

1. 数据库打开的状态

```
ALTER DATABASE OPEN
ALTER DATABASE READ ONLY
ALTER DATABASE READ WRITE
```

2. 把数据库置于限制模式

有时需要维护数据库结构，或者对数据库进行导入和导出等操作时，需要用到限制模式。打开限制模式有两种方法：

2.1. 实例启动的时候把数据库置于限制模式

```
STARTUP RESTRICT
```

2.2. 在数据库打开后可以用命令置于限制模式

```
ALTER SYSTEM ENABLE RESTRICTED SESSION
```

2.3. 取消数据库的限制模式

```
ALTER SYSTEM DISABLE RESTRICTED SESSION
```

3. 管理员强制断开某个客户端的连接。

3.1 查看哪些用户连到服务器

```
SELECT SID,SADDR,SERIAL#,USER#,USERNAME FROM V$SESSION;
```

3.2 得到 S I D 的值和 S E R I A L # 的值。根据这两个值来强制断开用户的连接。

```
ALTER SYSTEM KILL SESSION 'SID,SERIAL#';
```

4. 数据字典视图

数据字典视图分为 3 大类。分别以 USER,ALL,DBA 前缀来区分。

USER_*: 有关用户所拥有的对象的信息，即用户自己创建的对象的信息。

ALL_*: 有关用户可以访问的对象信息，即用户自己创建的对象的信息再加上其他用户创建的对象但该用户有权访问的信息。

DBA_*: 有关整个数据库中对象的信息。

4.1 查询不同的数据字典中有哪些信息。

```
DESC USER_OBJECTS
```

```
DESC ALL_OBJECTS
```

```
DESC DBA_OBJECTS
```

```
SELECT OBJECT_NAME,OBJECT_TYPE FROM USER_OBJECTS WHERE OBJECT_TYPE='VIEW';
```

```
SELECT OBJECT_NAME,OBJECT_TYPE FROM ALL_OBJECTS WHERE OBJECT_TYPE='VIEW';
```

```
SELECT OBJECT_NAME,OBJECT_TYPE FROM DBA_OBJECTS WHERE OBJECT_TYPE='VIEW';
```

4.2 查看用户拥有的对象。

```
SELECT OBJECT_NAME,OBJECT_TYPE FROM ALL_OBJECTS WHERE OWNER='SCOTT';
```

4.3 动态性能表（视图）

s y s 用户拥有所有的动态性能视图。他们的名字都是以 v\$ 开头。这些动态性能视图是不能进行 D M L 操作的。利用动态性能视图可以获取如下信息：

会话活动的信息。

对象打开或者关闭的信息。

对象在线或者离线的信息等。

4.4 如何能知道所使用的数据库上有哪些数据字典

可以通过查询数据字典 v\$fixed_table 或者 dictionary 来得到 Oracle 数据库中所有的数据字典。

```
SELECT TABLE_NAME,COMMENTS FROM DICTIONARY
```

这个表中的查询结果非常多可以加上一些条件过滤掉一些内容。

```
SELECT TABLE_NAME,COMMENTS FROM DICTIONARY WHERE TABLE_NAME LIKE '%TABLE%';
```

4.4.1 系统中常用的动态性能表有

```
SELECT NAME,CREATED,LOG_MODE,OPEN_MODE FROM V$DATABASE;
```

此视图查看数据名称，创建日期，日志是否归档，数据库的打开模式。

```
SELECT HOST_NAME,INSTANCE_NAME,VERSION FROM V$INSTANCE;
```

此视图查看实例启动的主机名，实例名称和实例的版本。

```
SELECT * FROM V$VERSION;
```

此视图查看数据库的版本。

```
SELECT * FROM V$CONTROLFILE;
```

此视图查看控制文件的信息。

```
SELECT GROUP#,MEMBERS,BYTES,STATUS,ARCHIVED FROM V$LOG;
```

此视图查看重做日志的配置信息。不显示重做日志存放的具体位置。

```
SELECT * FROM V$LOGFILE;
```

此视图查看重做日志保存的具体的位置。

```
ARCHIVE LOG LIST;
```

查看日志是否归档。

```
SELECT TABLESPACE_NAME,BLOCK_SIZE,STATUS,CONTENTS,LOGGING FROM DBA_TABLESPACES;
```

查看表空间的名称, 块大小, 状态是否在线, 表空间的类型, 表空间是否打开的日志记录功能。

```
SELECT FILE_ID,FILE_NAME,TABLESPACE_NAME,STATUS,BYTES FROM DBA_DATA_FILES;
```

此视图查看了数据文件保存的位置, 所属的表空间名称, 是否可用和数据文件的大小。

```
SELECT USERNAME,CREATED FROM DBA_USERS;
```

此视图查看当前系统中的用户和用户创建的时间。

5. 控制文件

5.1 控制文件中存放的内容

1) 数据库的名字。

2) 数据库标识符。

3) 数据库创建的时间戳。

4) 联机重做日志文件的名称和准确位置。

5) 当前日志的序列号。

6) 校验点信息。

7) 日志的历史信息。

8) 归档日志文件的准确位置和状态。

9) 数据文件的名称和准确位置。

10) 表空间信息。

11) 备份的准确位置和状态。

5.2 从控制文件中获取信息的数据字典

可以使用如下数据字典从控制文件中抽取信息:

```
V$ARCHIVE
```

```
V$ARCHIVED_LOG
```

```
V$BACKUP
```

```
V$DATABASE
```

```
V$DATAFILE
```

```
V$LOGFILE
```

```
V$LOG
```

```
V$LOGHIST
```

```
V$TABLESPACE
```

```
V$TEMPFILE
```

5.3 获取控制文件的全部信息

```
SELECT TYPE,RECORD_SIZE,RECORDS_TOTAL,RECORDS_USED FROM  
V$CONTROLFILE_RECORD_SECTION;
```

5.4 控制文件的备份

```
ALTER DATABASE BACKUP CONTROLFILE TO '/home/oracle/control.bak'
```

6. 重做日志文件

要保证oracle数据库正常工作需要至少两组重做日志组。

重做日志写进程 (LGWR) 在任意时刻只能写一组重做日志组, LGWR后台进程正在写的重做日志组成为当前 (CURRENT) 重做日志组。当LGWR写满了一组重做日志时, 它就开始写下一个重做日志。这称为日志切换。当写满了最后一组重做日志时, LGWR开始写第一组重做日志。

当LGWR的写操作从一个重做日志组切换到另一个重做日志组后, 归档写进程 (ARCH/ARCRO) 就会将原来的重做日志文件中的信息复制到归档日志文件中。

如果重做日志组不是oracle服务器当前正在操作的组, 即使该组中所有的成员都坏了, oracle服务器照常工作, 只有等oracle切换到该组时oracle才会关闭系统。

6.1 强制产生日志切换和检查点的命令

```
ALTER SYSTEM SWITCH LOGFILE;
```

强制进行日志切换

```
ALTER SYSTEM CHECKPOINT;
```

强制产生检查点的命令。

6.3 通过参数强制产生检查点

SQL> ALTER SYSTEM SET fast_start_mttr_target=900;

此句话的意思是：实例恢复的时间不会超过 900 秒（15 分钟）。这样就可以满足：如果 oracle 系统崩溃了，实例恢复的时间最长为 15 分钟的要求。

设置

6.4 获取重做日志组的信息

SQL> SELECT GROUP#,SEQUENCE#,MEMBERS,BYTES,STATUS,ARCHIVED FROM V\$LOG;

GROUP#	SEQUENCE#	MEMBERS	BYTES	STATUS	ARC
1	12	2	52428800	INACTIVE	YES
3	13	1	52428800	INACTIVE	YES
4	14	2	10485760	CURRENT	NO

重做日志状态含义解释：

INACTIVE: 表示实例恢复已经不再需要这组联机重做日志组了。在 oracle 日志组大于 2 组的情况下可以被删除。

ACTIVE: 表示这组联机重做日志组是活动的但不是当前组，在实例恢复时需要这组联机重做日志组。如这组重做日志正在归档。不可以删除。

CURRENT: 表示这组联机重做日志组是当前组，并也隐含该联机重做日志组是活动的。不可以删除。

UNUSED: 表示 oracle 服务器从来没写过该组联机重做日志组，这是重做日志刚被添加到数据库中的状态。可以随时删除。

6.5 获取联机重做日志文件的信息

SQL> SELECT * FROM V\$LOGFILE;

联机重做日志的状态含义解释：

空白: 表示该文件正在使用中。

stale: 表示该文件中的内容是不完全的。

invalid: 表示该文件不可以被访问。

deleted: 表示该文件已不再有用了。

6.6 添加了删除联机重做日志文件组

ALTER DATABASE ADD LOGFILE GROUP 5 '/tmp/log51.LOG' SIZE 100M;

其中参数 GROUP 是可以不写的，如果后面没有 GROUP 选项，所以 oracle 系统会自动地在最大的组号上加 1 来产生新的组号。但是如果之前有一个中间的重做日志组不存在了，oracle 会先填充这个组号。如下面所示

ALTER DATABASE ADD LOGFILE '/tmp/log61.log' SIZE 100M;

上面的命令就会产生一个新的日志文件组。组号是之前最大的组号加 1。

ALTER DATABASE DROP LOGFILE GROUP 5;

删除日志文件组 5。请查看 6.4 中日志组处于不同状态，来判断是否此日志文件组能否删除。

6.7 添加删除联机重做日志成员

ALTER DATABASE ADD LOGFILE MEMBER '/tmp/log21.log' TO GROUP 2;

为联机重做日志组 2 添加一个新的日志成员文件。

ALTER DATABASE DROP LOGFILE MEMBER '/tmp/log21.log';

删除制定联机重做日志文件。

不能删除当前重做日志组中的成员。如果要删除，应该先使用 ALTER SYSTEM SWITCH LOGFILE 命令进行切换。

每个重做日志组中至少有一个成员才能正常工作。因此如果要删除的成员是该组中最后一个有效的成员，就不能删除它。如果数据库运行在归档模式，而要删除的成员还没有被归档完，那也无法删除它。

每次删除重做日志成员后，还要用系统命令手工删除保存在硬盘上的文件。

6.8 设置成归档日志模式

数据库在 mount 状态下。

ALTER DATABASE ARCHIVELOG;

取消数据库的归档日志模式

ALTER DATABASE NOARCHIVELOG;

6.9 清除联机重做日志文件

如果不能用“将有问题的重做日志组或成员删除掉，之后再重建他们”的方法来进行重做日志的维护和修复。

可以使用 ALTER DATABASE CLEAR LOGFILE 命令重新初始化联机重做日志文件。

ALTER DATABASE CLEAR LOGFILE GROUP 组号

如果崩溃的重做日志文件已经不能归档，可以在以上的命令中使用 UNARCHIVED 关键字来清除已崩溃的重做

日志文件，而避免对他们进行归档。

ALTER DATABASE CLEAR UNARCHIVED LOGFILE GROUP 组号

6.9.1 在做清空日志文件的操作之前应该先做一个完全备份。

6.9.2 清空日志文件

ALTER DATABASE CLEAR LOGFILE GROUP 3;

6.9.3 查看重做日志组的信息

SELECT GROUP#,SEQUENCE#,MEMBERS,BYTES,STATUS,ARCHIVED FROM V\$LOG;

可以看出，第3组重做日志的序列号（SEQUENCE#）已经变为0；

7. 调优相关(了解)

7.1 数据库数据高速缓存

高速缓存存储最近使用的数据块。

尺寸由初始化参数 DB_CACHE_SIZE 决定。

7.2 共享池

共享池的尺寸由初始化参数 SHARED_POOL_SIZE 决定。

共享池包括库缓存：

库缓存包含 SQL 和 PL/SQL 文本，经过编译后的代码，以及执行计划。

和数据字典缓存：

数据字典缓存包含表，列等对象定义以及权限信息。

注意：不能手工设定库缓存和数据字典缓存，只要设定 SHARED_POOL_SIZE 就可以了，oracle 会自动分配它们的大小。

注意：在查看 DB_CACHE_SIZE 和 SHARED_POOL_SIZE 时得到其值为 0，是由于 oracle10g 以后有自动共享内存管理机制来分配它们的大小。

由参数 SGA_TARGET 决定。当 SGA_TARGET 值为 0，则每个内存手工设定；当 SGA_TARGET 的值大于 0，完全有 oracle 自己调整并且当给 SHARED_POOL_SIZE 设置某值后，则设定的值就是 SHARED_POOL_SIZE 的最小值。

在 oracle11g 后更进一步。SGA_TARGET 和 PGA_TARGET 加在一起设定一个总内存值。由 oracle 自动为 SGA_TARGET 和 PGA_TARGET 分配他们的值。

7.3 重做日志缓存

尺寸由初始化参数 LOG_BUFFER 决定。日志缓冲中大小够 1M 就会把日志写入日志文件。

查看日志缓存大小信息

SQL> SHOW PARAMETER LOG_B;

7.4 自动共享内存管理

由参数 SGA_TARGET 决定 SGA 的总大小。建议是总内存的 1/2~2/3

SGA 中包括：共享池，流池，大池，java 池，数据缓存，重做日志缓存。

如果 SGA_MAX_SIZE 没有写到参数文件中，则 SGA_MAX_SIZE 的大小随 SGA_TARGET 的大小而变化。如果设定了值，这个值就是最大可以分配的内存大小，SGA_TARGET 不能超过这个值。一般建议这两个参数的值一样大就可以了。

查看 SGA 中各个内存参数的信息。

SQL> SELECT * FROM V\$SGAINFO;

其中如果 RES 为 YES 则为自动调整，除了 FIXED SGA SIZE 与 REDO BUFFERS 要加在一起才是 GRANULE SIZE 的整数倍，其他的参数一定要为 GRANULE SIZE 的整数倍。

NAME BYTES RES (是否自动调节)

```
-----
Fixed SGA Size          2020224 No  固定内存.
Redo Buffers            2170880 No
Buffer Cache Size       176160768 Yes
Shared Pool Size         96468992 Yes
Large Pool Size          4194304 Yes
Java Pool Size           4194304 Yes
Streams Pool Size        0 Yes
Granule Size             4194304 No  给其他的参数分配的此参数值的倍数.
Maximum SGA Size        285212672 No
Startup overhead in Shared Pool 41943040 No
Free SGA Memory Available 0
```

11 rows selected.

7.5 程序全局区 PGA

查看 PGA 的信息.

```
SQL> SHOW PARAMETER PGA;
```

是用户私有的,不可共享.

仅仅可由服务器进程写.

包含:

- 排序区

- 用户会话信息

- 光标信息

- 堆栈空间.

在联机事物处理中,如果排序很小的时候 PGA 应该占用总大小的 20%而 SGA 应该占 80%;在排序很大的时候应该各占 50%.

7.6 参数文件中参数的查找

SELECT NAME,VALUE FROM V\$SPPARAMETER WHERE NAME='sga_max_size';查看参数文件里面的值.

SELECT NAME,VALUE/1024/1024 FROM V\$PARAMETER WHERE NAME='SGA_MAX_SIZE';查看当前值.