oracle 常用SQL查询

一、ORACLE的启动和关闭

 1 、在单机环境下

要想启动或关闭ORACLE系统必须首先切换到ORACLE用户，如下

su  -  oracle

a、启动ORACLE系统

oracle > svrmgrl

SVRMGR > connect internal

SVRMGR > startup

SVRMGR > quit

b、关闭ORACLE系统

oracle > svrmgrl

SVRMGR > connect internal

SVRMGR > shutdown

SVRMGR > quit

启动oracle9i数据库命令：

$ sqlplus  / nolog

SQL \* Plus: Release  9.2 . 0.1 . 0   -  Production  on  Fri Oct  31   13 : 53 : 53   2003

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

SQL >  connect  /   as  sysdba

Connected  to  an idle instance.

SQL >  startup ^ C

SQL >  startup

ORACLE instance started.

 2 、在双机环境下

要想启动或关闭ORACLE系统必须首先切换到root用户，如下

su － root

a、启动ORACLE系统

hareg －y oracle

b、关闭ORACLE系统

hareg －n oracle

Oracle数据库有哪几种启动方式

说明：

有以下几种启动方式：

 1 、startup nomount

非安装启动，这种方式启动下可执行：重建控制文件、重建数据库

读取init.ora文件，启动instance，即启动SGA和后台进程，这种启动只需要init.ora文件。

 2 、startup mount dbname

安装启动，这种方式启动下可执行：

数据库日志归档、

数据库介质恢复、

使数据文件联机或脱机，

重新定位数据文件、重做日志文件。

执行“nomount”，然后打开控制文件，确认数据文件和联机日志文件的位置，

但此时不对数据文件和日志文件进行校验检查。

 3 、startup  open  dbname

先执行“nomount”，然后执行“mount”，再打开包括Redo log文件在内的所有数据库文件，

这种方式下可访问数据库中的数据。

 4 、startup，等于以下三个命令

startup nomount

 alter   database  mount

 alter   database   open

 5 、startup  restrict

约束方式启动

这种方式能够启动数据库，但只允许具有一定特权的用户访问

非特权用户访问时，会出现以下提示：

ERROR：

ORA - 01035 : ORACLE 只允许具有 RESTRICTED SESSION 权限的用户使用

 6 、startup force

强制启动方式

当不能关闭数据库时，可以用startup force来完成数据库的关闭

先关闭数据库，再执行正常启动数据库命令

 7 、startup pfile = 参数文件名

带初始化参数文件的启动方式

先读取参数文件，再按参数文件中的设置启动数据库

例：startup pfile = E:Oracleadminoradbpfileinit.ora

 8 、startup EXCLUSIVE

二、用户如何有效地利用数据字典

　ORACLE的数据字典是数据库的重要组成部分之一，它随着数据库的产生而产生, 随着数据库的变化而变化,

体现为sys用户下的一些表和视图。数据字典名称是大写的英文字符。

数据字典里存有用户信息、用户的权限信息、所有数据对象信息、表的约束条件、统计分析数据库的视图等。

我们不能手工修改数据字典里的信息。

　　很多时候，一般的ORACLE用户不知道如何有效地利用它。

　　dictionary　　　全部数据字典表的名称和解释，它有一个同义词dict

dict\_column　　 全部数据字典表里字段名称和解释

如果我们想查询跟索引有关的数据字典时，可以用下面这条SQL语句:

SQL > select   \*   from  dictionary  where  instr(comments, ' index ' ) > 0 ;

如果我们想知道user\_indexes表各字段名称的详细含义，可以用下面这条SQL语句:

SQL > select  column\_name,comments  from  dict\_columns  where  table\_name = ' USER\_INDEXES ' ;

依此类推，就可以轻松知道数据字典的详细名称和解释，不用查看ORACLE的其它文档资料了。

下面按类别列出一些ORACLE用户常用数据字典的查询使用方法。

 1 、用户

查看当前用户的缺省表空间

SQL > select  username,default\_tablespace  from  user\_users;

查看当前用户的角色

SQL > select   \*   from  user\_role\_privs;

查看当前用户的系统权限和表级权限

SQL > select   \*   from  user\_sys\_privs;

SQL > select   \*   from  user\_tab\_privs;

 2 、表

查看用户下所有的表

SQL > select   \*   from  user\_tables;

查看名称包含log字符的表

SQL > select   object\_name , object\_id   from  user\_objects

 where  instr( object\_name , ' LOG ' ) > 0 ;

查看某表的创建时间

SQL > select   object\_name ,created  from  user\_objects  where   object\_name = upper ( ' &table\_name ' );

查看某表的大小

SQL > select   sum (bytes) / ( 1024 \* 1024 )  as  "size(M)"  from  user\_segments

 where  segment\_name = upper ( ' &table\_name ' );

查看放在ORACLE的内存区里的表

SQL > select  table\_name,cache  from  user\_tables  where  instr(cache, ' Y ' ) > 0 ;

 3 、索引

查看索引个数和类别

SQL > select  index\_name,index\_type,table\_name  from  user\_indexes  order   by  table\_name;

查看索引被索引的字段

SQL > select   \*   from  user\_ind\_columns  where  index\_name = upper ( ' &index\_name ' );

查看索引的大小

SQL > select   sum (bytes) / ( 1024 \* 1024 )  as  "size(M)"  from  user\_segments

 where  segment\_name = upper ( ' &index\_name ' );

 4 、序列号

查看序列号，last\_number是当前值

SQL > select   \*   from  user\_sequences;

 5 、视图

查看视图的名称

SQL > select  view\_name  from  user\_views;

查看创建视图的select语句

SQL > set  view\_name,text\_length  from  user\_views;

SQL > set   long   2000 ; 说明：可以根据视图的text\_length值设定set  long  的大小

SQL > select   text   from  user\_views  where  view\_name = upper ( ' &view\_name ' );

 6 、同义词

查看同义词的名称

SQL > select   \*   from  user\_synonyms;

 7 、约束条件

查看某表的约束条件

SQL > select  constraint\_name, constraint\_type,search\_condition, r\_constraint\_name

 from  user\_constraints  where  table\_name  =   upper ( ' &table\_name ' );

SQL > select  c.constraint\_name,c.constraint\_type,cc.column\_name

 from  user\_constraints c,user\_cons\_columns cc

 where  c.owner  =   upper ( ' &table\_owner ' )  and  c.table\_name  =   upper ( ' &table\_name ' )

 and  c.owner  =  cc.owner  and  c.constraint\_name  =  cc.constraint\_name

 order   by  cc.position;

 8 、存储函数和过程

查看函数和过程的状态

SQL > select   object\_name ,status  from  user\_objects  where  object\_type = ' FUNCTION ' ;

SQL > select   object\_name ,status  from  user\_objects  where  object\_type = ' PROCEDURE ' ;

查看函数和过程的源代码

SQL > select   text   from  all\_source  where  owner = user   and  name = upper ( ' &plsql\_name ' );

三、查看数据库的SQL

 1 、查看表空间的名称及大小

 select  t.tablespace\_name,  round ( sum (bytes / ( 1024 \* 1024 )), 0 ) ts\_size

 from  dba\_tablespaces t, dba\_data\_files d

 where  t.tablespace\_name  =  d.tablespace\_name

 group   by  t.tablespace\_name;

 2 、查看表空间物理文件的名称及大小

 select  tablespace\_name,  file\_id ,  file\_name ,

 round (bytes / ( 1024 \* 1024 ), 0 ) total\_space

 from  dba\_data\_files

 order   by  tablespace\_name;

 3 、查看回滚段名称及大小

 select  segment\_name, tablespace\_name, r.status,

(initial\_extent / 1024 ) InitialExtent,(next\_extent / 1024 ) NextExtent,

max\_extents, v.curext CurExtent

 From  dba\_rollback\_segs r, v$rollstat v

 Where  r.segment\_id  =  v.usn( + )

 order   by  segment\_name ;

 4 、查看控制文件

 select  name  from  v$controlfile;

 5 、查看日志文件

 select  member  from  v$logfile;

 6 、查看表空间的使用情况

 select   sum (bytes) / ( 1024 \* 1024 )  as  free\_space,tablespace\_name

 from  dba\_free\_space

 group   by  tablespace\_name;

 SELECT  A.TABLESPACE\_NAME,A.BYTES TOTAL,B.BYTES USED, C.BYTES FREE,

(B.BYTES \* 100 ) / A.BYTES " %  USED",(C.BYTES \* 100 ) / A.BYTES " %  FREE"

 FROM  SYS.SM$TS\_AVAIL A,SYS.SM$TS\_USED B,SYS.SM$TS\_FREE C

 WHERE  A.TABLESPACE\_NAME = B.TABLESPACE\_NAME  AND  A.TABLESPACE\_NAME = C.TABLESPACE\_NAME;

 7 、查看数据库库对象

 select  owner, object\_type, status,  count ( \* )  count #  from  all\_objects  group   by  owner, object\_type, status;

 8 、查看数据库的版本

 Select  version  FROM  Product\_component\_version

 Where  SUBSTR(PRODUCT, 1 , 6 ) = ' Oracle ' ;

 9 、查看数据库的创建日期和归档方式

 Select  Created, Log\_Mode, Log\_Mode  From  V$ Database ;

 -- ------------------------------------------------------------------------------

jxdco

四、ORACLE用户连接的管理

用系统管理员，查看当前数据库有几个用户连接：

SQL >   select  username,sid,serial#  from  v$session;

如果要停某个连接用

SQL >   alter  system  kill  session  ' sid,serial# ' ;

如果这命令不行,找它UNIX的进程数

SQL >   select  pro.spid  from  v$session ses,v$process pro  where  ses.sid = 21   and  ses.paddr = pro.addr;

说明：21是某个连接的sid数

然后用  kill  命令杀此进程号。

五、SQL \* PLUS使用

a、近入SQL \* Plus

$sqlplus 用户名 / 密码

退出SQL \* Plus

SQL > exit

b、在sqlplus下得到帮助信息

列出全部SQL命令和SQL \* Plus命令

SQL > help

列出某个特定的命令的信息

SQL > help 命令名

c、显示表结构命令DESCRIBE

SQL > DESC  表名

d、SQL \* Plus中的编辑命令

显示SQL缓冲区命令

SQL > L

修改SQL命令

首先要将待改正行变为当前行

SQL > n

用CHANGE命令修改内容

SQL > c / 旧 / 新

重新确认是否已正确

SQL > L

使用INPUT命令可以在SQL缓冲区中增加一行或多行

SQL > i

SQL > 输入内容

e、调用外部系统编辑器

SQL > edit 文件名

可以使用DEFINE命令设置系统变量EDITOR来改变文本编辑器的类型，在login.sql文件中定义如下一行

DEFINE\_EDITOR = vi

f、运行命令文件

SQL > START test

SQL > @test

常用SQL \* Plus语句

a、表的创建、修改、删除

创建表的命令格式如下：

 create   table  表名 （列说明列表）；

为基表增加新列命令如下：

 ALTER   TABLE  表名  ADD  （列说明列表）

例：为test表增加一列Age，用来存放年龄

sql > alter   table  test

 add  （Age  number ( 3 )）；

修改基表列定义命令如下：

 ALTER   TABLE  表名

MODIFY （列名 数据类型）

例：将test表中的Count列宽度加长为10个字符

sql > alter  atble test

modify （County  char ( 10 )）；

b、将一张表删除语句的格式如下：

DORP  TABLE  表名；

例：表删除将同时删除表的数据和表的定义

sql > drop   table  test

c、表空间的创建、删除

六、ORACLE逻辑备份的SH文件

完全备份的SH文件：exp\_comp.sh

rq = ` date  + " % m % d" `

su  -  oracle  - c " exp  system / manager  full = y inctype = complete  file =/ oracle / export / db\_comp$rq.dmp"

累计备份的SH文件：exp\_cumu.sh

rq = ` date  + " % m % d" `

su  -  oracle  - c " exp  system / manager  full = y inctype = cumulative  file =/ oracle / export / db\_cumu$rq.dmp"

增量备份的SH文件: exp\_incr.sh

rq = ` date  + " % m % d" `

su  -  oracle  - c " exp  system / manager  full = y inctype = incremental  file =/ oracle / export / db\_incr$rq.dmp"

root用户crontab文件

 / var / spool / cron / crontabs / root增加以下内容

 0   2   1   \*   \*   / oracle / exp\_comp.sh

 30   2   \*   \*   0 - 5   / oracle / exp\_incr.sh

 45   2   \*   \*   6   / oracle / exp\_cumu.sh

当然这个时间表可以根据不同的需求来改变的，这只是一个例子。

七、ORACLE 常用的SQL语法和数据对象

一.数据控制语句 (DML) 部分

 1 . INSERT  (往数据表里插入记录的语句)

 INSERT   INTO  表名(字段名1, 字段名2, ……)  VALUES  ( 值1, 值2, ……);

 INSERT   INTO  表名(字段名1, 字段名2, ……)  SELECT  (字段名1, 字段名2, ……)  FROM  另外的表名;

字符串类型的字段值必须用单引号括起来, 例如: ’GOOD  DAY ’

如果字段值里包含单引号’ 需要进行字符串转换, 我们把它替换成两个单引号 '' .

字符串类型的字段值超过定义的长度会出错, 最好在插入前进行长度校验.

日期字段的字段值可以用当前数据库的系统时间SYSDATE, 精确到秒

或者用字符串转换成日期型函数TO\_DATE(‘ 2001 - 08 - 01 ’,’YYYY - MM - DD’)

TO\_DATE()还有很多种日期格式, 可以参看ORACLE DOC.

年 - 月 - 日 小时:分钟:秒 的格式YYYY - MM - DD HH24:MI:SS

INSERT时最大可操作的字符串长度小于等于4000个单字节, 如果要插入更长的字符串, 请考虑字段用CLOB类型,

方法借用ORACLE里自带的DBMS\_LOB程序包.

INSERT时如果要用到从1开始自动增长的序列号, 应该先建立一个序列号

 CREATE  SEQUENCE 序列号的名称 (最好是表名 + 序列号标记) INCREMENT  BY   1  START  WITH   1

MAXVALUE  99999  CYCLE NOCACHE;

其中最大的值按字段的长度来定, 如果定义的自动增长的序列号  NUMBER ( 6 ) , 最大值为999999

 INSERT  语句插入这个字段值为: 序列号的名称.NEXTVAL

 2 . DELETE  (删除数据表里记录的语句)

 DELETE  FROM表名  WHERE  条件;

注意：删除记录并不能释放ORACLE里被占用的数据块表空间. 它只把那些被删除的数据块标成unused.

如果确实要删除一个大表里的全部记录, 可以用  TRUNCATE  命令, 它可以释放占用的数据块表空间

 TRUNCATE   TABLE  表名;

此操作不可回退.

 3 . UPDATE  (修改数据表里记录的语句)

UPDATE表名  SET  字段名1 = 值1, 字段名2 = 值2, ……  WHERE  条件;

如果修改的值N没有赋值或定义时, 将把原来的记录内容清为NULL, 最好在修改前进行非空校验;

值N超过定义的长度会出错, 最好在插入前进行长度校验..

注意事项:

A. 以上SQL语句对表都加上了行级锁,

确认完成后, 必须加上事物处理结束的命令  COMMIT  才能正式生效,

否则改变不一定写入数据库里.

如果想撤回这些操作, 可以用命令  ROLLBACK  复原.

B. 在运行INSERT,  DELETE  和  UPDATE  语句前最好估算一下可能操作的记录范围,

应该把它限定在较小 (一万条记录) 范围内,. 否则ORACLE处理这个事物用到很大的回退段.

程序响应慢甚至失去响应. 如果记录数上十万以上这些操作, 可以把这些SQL语句分段分次完成,

其间加上COMMIT 确认事物处理.

 -- ------------------------------------------------------------------------------

jxdco

二.数据定义 (DDL) 部分

 1 . CREATE  (创建表, 索引, 视图, 同义词, 过程, 函数, 数据库链接等)

ORACLE常用的字段类型有

 CHAR  固定长度的字符串

 VARCHAR2  可变长度的字符串

 NUMBER (M,N) 数字型M是位数总长度, N是小数的长度

DATE 日期类型

创建表时要把较小的不为空的字段放在前面, 可能为空的字段放在后面

创建表时可以用中文的字段名, 但最好还是用英文的字段名

创建表时可以给字段加上默认值, 例如  DEFAULT  SYSDATE

这样每次插入和修改时, 不用程序操作这个字段都能得到动作的时间

创建表时可以给字段加上约束条件

例如 不允许重复  UNIQUE , 关键字  PRIMARY   KEY

 2 . ALTER  (改变表, 索引, 视图等)

改变表的名称

 ALTER   TABLE  表名1  TO  表名2;

在表的后面增加一个字段

 ALTER  TABLE表名  ADD  字段名 字段名描述;

修改表里字段的定义描述

 ALTER  TABLE表名 MODIFY字段名 字段名描述;

给表里的字段加上约束条件

 ALTER   TABLE  表名  ADD   CONSTRAINT  约束名  PRIMARY   KEY  (字段名);

 ALTER   TABLE  表名  ADD   CONSTRAINT  约束名  UNIQUE  (字段名);

把表放在或取出数据库的内存区

 ALTER   TABLE  表名 CACHE;

 ALTER   TABLE  表名 NOCACHE;

 3 . DROP  (删除表, 索引, 视图, 同义词, 过程, 函数, 数据库链接等)

删除表和它所有的约束条件

 DROP   TABLE  表名  CASCADE  CONSTRAINTS;

 4 . TRUNCATE  (清空表里的所有记录, 保留表的结构)

 TRUNCATE  表名;

三.查询语句 ( SELECT ) 部分

SELECT字段名1, 字段名2, ……  FROM  表名1,  [ 表名2, …… ]   WHERE  条件;

字段名可以带入函数

例如:  COUNT ( \* ),  MIN (字段名),  MAX (字段名),  AVG (字段名),  DISTINCT (字段名),

TO\_CHAR(DATE字段名, ' YYYY-MM-DD HH24:MI:SS ' )

NVL(EXPR1, EXPR2)函数

解释:

 IF  EXPR1 = NULL

 RETURN  EXPR2

 ELSE

 RETURN  EXPR1

DECODE(AA﹐V1﹐R1﹐V2﹐R2.)函数

解释:

 IF  AA = V1  THEN   RETURN  R1

 IF  AA = V2  THEN   RETURN  R2

..…

 -- ------------------------------------------------------------------------------

jxdco

 ELSE

 RETURN   NULL

LPAD(char1,n,char2)函数

解释:

字符char1按制定的位数n显示，不足的位数用char2字符串替换左边的空位

字段名之间可以进行算术运算

例如: (字段名1 \* 字段名1) / 3

查询语句可以嵌套

例如:  SELECT  ……  FROM

( SELECT  …… FROM表名1,  [ 表名2, …… ]   WHERE  条件)  WHERE  条件2;

两个查询语句的结果可以做集合操作

例如: 并集UNION(去掉重复记录), 并集UNION  ALL (不去掉重复记录), 差集MINUS, 交集INTERSECT

分组查询

SELECT字段名1, 字段名2, ……  FROM  表名1,  [ 表名2, …… ]   GROUP  BY字段名1

 [ HAVING 条件 ]  ;

两个以上表之间的连接查询

SELECT字段名1, 字段名2, ……  FROM  表名1,  [ 表名2, …… ]   WHERE

表名1.字段名  =  表名2. 字段名  [  AND …… ]  ;

SELECT字段名1, 字段名2, ……  FROM  表名1,  [ 表名2, …… ]   WHERE

表名1.字段名  =  表名2. 字段名( + )  [  AND …… ]  ;

有( + )号的字段位置自动补空值

查询结果集的排序操作, 默认的排序是升序ASC, 降序是DESC

SELECT字段名1, 字段名2, ……  FROM  表名1,  [ 表名2, …… ]

 ORDER  BY字段名1, 字段名2  DESC ;

字符串模糊比较的方法

INSTR(字段名, ‘字符串’) > 0

字段名  LIKE  ‘字符串 % ’  [ ‘%字符串%’ ]

每个表都有一个隐含的字段ROWID, 它标记着记录的唯一性.

四.ORACLE里常用的数据对象 ( SCHEMA )

 1 .索引 ( INDEX )

 CREATE   INDEX  索引名ON 表名 ( 字段1,  [ 字段2, …… ]  );

 ALTER   INDEX  索引名 REBUILD;

一个表的索引最好不要超过三个 (特殊的大表除外), 最好用单字段索引, 结合SQL语句的分析执行情况,

也可以建立多字段的组合索引和基于函数的索引

ORACLE8.1.7字符串可以索引的最大长度为1578 单字节

ORACLE8.0.6字符串可以索引的最大长度为758 单字节

 2 .视图 ( VIEW )

 CREATE   VIEW  视图名AS  SELECT  ….  FROM  …..;

 ALTER  VIEW视图名 COMPILE;

视图仅是一个SQL查询语句, 它可以把表之间复杂的关系简洁化.

 3 .同义词 (SYNONMY)

 CREATE  SYNONYM同义词名FOR 表名;

 CREATE  SYNONYM同义词名FOR 表名 @数据库链接名 ;

 4 .数据库链接 ( DATABASE  LINK)

 CREATE   DATABASE  LINK数据库链接名CONNECT  TO  用户名 IDENTIFIED  BY  密码 USING ‘数据库连接字符串’;

数据库连接字符串可以用NET8 EASY CONFIG或者直接修改TNSNAMES.ORA里定义.

数据库参数global\_name = true时要求数据库链接名称跟远端数据库名称一样

数据库全局名称可以用以下命令查出

 SELECT   \*   FROM  GLOBAL\_NAME;

查询远端数据库里的表

 SELECT  ……  FROM  表名 @数据库链接名 ;

五.权限管理 (DCL) 语句

 1 . GRANT  赋于权限

常用的系统权限集合有以下三个:

CONNECT(基本的连接), RESOURCE(程序开发), DBA(数据库管理)

常用的数据对象权限有以下五个:

 ALL   ON  数据对象名,  SELECT   ON  数据对象名,  UPDATE   ON  数据对象名,

 DELETE   ON  数据对象名,  INSERT   ON  数据对象名,  ALTER   ON  数据对象名

 GRANT  CONNECT, RESOURCE  TO  用户名;

 GRANT   SELECT   ON  表名  TO  用户名;

 GRANT   SELECT ,  INSERT ,  DELETE  ON表名  TO  用户名1, 用户名2;

 2 . REVOKE  回收权限

 REVOKE  CONNECT, RESOURCE  FROM  用户名;

 REVOKE   SELECT   ON  表名  FROM  用户名;

 REVOKE   SELECT ,  INSERT ,  DELETE  ON表名  FROM  用户名1, 用户名2;

查询数据库中第63号错误：

 select  orgaddr,destaddr  from  sm\_histable0116  where  error\_code = ' 63 ' ;

查询数据库中开户用户最大提交和最大下发数：  select  MSISDN,TCOS,OCOS  from  ms\_usertable；

查询数据库中各种错误代码的总和：

 select  error\_code, count ( \* )  from  sm\_histable0513  group   by  error\_code  order

 by  error\_code;

查询报表数据库中话单统计种类查询。

 select   sum (Successcount)  from  tbl\_MiddleMt0411  where  ServiceType2 = 111

 select   sum (successcount),servicetype  from  tbl\_middlemt0411  group   by  servicetype

以下转自:WWW.LOVEUNIX.COM MADE BY SDAWNYJ

1、查看表空间的名称及大小

select t.tablespace\_name, round(sum(bytes/(1024\*1024)),0) ts\_size

from dba\_tablespaces t, dba\_data\_files d

where t.tablespace\_name = d.tablespace\_name

group by t.tablespace\_name;

2、查看表空间物理文件的名称及大小

select tablespace\_name, file\_id, file\_name,

round(bytes/(1024\*1024),0) total\_space

from dba\_data\_files

order by tablespace\_name;

3、查看回滚段名称及大小

select segment\_name, tablespace\_name, r.status,

(initial\_extent/1024) InitialExtent,(next\_extent/1024) NextExtent,

max\_extents, v.curext CurExtent

From dba\_rollback\_segs r, v$rollstat v

Where r.segment\_id = v.usn(+)

order by segment\_name ;

4、查看控制文件

select name from v$controlfile;

5、查看日志文件

select member from v$logfile;

6、查看表空间的使用情况

select sum(bytes)/(1024\*1024) as free\_space,tablespace\_name

from dba\_free\_space

group by tablespace\_name;

SELECT A.TABLESPACE\_NAME,A.BYTES TOTAL,B.BYTES USED, C.BYTES FREE,

(B.BYTES\*100)/A.BYTES "% USED",(C.BYTES\*100)/A.BYTES "% FREE"

FROM SYS.SM$TS\_AVAIL A,SYS.SM$TS\_USED B,SYS.SM$TS\_FREE C

WHERE A.TABLESPACE\_NAME=B.TABLESPACE\_NAME AND A.TABLESPACE\_NAME=C.TABLESPACE\_NAME;

7、查看数据库库对象

select owner, object\_type, status, count(\*) count# from all\_objects group by owner, object\_type, status;

8、查看数据库的版本

Select version FROM Product\_component\_version

Where SUBSTR(PRODUCT,1,6)='Oracle';

9、查看数据库的创建日期和归档方式

Select Created, Log\_Mode, Log\_Mode From V$Database;

10、捕捉运行很久的SQL

column username format a12

column opname format a16

column progress format a8

select username,sid,opname,

round(sofar\*100 / totalwork,0) || '%' as progress,

time\_remaining,sql\_text

from v$session\_longops , v$sql

where time\_remaining <> 0

and sql\_address = address

and sql\_hash\_value = hash\_value

/

11。查看数据表的参数信息

SELECT partition\_name, high\_value, high\_value\_length, tablespace\_name,

pct\_free, pct\_used, ini\_trans, max\_trans, initial\_extent,

next\_extent, min\_extent, max\_extent, pct\_increase, FREELISTS,

freelist\_groups, LOGGING, BUFFER\_POOL, num\_rows, blocks,

empty\_blocks, avg\_space, chain\_cnt, avg\_row\_len, sample\_size,

last\_analyzed

FROM dba\_tab\_partitions

--WHERE table\_name = :tname AND table\_owner = :towner

ORDER BY partition\_position

12.查看还没提交的事务

select \* from v$locked\_object;

select \* from v$transaction;

13。查找object为哪些进程所用

select

p.spid,

s.sid,

s.serial# serial\_num,

s.username user\_name,

a.type object\_type,

s.osuser os\_user\_name,

a.owner,

a.object object\_name,

decode(sign(48 - command),

1,

to\_char(command), 'Action Code #' || to\_char(command) ) action,

p.program oracle\_process,

s.terminal terminal,

s.program program,

s.status session\_status

from v$session s, v$access a, v$process p

where s.paddr = p.addr and

s.type = 'USER' and

a.sid = s.sid and

a.object='SUBSCRIBER\_ATTR'

order by s.username, s.osuser

14。回滚段查看

select rownum, sys.dba\_rollback\_segs.segment\_name Name, v$rollstat.extents

Extents, v$rollstat.rssize Size\_in\_Bytes, v$rollstat.xacts XActs,

v$rollstat.gets Gets, v$rollstat.waits Waits, v$rollstat.writes Writes,

sys.dba\_rollback\_segs.status status from v$rollstat, sys.dba\_rollback\_segs,

v$rollname where v$rollname.name(+) = sys.dba\_rollback\_segs.segment\_name and

v$rollstat.usn (+) = v$rollname.usn order by rownum

15。耗资源的进程（top session）

select s.schemaname schema\_name, decode(sign(48 - command), 1,

to\_char(command), 'Action Code #' || to\_char(command) ) action, status

session\_status, s.osuser os\_user\_name, s.sid, p.spid , s.serial# serial\_num,

nvl(s.username, '[Oracle process]') user\_name, s.terminal terminal,

s.program program, st.value criteria\_value from v$sesstat st, v$session s , v$process p

where st.sid = s.sid and st.statistic# = to\_number('38') and ('ALL' = 'ALL'

or s.status = 'ALL') and p.addr = s.paddr order by st.value desc, p.spid asc, s.username asc, s.osuser asc

16。查看锁（lock）情况

select /\*\*//\*+ RULE \*/ ls.osuser os\_user\_name, ls.username user\_name,

decode(ls.type, 'RW', 'Row wait enqueue lock', 'TM', 'DML enqueue lock', 'TX',

'Transaction enqueue lock', 'UL', 'User supplied lock') lock\_type,

o.object\_name object, decode(ls.lmode, 1, null, 2, 'Row Share', 3,

'Row Exclusive', 4, 'Share', 5, 'Share Row Exclusive', 6, 'Exclusive', null)

lock\_mode, o.owner, ls.sid, ls.serial# serial\_num, ls.id1, ls.id2

from sys.dba\_objects o, ( select s.osuser, s.username, l.type,

l.lmode, s.sid, s.serial#, l.id1, l.id2 from v$session s,

v$lock l where s.sid = l.sid ) ls where o.object\_id = ls.id1 and o.owner

<> 'SYS' order by o.owner, o.object\_name

17。查看等待（wait）情况

SELECT v$waitstat.class, v$waitstat.count count, SUM(v$sysstat.value) sum\_value

FROM v$waitstat, v$sysstat WHERE v$sysstat.name IN ('db block gets',

'consistent gets') group by v$waitstat.class, v$waitstat.count

18。查看sga情况

SELECT NAME, BYTES FROM SYS.V\_$SGASTAT ORDER BY NAME ASC

19。查看catched object

SELECT owner, name, db\_link, namespace,

type, sharable\_mem, loads, executions,

locks, pins, kept FROM v$db\_object\_cache

20。查看V$SQLAREA

SELECT SQL\_TEXT, SHARABLE\_MEM, PERSISTENT\_MEM, RUNTIME\_MEM, SORTS,

VERSION\_COUNT, LOADED\_VERSIONS, OPEN\_VERSIONS, USERS\_OPENING, EXECUTIONS,

USERS\_EXECUTING, LOADS, FIRST\_LOAD\_TIME, INVALIDATIONS, PARSE\_CALLS, DISK\_READS,

BUFFER\_GETS, ROWS\_PROCESSED FROM V$SQLAREA

21。查看object分类数量

select decode (o.type#,1,'INDEX' , 2,'TABLE' , 3 , 'CLUSTER' , 4, 'VIEW' , 5 ,

'SYNONYM' , 6 , 'SEQUENCE' , 'OTHER' ) object\_type , count(\*) quantity from

sys.obj$ o where o.type# > 1 group by decode (o.type#,1,'INDEX' , 2,'TABLE' , 3

, 'CLUSTER' , 4, 'VIEW' , 5 , 'SYNONYM' , 6 , 'SEQUENCE' , 'OTHER' ) union select

'COLUMN' , count(\*) from sys.col$ union select 'DB LINK' , count(\*) from

22。按用户查看object种类

select u.name schema, sum(decode(o.type#, 1, 1, NULL)) indexes,

sum(decode(o.type#, 2, 1, NULL)) tables, sum(decode(o.type#, 3, 1, NULL))

clusters, sum(decode(o.type#, 4, 1, NULL)) views, sum(decode(o.type#, 5, 1,

NULL)) synonyms, sum(decode(o.type#, 6, 1, NULL)) sequences,

sum(decode(o.type#, 1, NULL, 2, NULL, 3, NULL, 4, NULL, 5, NULL, 6, NULL, 1))

others from sys.obj$ o, sys.user$ u where o.type# >= 1 and u.user# =

o.owner# and u.name <> 'PUBLIC' group by u.name order by

sys.link$ union select 'CONSTRAINT' , count(\*) from sys.con$

23。有关connection的相关信息

1）查看有哪些用户连接

select s.osuser os\_user\_name, decode(sign(48 - command), 1, to\_char(command),

'Action Code #' || to\_char(command) ) action, p.program oracle\_process,

status session\_status, s.terminal terminal, s.program program,

s.username user\_name, s.fixed\_table\_sequence activity\_meter, '' query,

0 memory, 0 max\_memory, 0 cpu\_usage, s.sid, s.serial# serial\_num

from v$session s, v$process p where s.paddr=p.addr and s.type = 'USER'

order by s.username, s.osuser

2）根据v.sid查看对应连接的资源占用等情况

select n.name,

v.value,

n.class,

n.statistic#

from v$statname n,

v$sesstat v

where v.sid = 71 and

v.statistic# = n.statistic#

order by n.class, n.statistic#

3）根据sid查看对应连接正在运行的sql

select /\*\*//\*+ PUSH\_SUBQ \*/

command\_type,

sql\_text,

sharable\_mem,

persistent\_mem,

runtime\_mem,

sorts,

version\_count,

loaded\_versions,

open\_versions,

users\_opening,

executions,

users\_executing,

loads,

first\_load\_time,

invalidations,

parse\_calls,

disk\_reads,

buffer\_gets,

rows\_processed,

sysdate start\_time,

sysdate finish\_time,

'>' || address sql\_address,

'N' status

from v$sqlarea

where address = (select sql\_address from v$session where sid = 71)

24．查询表空间使用情况select a.tablespace\_name "表空间名称",

100-round((nvl(b.bytes\_free,0)/a.bytes\_alloc)\*100,2) "占用率(%)",

round(a.bytes\_alloc/1024/1024,2) "容量(M)",

round(nvl(b.bytes\_free,0)/1024/1024,2) "空闲(M)",

round((a.bytes\_alloc-nvl(b.bytes\_free,0))/1024/1024,2) "使用(M)",

Largest "最大扩展段(M)",

to\_char(sysdate,'yyyy-mm-dd hh24:mi:ss') "采样时间"

from (select f.tablespace\_name,

sum(f.bytes) bytes\_alloc,

sum(decode(f.autoextensible,'YES',f.maxbytes,'NO',f.bytes)) maxbytes

from dba\_data\_files f

group by tablespace\_name) a,

(select f.tablespace\_name,

sum(f.bytes) bytes\_free

from dba\_free\_space f

group by tablespace\_name) b,

(select round(max(ff.length)\*16/1024,2) Largest,

ts.name tablespace\_name

from sys.fet$ ff, sys.file$ tf,sys.ts$ ts

where ts.ts#=ff.ts# and ff.file#=tf.relfile# and ts.ts#=tf.ts#

group by ts.name, tf.blocks) c

where a.tablespace\_name = b.tablespace\_name and a.tablespace\_name = c.tablespace\_name

25. 查询表空间的碎片程度

select tablespace\_name,count(tablespace\_name) from dba\_free\_space group by tablespace\_name

having count(tablespace\_name)>10;

alter tablespace name coalesce;

alter table name deallocate unused;

create or replace view ts\_blocks\_v as

select tablespace\_name,block\_id,bytes,blocks,'free space' segment\_name from dba\_free\_space

union all

select tablespace\_name,block\_id,bytes,blocks,segment\_name from dba\_extents;

select \* from ts\_blocks\_v;

select tablespace\_name,sum(bytes),max(bytes),count(block\_id) from dba\_free\_space

group by tablespace\_name;

26.查看有哪些实例在运行：

select \* from v$active\_instances;