data file 1: '<filename>'
ORA-27092: skgfofi: size of file exceeds file size limit of the process
Additional information: xxxxx
Additional information: yyyyy
```

Oracle8.1.7 对于打开数据库会执行很严格的在操作系统的上的 ULIMIT 的检查, 如果文件大小的限制不够大, 则数据库就会打不开, 出现以上的错误。因此就要增大 ULIMIT:

```
ULIMIT -f <require_size_of_file_in_os_blocks>;
```

#### 四. 在移植过程中出现 ORA-01157 的错误

1. 如果使用移植工具把 Oracle7 数据库升级到 Oracle8i 数据库, 当执行数据库转换的时候有可能出现以下的错误:

```
ORA-1157 cannot identify datafile <name> - file not found
ORA-1110 datafile <name>: <str>
```

移植工具首先使用 Oracle7 的控制文件去创建一个 CONVERT.ORA 文件,当增加一个新的表空间或者新的数据文件如果新增数据文件没有包含全路径,导致在 CONVERT.ORA 文件中就没有数据文件路径正确的指向。

解决方法一是要修改%Oracle\_home%\rdbmsxx\convert.ora 下的 CONVERT.ORA 文件中的数据文件的路径为正确的路径,然后重新执行数据库转换。

解决方法二是先用备份恢复 Oracle7 的数据库,然后重新创建控制文件,修改数据文件的路径为正确的路径,然后重新执行移植过程。

## 2. 使用移植工具把数据库 Oracle7.3.X 移植到 Oracle8.1.X,可能会出现以下错误:

ORA-01157: cannot identify/lock data file 2 - see DBWR tracefile

ORA-01110: data file 2: '/oradata/V734/users01.dbf'

ORA-27046: file size is not a multiple of logical block size

Additional information:1

一般是数据文件从裸设备 dd 到文件系统中,数据文件的大小不是严格的 Oracle Block Size 的整数倍造成的。例如:

file size = 839911424 bytes

Oracle block size = 8092 bytes

解决方法一是把数据文件 RESIZE 到一个 Oracle Block Size 的整数倍:

ALTER DATABASE DATAFILE '<filename>' RESIZE <VALUE IN KB OR MB>;

- the integer should be a multiple of 8 in our example

解决方法二:

### 1) 使用 dbfsize 命令去获取数据文件的在数据库中的大小:

dbfsize <file\_name>

### 2) 查看数据文件在操作系统上的大小:

ls -lt <file\_name>

### 3) 使用 MOD 函数对比 1) 和 2) 的值,得出余数。

### 4) 确定数据库已经关闭了,然后使用 dd 命令。

例如:

操作系统上的文件大小是 2097203200 bytes,使用 dbfsize 得出的结果是 511744 4096 byte blocks,那么使用以下命令:

dd if=<some\_name>bs=4096 count=511745

NB: count= 511744 + 1 (1 for recovering from this problem)

mv <some\_file> to <file\_name>

### 5) startup nomount

alter database convert;

## 五. 其他一些可能产生 ORA-01157 错误的原因

### 1. 控制文件的突然中断引起 ORA-01157 的错误。

A. 一种可能的原因是在控制文件中的文件名的结尾处有一个空格。

可以使用 'ALTER DATABASE BACKUP CONTROLFILE TO TRACE' 命令,然后在初始化参数 user\_dump\_dest 所指向的目录下面查找相应的 TRACE 文件,查看控制文件的内容。例如:

'/home/d/Oracle/oradata/ecn/rdx02.dbf ' <-- corrupt

'/home/d/Oracle/oradata/ecn/rdx02.dbf' <-- non-corrupt

这种情况下用好的控制文件代替坏了的控制文件，并修改初始化参数文件中的 CONTROL\_FILES 参数，去掉坏了的控制文件。如果所有的控制文件都损坏了，那就需要重建控制文件了。

重建控制文件的方法：

1) 以 SYS 用户登陆，执行

```
ALTER DATABASE BACKUP CONTROLFILE TO TRACE;
```

2) 生成的 TRACE 文件在 USER\_DUMP\_DEST 的目录下，然后查看一下

USER\_DUMP\_DEST 的具体目录路径：

```
SELECT VALUE FROM V$PARAMETER WHERE NAME= ' USER_DUMP_DEST ' ;
```

```
或者 SHOW PARAMETER USER_DUMP_DEST ;
```

3) 找出相应的 TRACE 文件，最简单的找正确的 TRACE 文件的方法是看 TRACE 文件的创建时间，然后修改 TRACE 文件保存成一个 SQL 脚本，例如：

```
STARTUP NOMOUNT
```

```
CREATE CONTROLFILE REUSE DATABASE "ORCL" NORESETLOGS NOARCHIVELOG
```

```
MAXLOGFILES 5
```

```
MAXLOGMEMBERS 3
```

```
MAXDATAFILES 100
```

```
MAXINSTANCES 1
```

```
MAXLOGHISTORY 453
```

```
LOGFILE
```

```
GROUP 1 'D:\ORACLE\ORADATA\ORCL\REDO01.LOG' SIZE 1M,
```

```
GROUP 2 'D:\ORACLE\ORADATA\ORCL\REDO02.LOG' SIZE 1M,
```

```
GROUP 3 'D:\ORACLE\ORADATA\ORCL\REDO03.LOG' SIZE 1M
```

```
DATAFILE
```

```
'D:\ORACLE\ORADATA\ORCL\SYSTEM01.DBF',
```

```
'D:\ORACLE\ORADATA\ORCL\UNDOTBS01.DBF',
```

```
'D:\ORACLE\ORADATA\ORCL\OEM_REPOSITORY.DBF'
```

```
CHARACTER SET ZHS16GBK
```

```
;
```

4) 关闭数据库：

```
SHUTDOWN IMMEDIATE ;
```

5) 对数据库做一个全库的冷备份。

6) 利用操作系统命令将原来的控制文件移走。

7) 在 SQLPLUS 中以 SYS 用户运行刚刚保存的那个脚本。

8) 打开数据库。

**2. 在 STANDBY 方式下，如果主数据库增加了表空间或者数据文件，而从数据库中没**  
**有手工增加的话也会出现 ORA-01157 的错误。**

**3. RMAN 恢复会在 ALERT.LOG 中产生 ' FAKE ' 引起 ORA-01157 的错误。**

在 RMAN 的恢复操作中，在 ALERT.LOG 中会产生以下的错误：

```
ORA-01157: cannot identify/lock data file N - see DBWR trace file
```

```
ORA-01110: data file N: '<filename>'
```

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

```
SVR4 Error: 2: No such file or directory
```

产生这种错误的原因主要是因为是在 RMAN 恢复之前数据文件已经被删除。