Hardware

Hardware type

OS

OS version

OS 32 bit or 64bit
who command alternative
server reboot time

Tracing

Tracing system calls
How to Trace Unix System Calls (Doc ID 110888.1)

identifiy zombie process
Full details abou a PID

Memory

RAM SIZE

Real Memory used by a Oracle

Virtual memory usage

Top Memory Users

Clear cache memory in Linux

Disks

sort by file size
List Disks
Disk size
Disk Cloning between two servers
Create a dummy disk in solaris = 1GB
copying the disks

Disk Hardware errors

os space not release after deletion

Device statistics
freespace in zpool
Os used disks

CPU

Number of CPU's

CPU Usage
TOP CPU Users
Top 10 CPU consuming process
CPU & Memory used by a PID
top CPU Usage

SAR

CPU:

Memory:

Disk:

Network:

ZIP

Zipping a folder in solaris

Compress old files

compressing the folder

Reading files under .gz without uncompress

FIND/Replace

Display above & below 2 lines on grep

Finding a file

Find

Remove Old files

Replace a word in all the files under a directory

find all files containing specific text on Linux?

MAIL

Sending a File through mail

sending mail from a server

Network

No of open connection to db from a server

Unplumb the network interface

MTU value

Enable remote desktop in linux

vncserver

SCP @ background

rsync the files between servers

scp faster file transfer method (ssh pwd required) file copy to remote server without pwd

identify who occupied the port

Check the physical status of a NIC card

Others

iostat

Error log

^M Error

Tree command in solaris'

History

History command

who logged in

Timing at unix prompt

Time taken for a execution of a command

Check the package installed or not

Installation of rpm without dependency

How to check server is a global or local zone

To check the HBA ports are online or offline

" Tmp file too larg"   error in vi command

C program

c program to execute a shell script

display two files in parallel

difference between two folders including all files inside

difference of two files (indicated with '|'   symbol)

Find a file name contain a particular string

Screen   Command

Password encryption

Grep Command to display two lines above and below

Folder size including hidden folders

file occupied by whom

```
uname -M  /   prtconf
cat /sys/devices/virtual/dmi/id/*
uname -a
```

```
oslevel -r
cat /etc/release
cat /etc/redhat-release
```

```
uname -m
x86_64 ==> 64-bit kernel
i686   ==> 32-bit kernel
pinky
who -b
```

```
truss -p <spid>          (or)       truss -aedfo sqlplus.trc sqlplus /nolog
strace -p <spid>    -o output.txt
trace -p <spid>
ps -ecl |grep "Z"
pargs   11637
```

```
/usr/sbin/prtconf | grep Mem
dmesg
vmstat 5 5
swapinfo -tam
glance and m
lsdev -C | grep mem
lsattr -El mem0 ( output from previous command)
free -h, cat /proc/meminfo, dmesg | grep Memory
uerf -r 300 | grep -I mem
prstat -a
vmstat 1 5
/usr/ucb/ps vax |head
```

```
sync; echo 1> /proc/sys/vm/drop_caches
```

```
du -sh * | sort -h
iostat -IEn       (or)   echo|format
fdisk -l | egrep 'Disk.*bytes' | awk '{ sub(/,/,""); sum +=$3; print $2" "$3" "$4 } END { print "——————"; print "total:
```

http://serverfault.com/questions/4906/using-dd-for-disk-cloning

```
dd if=/dev/zero of=test.img bs=1024k count=1000
dd if=test.img of=test.img.bkp bs=1024000k
```

```
root@server # iostat -IEn|grep c0t600507680181050F20000000000009F2d0
c0t600507680181050F20000000000009F2d0 Soft Errors: 0 Hard Errors: 770 Transport Errors: 698
root@server #
$ /usr/sbin/lsof | grep deleted
ora    25575 data   33u   REG      65,65  4294983680   31014933 /oradata/dbname/something.dbf (deleted)
 $ file /proc/25575/fd/33
    /proc/25575/fd/33: broken symbolic link to `/oradata/DATAPRE/UNDOTBS009.dbf (deleted)'
     $ echo > /proc/25575/fd/33
     $ df -h .
```

```
iostat -xnp
zpool list
zpool status
```

Solaris --suppose virtual CPU
lscpu (or) cat /proc/cpuinfo|grep processor|wc -l
psrinfo -v|grep "Status of processor"|wc -l
lsdev -C|grep Process|wc -l
ioscan -C processor | grep processor | wc -l
kstat | grep -i core_id | uniq #psrinfo -p 1 #psrinfo |wc -l 64

```
sar -u 1 10
/usr/ucb/ps uax |head
date ; ps auwx | sort -r +2 | head -10
/usr/ucb/ps auxvv 11637
 ps -eo pcpu,pid,user,args | sort -k 1 -r | head -10
```

Basic CPU: sar [-u] [interval [count]] Load Average: sar -q [interval [count]]
Kernel Paging: sar -B [interval [count]] Unused Memory: sar -r [interval [count]] Swap Space: sar -S [interval [count]]
Average Disk I/O: sar -b [interval [count]] Disk I/O: sar -dp [interval [count]]
Network: sar -n DEV [interval [count]] Network Errors: sar -n EDEV [interval [count]]

```
zip -r archive_name.zip folder
```

```
gzip  logfile.log
tar -zcvf archive.tar.gz directory/
zcat filename  (or)   zhead  filname (or) zless filename
```

```
cat a | grep 5 -C 2
find /opt/tivoli/tsm/client -exec grep -l  "servername" {} \;

find / -print | grep -i dbmspool.sql
```

```
 find /opt/oracle/admin//cdmp* -mtime +1 -exec rm {} \;
#Verify it's using /home/oracle/oraInventory
find oraInventory -type f -exec grep oraInventory {} \; | head
#Replace old oraInventory strings in all files under oraInventory
find oraInventory -type f -exec perl -pi -e 's#/home/oracle/oraInventory#/u01/app/oracle/oraInventory#g' {} \;
#Verify it's using /u01/app/oracle/oraInventory
find oraInventory -type f -exec grep oraInventory {} \; | head
find . -type f | xargs grep dc7h92zrnpbnq
find . -type f -exec grep -l "dc7h92zrnpbnq" {} \;
```

```
uuencode file.txt file.txt | mailx testmail@domain.com
uuencode file.txt file.txt | mailx -s "this mail has attachments" testmail@domain.com
mailx -a textfile.txt testmailreceiver@domain.com
mailx -s "Test email  `hostname`"  testmail@domain.com  < /dev/null
```

```
lsof -i:1521 | grep ESTABLISHED | grep db_name| wc -l
ifconfig nxge2:2 unplumb
```

Ping all public and private nodename or IP of all nodes with corresponding MTU:

```
   /usr/sbin/ping -s nodename mtu 2
  Example in three node environment
   /usr/sbin/ping -s node1-pub 1500 2
   /usr/sbin/ping -s node2-pub 1500 2
   /usr/sbin/ping -s node3-pub 1500 2
   /usr/sbin/ping -s node1-priv 9000 2
   /usr/sbin/ping -s node2-priv 9000 2
   /usr/sbin/ping -s node3-priv 9000 2
```

```
Enable
------
gconftool-2 -s -t bool /desktop/gnome/remote_access/enabled true
gconftool-2 -s -t bool /desktop/gnome/remote_access/prompt_enabled false
Diable
-------
gconftool-2 -s -t bool /desktop/gnome/remote_access/enabled false
```

Then edit /opt/oracle/admin/.vnc/xstartup & uncomment the two lines

```
export
PATH=/:/bin:/usr/local/bin:/usr/ucb:/etc:/usr/sbin:/sbin:/usr/X11/bin:/usr/openwin/bin:/opt/sfw/bin:/bin:/OPatch:$PATH
vncserver
nohup scp -P 223 /tmp/myfile.dat user@server:/tmp &
rsync -avz --progress username@sourceserver:/path/to/file.dmp  .
scp -c arcfour  -r username@sourceserver:/path/to/file.dmp  .
```

```
# SOURCE:
 follow same as below - dont add location directly to dumpfile

cd /location
tar -cf - expdpmyfile.dmp | gzip -1 | nc -l 9999

# TARGET:
cd /thelocation/u/need
nc Sourceserver 9999 | gzip -d | tar xf - -C  .
netstat -anlp  | grep 1587
/sbin/ip link show

prstat -n 5 -s cpu
iostat -x 1 3

Linux: /var/log/messages, Solaris, HP Tru64: /var/adm/messages, HP-UX: /var/adm/syslog/syslog.log, AIX: /bin/err
dos2unix   filename filename
ls -R | grep ":$" | sed -e 's/:$//' -e 's/[^-][^\/]*\//--/g' -e 's/^/   /' -e 's/-/|/'

 ksh
HISTTIMEFORMAT="%d/%m/%y %T "
history
last | grep root | more


PS1="[\d \t \u@\h:\w ] $ "
Ex :   $ time dd if=test.img of=test.img.bkp bs=1M
dd: bad numeric argument: "1M"
real    0m0.01s
user    0m0.00s
sys     0m0.00s
pkginfo | grep  xvnc
rpm -Uvh  /opt/oracle/admin/redhat-release-6Server-1.noarch.rpm --nodeps

db/server: /dev/rdsk> zonename
Server1
Servernode2>  zonename
global

bash-3.2# luxadm -e port
/devices/pci@0/pci@0/pci@8/pci@0/pci@8/SUNW,emlxs@0/fp@0,0:devctl  NOT CONNECTED
/devices/pci@0/pci@0/pci@8/pci@0/pci@8/SUNW,emlxs@0,1/fp@0,0:devctl  NOT CONNECTED
/devices/pci@0/pci@0/pci@9/SUNW,emlxs@0/fp@0,0:devctl              NOT CONNECTED
/devices/pci@0/pci@0/pci@9/SUNW,emlxs@0,1/fp@0,0:devctl            NOT CONNECTED
stty columns 120
vi  file.log
```

```
$ cat a.c
#include <stdio.h>
#include <stdlib.h>

#define SHELLSCRIPT "\
ssh mainserver '/work/oracle/dba/bin/mygrid gridname'"


int main(void)
{
  system(SHELLSCRIPT);
  return 0;
}
$ cc a.c
$ ./a.out
```

```
paste c.txt d.txt | awk -F'\t' '{
    if (length($1)>max1) {max1=length($1)};
    col1[NR] = $1; col2[NR] = $2 }
    END {for (i = 1; i<=NR; i++) {printf ("%-*s     %s\n", max1, col1[i], col2[i])}
}'
```

```
diff -r   folder1 folder2
```

```
diff -y -W 170 c.txt d.txt
vimdiff file1 file2
sdiff -w 250 presnap.txt postsnap.txt  | more                 (Solaris )

find . -type f -exec grep -l "DBMS_STATS: GATHER_STATS_JOB: Stopped by Scheduler." {} \;
```

```
screen -S   Patching               (   Starting New session with Name incase of multiple screen session going to open)
screen -ls                         ( To list the existing screen sessions )
screen -r -d   screen_name         ( For attaching a existing screen name)
screen -r                           (   Attaching the existing screen automatically if only one session found )
screen -x Screen_name               (  Attaching Multiple display Screen )

CTRL ad                             (   dettaching  your screen without terminating )
```

```
 echo -e mypwd | base64

grep -A  2  FAILED   file_name
grep -B 2   FAILED  file_name
grep -C 2  FAILED  file_name

du -sch .[!.]* * |sort -h
```

```
/usr/sbin/lsof  +L1 /opt/oracle
```

AIX
Linux
Solaris

AIX
Solaris
linux

solaris,AIX
Linux
Tru 64

http://www.c0t0d0s0.org/archives/4778-Less-known-Solaris-fea

Solaris
HP/UX
AIX
Linux
DEC-UNIX

Solaris

svmon

```
# echo 1 > /proc/sys/vm/drop_caches
```

To free dentries and inodes:

Linux

http://serverfault.com/questions/4906/using-dd-for-disk-cloning solaris

psrinfo | wc -l
Linux
Solaris
AIX
HP/UX
Real CPU in Solaris

```
ps -e -o pcpu -o pid -o user -o args
topas
```

```
ps auxw | sort -r +2 |head -10
```

```
find /backup/logs/ -name daily_backup* -mtime +21 -exec rm -
```

http://yong321.freeshell.org/oranotes/MoveOraInventory.txt

```
 grep -rnw /location -e "dc7h92zrnpbnq"
```

solaris

works only in linux

pt
it will remove all CTRL ^ M Errors

fc -l -99

history -200

solaris

Note :   if the output is global then , the server is configured as global
        if the output is same server name then , the server is configured as local (virtual machine)

```
$ cat b.c
#include <stdlib.h>

int main() {
    system("/work/oracle/s/mygrid.sh");
    return 0;
}
$ cc b.c
$ ./a.out
```

https://stackoverflow.com/questions/13341832/display-two-files

Just to brief the changes use below command

```
diff --brief -r folder1 folder2
diff --old-group-format=$'\e[0;31m%<\e[0m' \
     --new-group-format=$'\e[0;31m%>\e[0m' \
     --unchanged-group-format=$'\e[0;32m%=\e[0m' \
       c.txt d.txt
```

display below 2 lines
display Above 2 lines
display above and below 2 lines

```
Solaris:  truss -d -E -p 1454
Linux  :  strace -ttT -p 5164
HP-UX  :  tusc
```

atures-Getting-rid-of-Zombies.html

```
prstat -a
```

```
getdev
cfgadm -la
prtconf
```

```
echo Cores = $(( $(lscpu | awk '/Socket/{ print $2 }') * $(lscpu | awk '/Core/{ print $4 }') ))
```

http://www.dba-oracle.com/t_tuning_cpu_usage_vmstat.htm

```
prstat -Z
```

Host CPU utilization report from DATABASE

Issue

command to delete files old trace file

```
oracle..> rm -rf  *.aud
ksh: /usr/bin/rm: arg list too long
```

ORA-09925: Unable to create audit trail file
Linux-x86_64 Error: 28: No space left on device (BUT SPACE Available)

Remove MGMT audit files

login to MGMT directory

ora-27102: out of memory

memory allocated to oracle user id in server

Tracing

Tracing Oracle process OS level
Tracing SQLPLUS at OS level

Others

Asking for oratab while logging in

List the databases running on server
Run Sql commands in terminal

Stopping the process
Killing clusterware processes

Killable processes

Instance-Critical: Non killable process

alias

SCRIPTS

Get the CPU/Memory Free Usage details from Database
script to extract db alias names from tnsnames.ora file

Os level scripting

Shell script to execute multiple database - 12c

cmd script for logging into multiple databases

OS prompt script

os level execute script

Profile for OS

ectract CRS_HOME from /etc/oratab

extract host/port/servicename from TNSPING

Execute the script after 4.5 hours (execute this in sqlplus)

Server maintenance

Now that you confirmed the process 16514 is consuming a lot of memory, you can "freeze" it—but not kill it—using the skill command.

Network Waitevent by database

```
col "00" for 99.9
col "01" for 99.9
col "02" for 99.9
col "03" for 99.9
col "04" for 99.9
col "05" for 99.9
col "06" for 99.9
```

Solution

```
find /opt/oracle/diag/rdbms/dbname/instance_name/trace -name "*.trc" -mtime +3 -exec rm {} \;
find /opt/oracle/diag/rdbms/dbname/instance_name/trace -name "*.trm" -mtime +3 -exec rm {} \;
find /opt/oracle/diag/rdbms/dbname/instance_name/trace -name "*.gz" -mtime +3 -exec rm {} \;
find /opt/oracle/diag/tnslsnr/hostname/listener_dbname/alert -name "log*xml" -mtime +3 -exec rm {} \;
(or)
 find . -name "*.aud" -mtime +7 -exec rm -f {} \;

find . -name '*.aud' -exec rm {} +
$ df -kh /u01
Filesystem            Size  Used Avail Use% Mounted on
/dev/mapper/VGExaDb-LVDbOra1
                       99G   82G   12G  88% /u01

$ df -h -i /u01
Filesystem            Inodes   IUsed   IFree IUse% Mounted on
/dev/mapper/VGExaDb-LVDbOra1
                         13M     13M       0  100% /u01
```

So delete old *.aud files from that location

```
rm: invalid option -- 'M'
Try `rm ./-MGMTDB_m000_10055_95.aud' to remove the file `-MGMTDB_m000_1.aud'.
Try `rm --help' for more information.
hostname$/opt/grid/12.1.0.2.PSUAPR2017/rdbms/audit>rm -- *.aud
hostname$:/opt/grid/12.1.0.2.PSUAPR2017/rdbms/audit>ls -ltr

[oracle@host01 _mgmtdb]$ cd -MGMTDB
-bash: cd: -M: invalid option
cd: usage: cd [-L|-P] [dir]
```

To get around that, I need to use “dot-slash” before the directory name.

```
[oracle@host01 _mgmtdb]$ cd ./-MGMTDB
[oracle@host01 -MGMTDB]$ cd trace
free -m
```

use this comman an see the cache memory .. It should be less… if it is more than 5Gb ask unix admin to lear the memory

```
prtconf | grep Mem
id -p
prctl -n project.max-shm-memory -i project 100


strace -fF -v -p 16311
strace /oracle/product/10.2.0.1/bin/sqlplus -V 2>&1 |less
```

```
grep "^[^#]" /var/opt/oracle/oratab |  awk -F: '{printf("\t%-6s\t%-30s\n",$1,$2)}'
. Oraenv
ps -ef|grep pmon | awk '{print $8}'|grep -v "grep"|  grep -v "ASM"| cut -d"_" -f3

ps -eaf | grep _pmon_ | grep -v grep | grep -v + | grep -v "^#" | awk '{print $NF}' | cut -c 10-
echo 'select count(*) from tab;' | sqlplus / as sysdba


ps -ef | grep ora_ | grep orcl2 | awk '{print $2}' | while read PID
do
kill -STOP $PID
done
ps -ef | grep <keyword> | grep -v grep | awk '{print $2}' | xargs kill -9
```

ARCn: Redo log archivers
CJQn: Job scheduler coordinator
Dnnn: Dispatchers
CKPT: Checkpoint
DBRM: Resource manager process
DBWn: Database writer processes
LGWR: Redo log writer

```
alias s="sqlplus / as sysdba"

SELECT
      STAT_NAME,
      DECODE(STAT_NAME,'PHYSICAL_MEMORY_BYTES',(ROUND(VALUE/1024/1024/1024,2))
      || ' GB','FREE_MEMORY_BYTES',(ROUND(VALUE           /1024/1024/1024,2))
      || ' GB',VALUE ) VALUE
    FROM
      v$osstat
    WHERE
     stat_name IN ( 'FREE_MEMORY_BYTES', 'LOAD', 'NUM_CPUS', 'NUM_CPU_CORES',
     'NUM_CPU_SOCKETS', 'PHYSICAL_MEMORY_BYTES' );
sort -u $TNS_ADMIN/tnsnames.ora | grep -v "^ " | grep -v "^#" | grep -v "^(" | grep -v "^)" | sed "s/ =//g" | grep -v "^$"


for i in `ps -ef|grep pmon | awk '{print $8}'|grep -v "grep"|  grep -v "ASM"| cut -d"_" -f3`
do
export ORACLE_SID=$i
sqlplus -s /nolog <<EOF
set head off echo off
conn / as sysdba
SELECT comp_name, version, status FROM dba_registry where status='INVALID';
exit
EOF
done
```

```
 #!/bin/ksh
rm -rf log.txt
for i in `cat a.txt`
do
echo ' ------'$i'----- ' >> /tmp/log.txt
sqlplus -L -S << EOF >> /tmp/log.txt
/ @$i
set feedback off
set lines 750 pages 0
col value for a20
select (select name from v\$database) dbname, (select NAME from v\$pdbs where con_id=(select con_id from v\$pdbs where rownum < 2)) con,i.host_name,p.inst_id,p.name,p.value from gv\$parameter p, gv\$instance i where p.name='global_names' and p.inst_id=i.inst_id  order by p.inst_id;
exit;
EOF
done
```

```
echo off
cls
for /f   "tokens=1-2 delims= " %%b in (1.txt) do (
   echo.***********************************************
   echo.Connect to %%b
rem    echo. Password: %%c
   echo.***********************************************
rem   echo.sqlplus -L sys/pwd@%%b @C:\dbname.sql
   sqlplus -L "sys/pwd@%%b as sysdba" @C:\dbname.sql

)
echo.
echo.DONE
echo.Press any key to exit.
pause >nul
```

```
  for i in `ps -ef |grep pmon|grep -v MGMTDB|grep -v ASM|grep -v grep|cut -d"_" -f3`
      do
      export ORACLE_SID=$i
      export ORACLE_HOME=`ps -ef |grep pmon|grep $i|awk {'print "pwdx " $2 '}|sh|awk {'print $2 '}|sed 's/....$//'`
      export PATH=$ORACLE_HOME/bin:$ORACLE_HOME/OPatch:$PATH
      sqlplus -s "/ as sysdba" <<EOF
      set lines 300
      col VALUE for a65
      select name, db_unique_name, open_mode,database_role from v\$database;
      show parameter log_archive_config
      EOF
      done

# ----------------------------------------------------------------------
# A simple kornshell script to force all running instances to re-register
# with the remote_listener (workaround for bug 13066936)
# ----------------------------------------------------------------------
for INST in `ps -aef | grep 'ora_pmon' | egrep -v '(grep|sed)' | sed 's/^.*ora_pmon_//'`
do
  echo . Reregistering $INST with remote_listener
  export ORAENV_ASK=NO
```

```
export ORACLE_HOME=/opt/oracle/product/11.1.0/db7
PATH=/usr/bin:/etc:/usr/sbin:/usr/ucb:$HOME/bin:/usr/bin/X11:/sbin:.:/usr/local/bin:${ORACLE_HOME}:${ORACLE_HOME}/bin:${ORACLE_HOME}/OPatch:/oracle/dba/bin

export PATH
export HISTFILE=$HOME/.histdir/$(tty|sed 's-/-_-g')
export DBA_INIT_INI=/oracle/dba/funcs/dba_init.ini

# NMON variable to stop oslevel checking
export NMONAIX=5.2.0.0

export EDITOR=vi
export FPATH=/oracle/dba/funcs
export TNS_ADMIN=/var/opt/oracle

export ORATAB="/var/opt/oracle/oratab"
export ORA_CRS_HOME=`cat /etc/oratab|grep -i asm|awk -F: '{print $2}'`

tnsping gpo_core_u |
awk '{FS="[()]+";for(i=1;i<=NF;i++) if($i ~ /(HOST|PORT|SERVICE_NAME)/) print $i}'

 script.sql which executes SQLT XTRACT with a delay of 4.5 hours
host sleep 4.5h
start run/sqltxtract.sql 51x6yr9ym5hdc sqltxplain
```

Presnap

```
cat > osprecheck.sh  <<\EOF
if [ `/bin/uname` = "SunOS" ]
then
      GREP=/usr/xpg4/bin/grep
   ECHO=echo
   DISK_LOC="/dev/asmdisks"
      export CRS_HOME=`(cat /var/opt/oracle/oratab  | grep ^+ASM | awk -F":" '{print $2}')`
   GG_HOME=`ps -ef | grep ./mgr | grep oracle | grep -v grep | awk '{ print $11 }'|cut -d "/" -f1-5`
else
      GREP=/bin/grep
   ECHO="echo -e"
   DISK_LOC="/dev/oracleasm/disks"


Freeze    =>   skill -STOP 1
Continue => skill -CONT 16514
skill -STOP oracle
skill -STOP rman

col c2 heading 'wait|class' format a20
col c3 heading 'time|waited' format 999,999,999,999

break on c1 skip 2

select
   trunc(end_interval_time) c1,
   wait_class c2,
```

Comments
https://weidongzhou.wordpress.com/2014/10/26/out-of-space-error-while-still-have-space/
for linux
Solaris

```
ps -eo comm| grep -v grep|grep pmon|sed s/ora_pmon_//
```

```
for i in `ps -ef|grep pmon | awk '{print $8}'|grep -v "grep"|  grep -v "ASM"| cut -d"_" -f3`
do
export ORACLE_SID=$i
sqlplus -s /nolog <<EOF
set head off echo off
conn / as sysdba
spool $i.sql
--select example script sql
select 'alter materialized view '||owner||'.'||object_name||' compile;'
from dba_objects
where status<>'VALID';
```

```
for i in `ps
-ef|grep
pmon |
awk
'{print
$8}'|grep -
v
"grep"|cut
-d"_" -f3`
do
export
ORACLE
```

```
for i in `ps -ef|grep pmon | awk '{print $8}'|grep -v "grep"|  grep -v "ASM"| cut -d"_" -f3`
do
export ORACLE_SID=$i
sqlplus -s /nolog <<EOF
set head off echo off feedback off
conn / as sysdba
set lines 160
select instance_name,logins,(select database_role from v\$database) database from v\$instance;
select PROCESS,status from gv\$managed_standby where process like '%MRP%';
col STBY_BEHIND_BY for a20
select (select sysdate from dual)"TIMESTAMP",s.thread#,s.MAX_SEQ#, a.MAX_APP_SEQ#,
(s.MAX_SEQ#-a.MAX_APP_SEQ#) "Difference" from
( select thread#,max(sequence#) "MAX_SEQ#" from v\$archived_log  group by thread# ) s,
(select thread#,max(sequence#) "MAX_APP_SEQ#" from v\$archived_log where applied='YES'
group by thread#) a where a.thread#=s.thread#;
set serveroutput on
exec dbms_output.put_line('------------------------------');
exit
EOF
done
#!/bin/ksh
rm -rf dbhealthoutput.txt
echo "Enter the syspassword"
stty -echo
read sys_pw
stty echo
for i in `cat dblist.txt`
do
   echo '*********************************************'
   echo connecting  to $i
   echo '*********************************************'

   echo '********************'     >>dbhealthoutput.txt
   echo connecting to $i               >>dbhealthoutput.txt
   echo '********************'     >>dbhealthoutput.txt
     sqlplus -L -S sys/$sys_pw@$i as sysdba @script.sql  >>dbhealthoutput.txt
   echo ' '  >>dbhealthoutput.txt
   echo
'*************************************************************************************************************
***********************************************************' >>dbhealthoutput.txt
   echo ' '  >>dbhealthoutput.txt
done

more dbhealthoutput.txt
```

postsnap

```
cat > ospostcheck.sh  <<\EOF
if [ `/bin/uname` = "SunOS" ]
then
      GREP=/usr/xpg4/bin/grep
   ECHO=echo
   DISK_LOC="/dev/asmdisks"
      export CRS_HOME=`(cat /var/opt/oracle/oratab  | grep ^+ASM | awk -F":" '{print $2}')`
   GG_HOME=`ps -ef | grep ./mgr | grep oracle | grep -v grep | awk '{ print $11 }'|cut -d "/" -f1-5`
else
      GREP=/bin/grep
   ECHO="echo -e"
   DISK_LOC="/dev/oracleasm/disks"
```

https://www.oracle.com/technetwork/articles/linux/part2-085179.html

You can use a user, a PID, a command or terminal id as argument. The following stops all rman commands
As you can see, skill decides that argument you entered—a process ID, userid, or command—and acts appropri

```
   for i in
`ps -ef
|grep
pmon|gre
p -v
MGMTDB
|grep -v
ASM|grep
-v
grep|cut -
d"_" -f3`
        do
```

ately. This may cause an issue in some cases, where you may have a user and a command in the same nam

e. The best example is the "oracle" process, which is typically run by the user "oracle". So, when you want to s

top the process called "oracle" and you issue:

Identify the Exadata Box type

cell versions

offloading saved

exadata resource manager plans

Exadata Serial Numbers

% Saved from exadata Storage

Session waiting for CPU - Resource manager

Dcli

http://www.centroid.com/blog/monitoring-exadata-smart-scan

```
grep -i MACHINETYPES  /opt/oracle.SupportTools/onecommand/databasemachine.xml
```

```
set lines 750 pages 9999
col CV_CELLNAME for a30
SELECT
  cellname cv_cellname
  , CAST(extract(xmltype(confval), '/cli-output/cell/releaseVersion/text()') AS VARCHAR2(20))
cv_cellVersion
  , CAST(extract(xmltype(confval), '/cli-output/cell/flashCacheMode/text()') AS
```

```
 select name, value  from v$statname join v$mystat using (statistic#)
   where name in
    ('cell physical IO bytes eligible for predicate offload',
     'cell physical IO interconnect bytes returned by smart scan',
      'cell physical IO bytes saved by storage index');
```

Scripts and Tips for Monitoring CPU Resource Manager (Doc ID 1338988.1)

```
/opt/oracle.SupportTools/CheckHWnFWProfile -S
```

```
select inst_id,sql_id,
 decode(IO_CELL_OFFLOAD_ELIGIBLE_BYTES,0,'No','Yes') Offloaded,
 decode(IO_CELL_OFFLOAD_ELIGIBLE_BYTES,0,0,
 100*(IO_CELL_OFFLOAD_ELIGIBLE_BYTES-
 IO_INTERCONNECT_BYTES)/
 IO_CELL_OFFLOAD_ELIGIBLE_BYTES) "IO_SAVED_%"
 from gv$sql
 where sql_id='&sql_id';
```

```
SELECT s.sid sess_id, g.name consumer_group,
s.state, s.consumed_cpu_time cpu_time, s.cpu_wait_time,
s.queued_time,(s.CURRENT_SMALL_READ_MEGABYTES+s.CURRENT_LARGE_READ_
MEGABYTES)
read_MB,(s.CURRENT_SMALL_WRITE_MEGABYTES+s.CURRENT_LARGE_WRITE_ME
GABYTES) write_mb
FROM v$rsrc_session_info s, v$rsrc_consumer_group g
WHERE s.current_consumer_group_id = g.id
```

```
dcli -l root -g dbs_group "ls -ltr / | grep -i acfs_u01"
```

HP => High Performance
HC => High Capcity

http://oradwstories.blogspot.com/2015/02/d

run as root

etecting-change-in-execution-plan-of.html

Nice url

Disk Details

Find Available disk to add to any diskgroup

Disk Group & disks Details

Disk group freespace & mounted or not

Disk group compatability

% used

DB Each Folder Utilization in MB in asm diskgroups

Each folder size in a diskgroup
asmdu script--Unix Script

ASM backup

Recreating disk sql's from metadata

Create Diskgroup Command

Dropping the Disk Group

Adding a disk

Copy an ASM file from a local ASM instance to a remote ASM instance

Formatting ASM disk header

Adding disks to disk group

Blocker in ASM

Kill all LOCAL connections

Removing all the files in ASM

Removing all archivelog Files from ASM

Files that are not in use currently under asm disks

Identify files in ASM not known to DB

Detecting disks using kfod

Check diskgroup usign kfod

Disk Size

Troubelshooting

Resizing disks

Masternode in CRS

Checking Disks

Database Cloning using kfed

undrop the diskgroup using kfed

Check the mirrored copies of a block in asm

Repair the Disk Corruption
Check do u have the backup block in the disk

Read a disk using od command (Alternative to strings command)

Show the dismounted diskgroups

ASM Disk I/O Time out issues

ASM diskgroup mount failure

password copy for 12c

ACFS file system not mounted

asmcmd privileged logins

ASM LIB

ASMLIB Device to disk mapping script

Oracleasm Lib diskgroup creation

unix kernal

asm module / Package

asm disks

discovery

config files

log

To Reload asm - module failure msg in /var/log/oracleasm http://satya-dba.blogspot.com/2010/03/automatic-storage-management-asm-10g.html

```
set pages 9999
column path format a20

select path, group_number group_#, disk_number disk_#, mount_status,
header_status, state, total_mb, free_mb

col PATH for a55
select path, group_number group_#, disk_number disk_#, mount_status,header_status, state,
total_mb, free_mb from v$asm_disk where header_status='CANDIDATE' or
header_status='FORMER' order by 1 ;
col PATH for a55
col DG_NAME for a15
col DG_STATE for a10
col FAILGROUP for a20
set lines 750 pages 9999


set pages 40000 lines 120
col NAME for a15
select GROUP_NUMBER DG#, name, ALLOCATION_UNIT_SIZE AU_SZ, STATE,
TYPE, TOTAL_MB, FREE_MB, OFFLINE_DISKS from v$asm_diskgroup;

 select GROUP_NUMBER,NAME,COMPATIBILITY,DATABASE_COMPATIBILITY from
v$asm_diskgroup;

COL % FORMAT 99.0
 SELECT name, free_mb, total_mb, ((total_mb-free_mb)/total_mb)*100 as "USED %",
free_mb/total_mb*100 "FREE%" from v$asm_diskgroup order by 1;

set pagesize 100
column inst format a10 heading 'Inst'
column file_type format a20 heading 'File type'
column mg format 99,999,999
break on inst skip 1

set pages 0 feed off  linesize 100 heading on
set pagesize 100
column inst format a10 heading 'Inst'
column file_type format a20 heading 'File type'
column mg format 99,999,999
break on inst skip 1
compute sum  of mg on inst

asmcmd > md_backup somefilename

asmcmd > md_backup myasmdiskbackup
asmcmd > md_restore -S mymdscript.sql myasmdiskbackup


CREATE DISKGROUP DATA_TEST EXTERNAL REDUNDANCY DISK '/dev/rdsk/c4t60050768018
CREATE DISKGROUP TEST EXTERNAL REDUNDANCY DISK '/dev/rdsk/c5t600507680181050F
```

```
DROP DISKGROUP TEST INCLUDING CONTENTS;

ALTER DISKGROUP DATA01 ADD DISK '/dev/oracleasm/disks/ASM_DISK01' SIZE 1000 M
rebalance power 10;
select * from v$asm_operation;
```

```
cp +DATA/orcl/datafile/tbsjfv.256.123456789
\sys@mydb . +ASM2 : +D2/jfv/tbsjfv.dbf
```

```
$DD if=/dev/zero skip=25 bs=4k count=2560 of=$OCRLOC

 ALTER DISKGROUP DATA_DBNAME ADD DISK
'/dev/rdsk/c6t6005076801808548600000000000000B2d0s0' SIZE 102400 M REBALANCE
POWER 6;
```

```
set linesize 200
set pagesize 1000

column username format a10
column module format a50
column blocker format a7
column waiter format a7
column lmode format 9999
column request format 9999
column inst_id format 9999
column sid format 9999
```

```
ps -eaf | grep ASM | grep beq | awk '{print "kill -9 " $2}'
```

```
SELECT 'ALTER DISKGROUP '||gname||' DROP FILE '''||full_path||''';' gsql FROM
(SELECT CONCAT('+'gname, SYS_CONNECT_BY_PATH(aname,'/')) full_path, gname FROM
(SELECT g.name gname, a.parent_index pindex, a.name aname,
a.reference_index rindex, a.ALIAS_DIRECTORY adir
FROM v$asm_alias a, v$asm_diskgroup g
WHERE a.group_number = g.group_number)
WHERE adir='N'
START WITH (MOD(pindex, POWER(2, 24))) = 0
CONNECT BY PRIOR rindex = pindex)
WHERE full_path LIKE UPPER('%&database%');
```

```
SELECT 'alter diskgroup '||dg.name||' drop file
''+'||dg.name||''||SYS_CONNECT_BY_PATH(al.name,'/')||''';'
FROM v$asm_alias al, v$asm_file fi, v$asm_diskgroup dg
WHERE al.file_number = fi.file_number(+)
AND al.group_number = dg.group_number
AND fi.type = 'ARCHIVELOG'
START WITH alias_index = 0
CONNECT BY PRIOR al.reference_index = al.parent_index;

set pagesize 0
set linesize 200
col full_alias_path format a80
/*+ ----------------------------------------------------------------
    Query will return all the files stored on ASM but not currenlty
    opened by any database client of the diskgroups
    ordered by group number, file type
    ----------------------------------------------------------------*/

select * from (
/*+ -----------------------------------------------------------------
    1st branch returns all the files stored on ASM
    -----------------------------------------------------------------*/
select  x.gnum,x.filnum,x.full_alias_path,f.ftype from (
SELECT gnum,filnum,concat('+'||gname, sys_connect_by_path(aname, '/')) full_alias_path
FROM (SELECT g.name gname, a.parent_index pindex, a.name aname,
            a.reference_index rindex,a.group_number gnum,a.file_number filnum
      FROM v$asm_alias a, v$asm_diskgroup g
      WHERE a.group_number = g.group_number)
START WITH (mod(pindex, power(2, 24))) = 0 CONNECT BY PRIOR rindex = pindex) x,
(select group_number gnum,file_number filnum, type ftype from v$asm_file order by
group_number,file_number) f
where x.filnum != 4294967295
and x.gnum=f.gnum and x.filnum=f.filnum
MINUS
/*+ --------------------------------------------------------------
    2nd branch returns all the files stored on ASM
    and currently opened by any database client of the diskgroups
    -----------------------------------------------------------------*/
select x.gnum,x.filnum,x.full_alias_path,f.ftype
from ( select id1 gnum,id2 filnum from v$lock where type='FA' and (lmode=4 or lmode=2)) l,
(
SELECT gnum,filnum,concat('+'||gname, sys_connect_by_path(aname, '/')) full_alias_path
```

```
select concat('+'||gname, sys_connect_by_path(aname, '/')) full_alias_path,
    system_created, alias_directory, file_type
from ( select b.name gname, a.parent_index pindex, a.name aname,
        a.reference_index rindex , a.system_created, a.alias_directory,
        c.type file_type
    from v$asm_alias a, v$asm_diskgroup b, v$asm_file c
    where a.group_number = b.group_number
        and a.group_number = c.group_number(+)
        and a.file_number = c.file_number(+)
        and a.file_incarnation = c.incarnation(+)
   )
where alias_directory = 'N'
start with (mod(pindex, power(2, 24))) = 0
       and rindex in
         ( select a.reference_index
           from v$asm_alias a, v$asm_diskgroup b
           where a.group_number = b.group_number
               and (mod(a.parent_index, power(2, 24))) = 0
               and a.name = '&DATABASENAME'
         )
connect by prior rindex = pindex;
```

```
export ORACLE_HOME=/tmp/OraInstall2014-01-30_01-59-24AM/ext/bin
export LD_LIBRARY_PATH=/tmp/OraInstall2014-01-30_01-59-24AM/ext/bin/../lib
cd /tmp/OraInstall2014-01-30_01-59-24AM/ext/bin
./kfod.bin verbose=true, disks=all status=true op=disks

kfod ds=true disks=all
kfod asm_diskstring='ORCL:*' disk=all
kfed read /dev/rdsk/c0d17s0 | egrep "dskname|dsksize"

truss -aefdD -o /tmp/kfod_'$hostname'.truss kfod asm_diskstring='/dev/rdsk/*'

SQL> alter diskgroup DATA resize all  size 1048288 M ;

SQL> alter diskgroup DATA resize disk DATA_0004 SIZE 1048288 M;


cd /opt/oracle/product/crs/log/wpsun564/cssd ; cat ocssd.log|grep -i 'master node'|tail -1

strings  /dev/rdsk/c0d17s0 | head -10
```

http://oraclehandson.wordpress.com/2010/08/06/cloning-oracle-asm-databases-with-kfed/

http://externaltable.blogspot.com/2013_12_01_archive.html

https://twiki.cern.ch/twiki/bin/view/PDBService/ASM_utilities

```
kfed repair /dev/rdsk/c0d17s0
kfed read /dev/rdsk/c6t600507680181050F2000000000000A61d0s0 aun=1 blkn=254 |grep KFBTY
```

```
od -c -N 128 /dev/rdsk/c6t60050768018E0279980000000000FA2d0s0
```

```
ASMCMD>  lsdg --discovery
```

```
alter system set "_asm_hbeatiowait"=200 scope=spfile sid='*';
```

And bounce the ASM instance

https://taliphakanozturken.wordpress.com/2013/03/31/mounting-asm-diskgroup-fails-with-ora-15063

```
srvctl config database -db PRODDB | grep -i pass
. oraenv  +ASM1
asmcmd --privilege sysdba pwcopy +DATA1/PRODDB/orapwPRODDB /tmp/orapwPRODDB
```

```
scp /tmp/orapwPRODDB userid@server.domain.com:/tmp
```

BCP:

```
. oraenv +ASM1
asmcmd --privilege sysdba pwcopy --dbuniquename STBYDB /tmp/orapwPRODDB
+DATA1/STBYDB/PASSWORD/orapwSTBYDB
```

(or)

```
ASMCMD> pwcopy /tmp/orapwPRODDB.295.999492555 +DATA1/STBYDB
```

```
. oraenv STBYDB
```

```
 ASMCMD> volinfo  --all
/opt/grid/12.1.0.2.PSUJAN2017/bin/acfsload start
```

incase if kernal fails then

```
/opt/grid/12.1.0.2.PSUJAN2017/bin/crsctl stop crs
/opt/grid/12.1.0.2.PSUJAN2017/bin/acfsroot install
/opt/grid/12.1.0.2.PSUJAN2017/bin/crsctl start crs
```

then try to start the filesystem if didn’t start automatically

```
srvctl start filesystem  -d /dev/asm/acf-somename-123
```

```
asmcmd --privilege sysdba -p
```

```
for asmlibdisk in  ls /dev/oracleasm/disks/*
do
  echo "ASMLIB disk name: $asmlibdisk"
  asmdisk=`kfed read $asmlibdisk | grep dskname | tr -s ' '| cut -f2 -d' '`
  echo "ASM disk name: $asmdisk"
  majorminor=`ls -l $asmlibdisk | tr -s ' ' | cut -f5,6 -d' '`
  device=`ls -l /dev | tr -s ' ' | grep "$majorminor" | cut -f10 -d' '`
  echo "Device path: /dev/$device"
done
```

```
/etc/init.d/oracleasm createdisk SYSTEMDG2 /dev/mapper/oradisk01p1
/etc/init.d/oracleasm createdisk DISK01 /dev/mapper/oradisk03p1
create diskgroup BKUP EXTERNAL REDUNDANCY disk 'ORCL:DISK01' SIZE 102398 M;

uname -r
lsmod | grep -i oracle
rpm -qa | grep oracleasm

ls -ltr /dev/oracleasm/disks | sort

/etc/init.d/oracleasm scandisks
/etc/init.d/oracleasm listdisks

cat /etc/sysconfig/oracleasm

tail -200f /var/log/oracleasm


/usr/sbin/oracleasm init
```

```
col path for a35
col Diskgroup for a15
col DiskName for a20
col disk# for 999
col total_mb for 999,999,999
col free_mb for 999,999,999
```

Good query

```
#!/bin/bash
ORANAME=ASM
DG=$1

DB=`ps -ef | grep smon | grep -v grep | grep $ORANAME | wc -l`

if [ "$DB" -eq 0 ]; then
```

recreating asm disk sql's are stored in mymdscript in the current directory…

```
3E0279980000000000010F6d0s0'  SIZE 102400M;
20000000000007D4d0s0' SIZE 102400 M DISK '/dev/rdsk/c5t600507680181050F20000000000007D5d0s0' SIZE
```

ASM Disks Offline Wi

3-error-and-resolving-with-kfed/

```
ASMCMD> pwcopy +DATA1/PRIMARY/PASSWORD/pwdprimary.1758.915635849
/home/oracle/orapwstdbydb
scp /home/oracle/orapwstdbydb standbyhost:/home/oracle
ASMCMD> pwcopy /home/oracle/orapwstdbydb +DATA1/STDBYDB/orapwstdbdy
$ srvctl modify database –d STDBYDB –pwfile +DATA1/STDBYDB/orapwstdbdy
$ srvctl config database -d STDBYDB
```

https://emilianofusaglia.net/2016/04/08/troubleshooting-not-mounting-acfs-file-system/

http://surachartopun.com/2009/09/asm-dvm-acfs-by-command-lines.html

http://www.idevelopment.info/data/Oracle/DBA_tips/Automatic_Storage_Management/ASM_50.shtml

cat /etc/sysconfig/oracleasm

```
/etc/init.d/oracleasm querydisk -d `/etc/init.d/oracleasm listdisks -d` | \
cut -f2,10,11 -d" " | \
perl -pe 's/"(.*)".*\[(.*), *(.*)\]/$1 $2 $3/g;' | \
while read v_asmdisk v_minor v_major
do
v_device=`ls -la /dev | grep " $v_minor, *$v_major " | awk '{print $10}'`
echo "ASM disk $v_asmdisk based on /dev/$v_device [$v_minor, $v_major]"
done
```

http://asmsupportguy.blogspot.com/201 https://ronr

```
/etc/init.d/oracleasm disable
/etc/init.d/oracleasm enable
```

```
102400 M DISK '/dev/rdsk/c5t600507680181050F20000000000007D6d0s0' SIZE 102400 M
```

RAC Enabled in ORACLE_HOME or not

How to disable RAC on Oracle_home

Capture resource definition

Node Eviction Demo

Killing clusterware processes

Verifying the JUMBO Frames

Verify the jumbo frames using MTU 9000 Value

Network connectivity check by OSW

Tracing RAC resources

Trace troubleshoot a RAC CRS error

Modifying the cluster services

Formatting OCR/Voting Disks

Diag Collections

Verify all the setting

Kill all CRS process

Switching RAC Home/Patching/Upgrade

How to skip mgmtdb on 12.1.0.2

Tracing cluvfy

crs resource script

crs resource creation script

database running on all RAC nodes

alternative to crsstat command (only for linux)

Grid all databases list

Grid all databases list with nodename

service name for the databases

Starting ACFS File system

If root.sh fails

RAC DB Load balancing across Instances

Ignore error till empty line found (or)
print output from empty line found
print output till empty line

Role based service check

```
/usr/ccs/bin/ar -t $ORACLE_HOME/rdbms/lib/libknlopt.a|grep kcsm.o
```

```
cd $ORACLE_HOME/rdbms/lib
make -f ins_rdbms.mk rac_off --Disabling RAC binaries
make -f ins_rdbms.mk ioracle --Linking to Oracle binaries
```

```
for res in `$ORA_CRS_HOME/bin/crs_stat -p | grep "^NAME=" | cut -d
= -f 2` ; do
   $ORA_CRS_HOME/bin/crs_stat -p $res
>/opt/oracle/resources/$res.cap
done
```

```
oracle@rac2:~/ [+ASM2] ps -ef | grep cssd | grep -v grep
root 3583 1 0 11:45 ? 00:00:17 /u00/app/grid/11.2.0.2/bin/cssdmonitor
root 3595 1 0 11:45 ? 00:00:16 /u00/app/grid/11.2.0.2/bin/cssdagent
oracle 3607 1 1 11:45 ? 00:04:22 /u00/app/grid/11.2.0.2/bin/ocssd.bin
```

```
oracle@rac2:~/ [+ASM2] chrt -p 3595
pid 3595's current scheduling policy: SCHED_RR
pid 3595's current scheduling priority: 99

root@rac1 ~]# kill -STOP 3595; sleep 27; kill -CONT 3814
```

```
ps -ef | grep <keyword> | grep -v grep | awk '{print $2}' | xargs kill -9
```

```
traceroute -F node02-priv 9000

/usr/sbin/traceroute -s node1-priv node2-priv 9000

traceroute -r -F 192.168.0.210
```

```
export SRVM_TRACE=true
export SRVM_TRACE=""
```

```
root> export SRVM_TRACE=true
root> cd $CRS_HOME/bin
root> ./cluvfy comp crs -n
```

This will generate a trace file in $CRS_HOME/cv/log

```
$GRID_HOME/bin/crsctl modify resource ora.net1.network -attr "USR_C
```

```
$DD if=/dev/zero skip=25 bs=4k count=2560 of=$OCRLOC
```

```
[root@server1 tmp]# cd /tmp/myfolder
[root@server1 tmp]# export GRID_HOME=/opt/crs/12.1.0/grid
[root@server1 tmp]# export ORA_CRS_HOME=/opt/crs/12.1.0/grid
[root@server1 tmp]# perl /opt/crs/12.1.0/grid/bin/diagcollection.pl --
collect -all --incidenttime 06/12/201415:00:00 --incidentduration 01:00
[root@server1 tmp]# chown oracle:dba *
[root@server1 tmp]# tar -zcvf  server1038diag.tar.gz myfolder
```

How to Validate Network and Name Resolution Setup for the Clusterwar

```
ps -ef | grep <keyword> | grep -v grep | awk '{print $2}' | xargs kill -9
$NEW_GRID_HOME/oui/bin/runInstaller -updateNodeList
ORACLE_HOME=<$NEW_GRID_HOME> CRS="false" –local
```

Make sure that only the current GI home in the inventory has “CRS=TRUE” flag on ALL nodes.
Ask OSE to run the following four command as root as given in the order given.

```
cp -pR  $OLD_GRID_HOME/cdata $NEW_GRID_HOME (Ex. cp -pR
/opt/grid/11.2.0.4.x /cdata opt/grid/11.2.0.4.y where x<y)
cp -pR  $OLD_GRID_HOME/gpnp $NEW_GRID_HOME (Ex. cp -pR
/opt/grid/11.2.0.4.x /gpnp opt/grid/11.2.0.4.y where x<y)
cp -p $OLD_GRID_HOME/crs/install/s_crsconfig_<hostname>_env.txt
$NEW_GRID_HOME/crs/install
```

adding -J-Doracle.install.mgmtDB=false on the installer command line tc

```
rm -rf /tmp/cvutrace
mkdir /tmp/cvutrace
export CV_TRACELOC=/tmp/cvutrace
export SRVM_TRACE=true
export SRVM_TRACE_LEVEL=1
<STAGE_AREA>/runcluvfy.sh stage -pre crsinst -n <node1>,<node2> -
verbose
```

```
for res in `/opt/crs/12.1.0/grid/bin/crs_stat -p | grep "^NAME=" | cut -d = -
f 2` ; do
/opt/crs/12.1.0/grid/bin/crs_stat -p $res >/opt/oracle/resources/$res.cap
done
```

http://dbaregistry.blogspot.com/search/label/How%20to%20export%20a

```
export ORACLE_HOME=`cat /etc/oratab | grep -i +ASM | cut -d: -f2`
export ORACLE_SID=`cat /etc/oratab | grep -i +ASM | cut -d: -f1`
export PATH=$ORACLE_HOME/bin:$PATH
rm -rf /tmp/tmp.log*
for i in `olsnodes`
do
chmod 777 chek.sh > /dev/null
/usr/bin/scp -o UserKnownHostsFile=/dev/null -o
StrictHostKeyChecking=no -q chek.sh $i:/tmp > /dev/null
done

for i in `olsnodes`
do
/usr/bin/ssh -o UserKnownHostsFile=/dev/null -o
StrictHostKeyChecking=no -q $i "chmod 777 /tmp/chek.sh" > /dev/null
/usr/bin/ssh -o UserKnownHostsFile=/dev/null -o
StrictHostKeyChecking=no -q $i "/tmp/chek.sh" > /tmp/tmp.log_$i
done
paste /tmp/tmp.log* | sed -e 's/\t/ \t/g' | column -t -s$'\t'

rm -rf /tmp/tmp.log*
```

```
crsctl status res |grep -v "^$"|awk -F "=" 'BEGIN {print " "}
{printf("%s",NR%4 ? $2"|" : $2"\n")}'|sed -e 's/  *, /,/g' -e 's/, /,/g'|\
awk -F "|" 'BEGIN { printf "%-40s%-35s%-20s%-50s\n","Resource
Name","Resource Type","Target ","State" }{ split ($3,trg,",") split
($4,st,",")}{for (i in trg) {printf "%-40s%-35s%-20s%-
50s\n",$1,$2,trg[i],st[i]}}'
```

```
for i in `olsnodes` ; do /usr/bin/ssh -q $i "ps -aef | grep pmon |awk '{print
```

```
for i in `olsnodes`
do
echo $i > /tmp/tmp.log_$i
/usr/bin/ssh -q $i " echo "================" ; ps -aef | grep pmon
|awk '{print $8}' | egrep -i -v 'grep' | cut -d "_" -f3" | sort >>
/tmp/tmp.log_$i
done

paste /tmp/tmp.log* | sed -e 's/\t/ \t/g' | column -t -s$'\t' ; rm -rf
/tmp/tmp.log*
```

```
awk -F ":" '{if ($1) print "srvctl status service -d "$1""}'
/var/opt/oracle/oratab |grep -v '#\|+' |xargs -0 bash -c |grep -v 'PRC'
```

As root, execute the below.

```
/opt/grid/12.1.0.2/bin/acfsload start
```

As oracle,

```
srvctl start filesystem  -d /dev/asm/acf-somename-123
srvctl start filesystem  -d /dev/asm/acf_ggtrlv1-108
srvctl start filesystem  -d /dev/asm/acf_repl1-108
```

```
[root rac2] /u01/app/11.2.0.3/grid/crs/install# /usr/bin/chuser capabilities=
```

```
WITH sys_time AS (
    SELECT inst_id, SUM(CASE stat_name WHEN 'DB time'
                        THEN VALUE END) db_time,
            SUM(CASE WHEN stat_name IN ('DB CPU', 'background cpu
time')
                THEN  VALUE  END) cpu_time
      FROM gv$sys_time_model
     GROUP BY inst_id                  )
SELECT instance_name,
       ROUND(db_time/1000000,2) db_time_secs,
       ROUND(db_time*100/SUM(db_time) over(),2) db_time_pct,
       ROUND(cpu_time/1000000,2) cpu_time_secs,
       ROUND(cpu_time*100/SUM(cpu_time) over(),2)  cpu_time_pct
  FROM  sys_time
  JOIN gv$instance USING (inst_id);
```

```
olsnodes | sed -n '/^$/,/^$/p' |  awk '{if (NR==1 && NF==0) next};1'
olsnodes | sed -n '/^$/,/^$/!p'
```

```
crsctl stat res ora.dbname.service_name.svc -p | egrep '(USR_ORA_OF
```

Result returns 1 then RAC enabled. If 0 then not enabled

http://www.pythian.com/blog/changing-hostnames-in-oracle-rac/

Ping: With ping we have to take into account an overhead of about 28 bytes per packet, so 8972 bytes go through with no errors, while 8973 fail, this is a correct configuration that supports a message of up to 9000 bytes with no fragmentation:

```
[node01]$ ping -c 2 -M do -s 8972 node02-priv
PING node02-priv (10.10.10.2) 1472(1500) bytes of data.
1480 bytes from node02-priv (10.10.10.2): icmp_seq=0 ttl=64
time=0.220 ms
1480 bytes from node02-priv (10.10.10.2): icmp_seq=1 ttl=64
time=0.197 ms

[node01]$ ping -c 2 -M do -s 8973 node02-priv
From node02-priv (10.10.10.1) icmp_seq=0 Frag needed and DF set
```

http://www.vmcd.org/2012/02/rec

RA_IF=vnet0 vnet2"

e and RAC (Doc ID 1054902.1)

http://www.pythian.com/blog/slimming-down-oracle-rac-12cs-resource-footprint/

(Doc ID 986822.1)

http://dbaregistry.blogspot.com/search/label/How%20to%20export%20and%20import%20crs%20resourc nd%20import%20crs%20resources

```
$8}' | egrep -i -v 'grep' | cut -d "_" -f3" | sort > /tmp/tmp.log_$i ; done ; paste /tmp/tmp.log* | column -t
```

```
crsctl status resource -t|grep -v ora.mgmtdb|grep -v grep|grep ora.*.db|cut -f2 -d.|
while read line; do echo -e "\033[0;31mDatabase Name: \033[0;0m"$line
echo -e "\033[0;31mServices: \033[0;0m"`srvctl config database -d $line|grep Services|awk '{print $2}'`;
echo -e "\033[0;31mServers: \033[0;0m"`srvctl status database -d $line|awk '{printf "%s.",$7}'`;
done
```

https://emilianofusaglia.net/2016/04/08/troubleshooting-not-mounting-acfs-file-system/

http://mahioracledba.blogspot.com/2014/01/rootsh-failed-when-installing-11gr2.html

useful when executing asm commands other than oracle user

Primary --> primary role
physical_standby --> standby role

ommendation-for-the-real-application-cluster-interconnect-and-jumbo-frames/comment-page-1/#comment-450

2

Log shipping status? Errors

How much transferred & applied - GAP Detection

Gaps

General health of dataguard &errors

Standby DB Name
Last applied archive log

```
LAST PRODUCTION SEQUENCE GENERATED - prod
```

```
LAST SEQUENCE RECIEVED FROM PRODUCTION
and APPLIED ON STANDBY
```

MRP is in WAIT_FOR_LOG in gv$managed_standby

DGMGRL

which files are applies - Max files

Equivalent Broker Commands to 'ALTER SYSTEM'

Delete OLD applied archive logs on standby database

dgmgrl logging
dgmgrl
Error: ORA-16664: unable to receive the result from a database
DGMGRL all errors & troubleshoot
dgmgrl log location

dataguard switchover validation

database is primary or standby

log_archive_dest_2

is MRP waiting for something ?

MRP apply speed

How to increase MRP apply speed

how long (Minutes) will it take to complete the gap

Dataguard switchover :

restoring missing archive log file

OS Level DGMGRL check

OS level GAP check for all dbs in a host

DG Build

standby duplicate using RMAN backupset

Restore Archivelog Backups taken on the Primary Database to the Standby Database

Standby Realtime apply enabled or not (run this on stby)

Full details of the DG db details from dgmgrl

```
 select distinct error from gv$archive_dest ;
set lines 160
col STBY_BEHIND_BY for a20
select (select sysdate from dual)"TIMESTAMP",s.thread#,s.MAX_SEQ#, a.MAX_APP_SEQ#, '
'||(s.MAX_SEQ#-a.MAX_APP_SEQ#)||' sequences' "STBY_BEHIND_BY" from
( select thread#,max(sequence#) "MAX_SEQ#" from v$archived_log where activation#=(select activation#
from v$database) group by thread# ) s,
(select thread#,max(sequence#) "MAX_APP_SEQ#" from v$archived_log where activation#=(select
activation# from v$database )and applied='YES' group by thread#) a where a.thread#=s.thread#;

select * from v$archive_gap;

-------------------
set pages 1000
set lines 120
column DEST_NAME format a20
column DESTINATION format a35
column ARCHIVER format a10
column TARGET format a15
column status format a10
column error format a15
select DEST_ID,DEST_NAME,DESTINATION,TARGET,STATUS,ERROR from v$archive_dest
where DESTINATION is NOT NULL
select (select name from v$database) PRIMARY ,DB_UNIQUE_NAME STANDBY_NAME from
v$archive_dest where TARGET='STANDBY' and status='VALID';
select thread#,max(sequence#) from v$log_history group by thread#;
select to_char(sysdate,'dd-mon-yyyy hh24:mi:ss') Date from dual;
SELECT distinct SEQUENCE# "Last Sequence Generated", THREAD# "Thread"
FROM V$ARCHIVED_LOG
WHERE (THREAD#,FIRST_TIME ) IN (SELECT THREAD#,MAX(FIRST_TIME) FROM V$ARCHIVED_LOG
GROUP BY THREAD#)
ORDER BY 1;
SELECT ARCH.THREAD# "Thread", ARCH.SEQUENCE# "Last Sequence Received", APPL.SEQUENCE#
"Last Sequence Applied", (ARCH.SEQUENCE# - APPL.SEQUENCE#) "Difference"
FROM
(SELECT THREAD# ,SEQUENCE# FROM V$ARCHIVED_LOG WHERE (THREAD#,FIRST_TIME ) IN
(SELECT THREAD#,MAX(FIRST_TIME) FROM V$ARCHIVED_LOG GROUP BY THREAD#)) ARCH,
(SELECT THREAD# ,SEQUENCE# FROM V$LOG_HISTORY WHERE (THREAD#,FIRST_TIME ) IN
(SELECT THREAD#,MAX(FIRST_TIME) FROM V$LOG_HISTORY GROUP BY THREAD#)) APPL
WHERE
ARCH.THREAD# = APPL.THREAD#;


SQL> select * from gv$managed_standby where process like '%MRP%';
SQL> recover managed standby database cancel  ;
Media recovery complete.
SQL> recover managed standby database using current logfile disconnect ;
Media recovery complete.
SQL> select * from gv$managed_standby where process like '%MRP%';

show configuration
show database verbose wdbname
show database wdbname InconsistentProperties
edit database mydbname set property  LogFileNameConvert='+DATA01/dbprimary/ , +DATA01/dbstandby/' ;
```

Standby_Side

```
select max(sequence#) from v$archived_log where applied='YES';
select thread#, max(sequence#) from v$archived_log where applied='YES' group by thread#;
```

```
delete force noprompt ARCHIVELOG from sequence 17101 until sequence 20340 thread=1;
delete force noprompt ARCHIVELOG from sequence 16151 until sequence 19560 thread=2;
```

```
SQL> alter database recover managed standby database cancel;
DGMGRL> edit database 'stby_dbname' set state='LOG-APPLY-OFF';

SQL> alter database recover managed standby database disconnect;
DGMGRL> edit database 'stby_dbname' set state='ONLINE';

SQL> alter system set log_archive_max_processes=4;
DGMGRL> edit database 'dbname' set property 'LogArchiveMaxProcesses'=4;

SQL> alter system set log_archive_dest_state_2='enable' scope=both;
DGMGRL> edit database 'stby_dbname' set property 'LogShipping'='ON';

SQL> alter system set log_archive_dest_state_2='defer' scope=both;
DGMGRL> edit database 'stby_dbname' set property 'LogShipping'='OFF';

DGMGRL> edit database 'pri_dbname' set state='LOG-TRANSPORT-OFF';
```

This will defer all standby databases

```
select 'delete archivelog until sequence '||max(sequence#)||' thread '||thread#||' ; '
from gv$archived_log
where applied='YES'
group by thread#
order by 1
/
```

```
dgmgrl -logfile observer.log / "start observer"
```

copy the password file from prod to standby

https://www.toadworld.com/platforms/oracle/w/wiki/11143.monitoring-troubleshooting-data-guard-using-broker

same as alert log location --> file name   drcp<sid>.log

https://www.pythian.com/blog/database-validation-in-data-guard-12c/

```
 select database_role from v$database;
```

On Primary database the value of controlfile_type in V$database is “CURRENT” and standby is “STANDBY”

```
SQL> SELECT controlfile_type FROM V$database;

alter system set log_archive_dest_1 = 'location=use_db_recovery_file_dest valid_for=(all_logfiles, all_roles)
db_unique_name=primarydbname';
alter system set log_archive_dest_2='service="stbydbname",LGWR ASYNC NOAFFIRM delay=0 optional
compression=disable max_failure=0 max_connections=1 reopen=300 valid_for=(all_logfiles,primary_role)
db_unique_name="stbydbname"';

select a.event, a.wait_time, a.seconds_in_wait from gv$session_wait a, gv$session b where a.sid=b.sid and
b.sid=(select SID from v$session where PADDR=(select PADDR from v$bgprocess where NAME='MRP0'))
set lines 1000 pages 9999
select to_char(START_TIME,'DD-MON-YYYY HH24:MI:SS') "Recovery Start Time",to_char(item)||' =
'||to_char(sofar)||' '||to_char(units) "Progress"
from v$recovery_progress where start_time=(select max(start_time) from v$recovery_progress);
SESSION;

DGMGRL> show database STBY maxconnections;
MaxConnections = '8'

SQL>  show parameter log_archive_max_processes
NAME                                 TYPE        VALUE
------------------------------------ ----------- ------------------------------
log_archive_max_processes            integer     8

DGMGRL> edit database STBY set property maxconnections=30;
Property "maxconnections" updated
prod ( Total MB to be applied )
-----------------------------------
select nvl(a.MB,0)+nvl(b.MB,0)+nvl(c.MB,0)+nvl(d.MB,0)  from
(select sum(round(blocks*block_size/1024/1024)) MB from v$archived_log  where  dest_id=1 and thread#=1
and sequence#
between (select max(sequence#) "MAX_APP_SEQ#" from v$archived_log where  applied='YES' and
thread#=1 )
and     (select max(sequence#) "MAX_SEQ#" from v$archived_log where thread#=1)) a ,
(select sum(round(blocks*block_size/1024/1024)) MB from v$archived_log  where  dest_id=1 and thread#=2
and sequence#
between (select max(sequence#) "MAX_APP_SEQ#" from v$archived_log where  applied='YES' and
thread#=2 )
and     (select max(sequence#) "MAX_SEQ#" from v$archived_log where thread#=2)) b ,
(select sum(round(blocks*block_size/1024/1024)) MB from v$archived_log  where  dest_id=1 and thread#=3
and sequence#
between (select max(sequence#) "MAX_APP_SEQ#" from v$archived_log where  applied='YES' and
thread#=3 )
```

execute this on primary

```
ALTER DATABASE COMMIT TO SWITCHOVER TO PHYSICAL STANDBY WITH SESSION SHUTDOWN;
```

```
run
{
set archivelog destination to '/ora_backup/rman/arch/';
restore archivelog from logseq=8619 until logseq=8632 thread=2;
}
```

```
for i in `ps -ef|grep pmon | awk '{print $8}'|grep -v "grep"|  grep -v "ASM"| cut -d"_" -f3`
do
export ORACLE_SID=$i
dgmgrl  -silent / "show configuration verbose" ; echo '------------------------------------'
done
for i in `ps -ef|grep pmon | awk '{print $8}'|grep -v "grep"|  grep -v "ASM"| cut -d"_" -f3`
do
export ORACLE_SID=$i
sqlplus -s /nolog <<EOF
set head off echo off feedback off
conn / as sysdba
set lines 160
select instance_name,logins,(select database_role from v\$database) database from v\$instance;
select PROCESS,status from gv\$managed_standby where process like '%MRP%';
col STBY_BEHIND_BY for a20
select (select sysdate from dual)"TIMESTAMP",s.thread#,s.MAX_SEQ#, a.MAX_APP_SEQ#, (s.MAX_SEQ#-
a.MAX_APP_SEQ#) "Difference" from
( select thread#,max(sequence#) "MAX_SEQ#" from v\$archived_log  group by thread# ) s,
(select thread#,max(sequence#) "MAX_APP_SEQ#" from v\$archived_log where applied='YES' group by
thread#) a where a.thread#=s.thread#;
set serveroutput on
exec dbms_output.put_line('------------------------------');
exit
EOF
done
```

How to create a physical Standby database with DG Broker

```
----------------------------------------------------
primary   : india
secondary : america


at primary
-----------
select name, force_logging from v$database;
NAME      FOR
--------- ---
india   NO

ALTER DATABASE FORCE LOGGING;

 set lines 750 pages 9999
 select * from v$logfile;
 select group#,thread#,bytes/1024/1024 from v$log;
 select group#,dbid,sequence#,status from v$standby_log;

 --------------------------------------------------------------- ---
        1        ONLINE  +DATA_DISK/india/onlinelog/group_1.563.952942141
        1        ONLINE  +FLASH_DISK/india/onlinelog/group_1.563.952942141
        2        ONLINE  +DATA_DISK/india/onlinelog/group_2.258.952942141
        2        ONLINE  +FLASH_DISK/india/onlinelog/group_2.258.952942141
        3        ONLINE  +DATA_DISK/india/onlinelog/group_3.259.952942141
        3        ONLINE  +FLASH_DISK/india/onlinelog/group_3.259.952942141
        4        ONLINE  +DATA_DISK/india/onlinelog/group_4.260.952942141
        4        ONLINE  +FLASH_DISK/india/onlinelog/group_4.260.952942141

alter database add standby logfile THREAD 1 group 5  ('+DATA_DISK','+FLASH_DISK') SIZE 500M;
alter database add standby logfile THREAD 1 group 6 ('+DATA_DISK','+FLASH_DISK') SIZE 500M;

RMAN>
echo set on
 run {
  allocate channel c1 device type disk;
  allocate auxiliary channel a1 device type disk;
  allocate auxiliary channel a2 device type disk;
  duplicate database for standby from active database using backupset nofilenamecheck;
}

$rman target sys/pass@<connect_string> catalog rcvcat/rcvcat@<connect_string> auxiliary /

run {
 set archivelog destination to '/data/archivelog';
 restore clone archivelog from sequence 55 until sequence 56;
}

select DEST_ID,dest_name,status,type,srl,recovery_mode from v$archive_dest_status where dest_id=1;
```

```
dgmgrl >  validate database verbose db_name
```

http://sanzdbaworld.blogspot.com/2012/09/how-query-dataguard-status-physical.ht

onprimary

```
--------------
col STBY_BEHIND_BY for a20
select (select sysdate from dual)"TIMESTAMP",s.thread#,s.MAX_SEQ#,
a.MAX_APP_SEQ#, (s.MAX_SEQ#-a.MAX_APP_SEQ#) "Difference" from
( select thread#,max(sequence#) "MAX_SEQ#" from v$archived_log  group by
thread# ) s,
(select thread#,max(sequence#) "MAX_APP_SEQ#" from v$archived_log where

select sequence#,ARCHIVED AS RECEIVED,applied from v$archived_log where a
```

The status should be APPLYING_LOG

to check inconsistency if any errors

```
run{
crosscheck archivelog all;
delete noprompt expired archivelog all;
delete noprompt archivelog all completed before 'sysdate - 1/24';
}
```

http://www.ritzyblogs.com/OraTalk/PostID/105/How-to-setup-DGMGRL-broker-with

<-- execute this in RMAN

The DB_UNIQUE_NAME must be specified right after the service name:

```
alter system set log_archive_dest_2 = 'SERVICE="PROD_STDBY", LGWR
ASYNC VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE)
DB_UNIQUE_NAME=PROD_STDBY';
```

if you want to increase the transport speed, increase the value in primary database also .

https://www.toadworld.com/platforms/oracle/w/wiki/11334.let-s-optimize-data-guard-configuration-I

http://emrebaransel.blogspot.com/2008/11/dataguard-performance.html

```
for i in `ps -ef|grep pmon | awk '{print $9}'|grep -v "grep"|  grep -v "ASM"| cut -d"_" -
f3`
do
export ORACLE_SID=$i
dgmgrl  -silent / "show configuration verbose" ; echo '------------------------------------'
done
```

```
 run {
   allocate channel prmy1 type disk;
   allocate channel prmy2 type disk;
   allocate channel prmy3 type disk;
   allocate channel prmy4 type disk;
   allocate auxiliary channel stby1 type disk;
   allocate auxiliary channel stby2 type disk;
   allocate auxiliary channel stby3 type disk;
   allocate auxiliary channel stby4 type disk;
   duplicate target database for standby
   from active database section size 10G dorecover;
}
```

ml

```
onprimary
--------------
select a.thread#, (select max (sequence#) from v$archived_log
where archived='YES' and thread#=a.thread#) archived,
 max(a.sequence#) applied, (select max(sequence#) from
v$archived_log where archived='YES' and thread#=a.thread#)-
max(a.sequence#) gap
 from v$archived_log a where a.applied='YES' group by a.thread#

applied='NO' order by sequence#;
```

Sessions

Total user count on database

Users Details Session

v$sql_monitor

Sql-Monitor report for a sql_id (Like OEM report)

SQL Monitor report for sql_id for a specific snap_id

Real-Time SQL Monitoring - AWR(FULL_CAPTURE)

Sid Level Debug

Long Running query - long ops

Wait event for a SID

RealTime Monitoring for sid

Sql details

Currently running SQL query's

RealTime Monitoring for sql_id

SQL History

Top 50 sql's where running at a particular snapshot time

Identify

statistics of objects of a specific sql id

PGA/TEMP Used by a SQL_ID

How to check the Run time changes of
sql execution Plans due to adaptive 12c features

Sql id current rows, I/O , read/wrire details

Check the progress of DML statements

Detailed history of SQL_ID

Sql Hanging or not

SQL to show the full SQL executing for active sessions

Worst and best PLAN_HASH_VALUE calulator for a sql_id

wait events

Database level Wait ratios/Events

Events Drilled down %

wait event for past 5 mins / 30 mins/ 1 hour

wait events for a sql_id

Wait event for objects in active session history

Explain Plan

explain plan for manual session

explain plan for sql id present/history

Process

Finding SID for a PID

Finding OS pid for a db sid

Finding db sid from OS spid

Finding own sid & serial number

Killing Sessions

Killing Own session

kill a sid

Killing old session (1 day old)   (or)   4 hours

OS level killing old session for database

Determine If Killed Session Is Rolling Back Transactions

Extra

CPU/REDO/PGA/READ/WRITE used by a SID

Cursor Value for a SID (All Sql query executed by SID)

Maximum no of session in the database
Cheking all sql from a user

wait event currently- ON CPU

command to purge explain plan for a specific program...

Enable Parallelism in session level

parallelism Hints

Sql_text that is rolling back

Uncommited transactions in the database

SQL Profiles & Baselines

sql patch

Sql Profiles

Sql Baselines

extract hint from all profile/baseline/patch - single script

Fixing the PLAN HASH VALUE

purge/flush the old plan hash value from memory

flush a cursor /sql_id from shared_pool'

Sql profile All in All

Sql_doctor script

AAS - Average active session from history/AWR

aas-per-min.sql

aas-per-hour.sql (AWR)

Unstable Plans/ More than one Plan Hash Value PHV

Forensic Analysis of sql

History of all sql between two sample times

Min and Max execution time for a sql_id based on PHV

Optimizer details for a sql_id (DBA_HIST_SQL_STAT)

explain plan for the sqlid with specific PHV

explain pan for PHV (dba_hist_sql_plan)

History of histogram changes

Automatic tuning advice for a specified SQL_ID

```
set lines 750 pages 9999
break on report
compute SUM of tot on report
compute SUM of active on report
compute SUM of inactive on report
col username for a50
select DECODE(username,NULL,'INTERNAL',USERNAME)
Username,
       count(*) TOT,
```

```
set linesize 750 pages 9999
column box format a30
column spid format a10
column username format a30
column program format a30
```

```
column sid format 9999
column serial for 999999
column status format a15
column username format a10
column sql_text format a80
```

```
set lines 750 pages 9999
set long 20000 longchunksize 20000
select
 dbms_sqltune.report_sql_monitor_list() text_line
from dual;
```

specific snap_id

```
EXTRACTVALUE(XMLType(report_summary),'/report_repository_summary/sql/@sql_id') sql_id,

EXTRACTVALUE(XMLType(report_summary),'/report_repository_sum
```

```
DBMS_AUTO_REPORT.START_REPORT_CAPTURE;
DBMS_AUTO_REPORT.FINISH_REPORT_CAPTURE;
```

gv$session

```
set linesize 750 pages 9999
column box format a30
column spid format a10
column username format a20
column program format a30
```

```
SELECT
a.sid,RPAD(a.opname,30),a.sofar,a.totalwork,a.ELAPSED_SECONDS
,ROUND(((a.sofar)*100)/a.totalwork,3) "%_COMPLETED",
RPAD(a.username,10) username,a.SQL_HASH_VALUE,B.STATUS
FROM gV$SESSION_LONGOPS a, gv$session b
```

Currently waiting ?

```
 col WAIT_CLASS for a10
SELECT sw.inst_id,NVL(s.username, '(oracle)') AS username,
       s.sid,
       s.serial#,
       sw.event,
```

Elapsed/CPU/Read/Write MB

```
SELECT *
     FROM
       (SELECT status,
         --username,
         sql_id,
```

v$sqlarea/v$sql   (ALL)

```
set lines 1000 pages 9999
column sid format 9999
column username format a15
column PARSING_SCHEMA_NAME format a15
column SQL_EXEC_START for a21
```

v$sqlarea/v$sql   (sql_id)

```
set lines 1000 pages 9999
COL instance_number FOR 9999 HEA 'Inst';
COL end_time HEA 'End Time';
COL plan_hash_value HEA 'Plan|Hash Value';
col sql_id for a15
```

Sql_monitor report

```
set pagesize 0 echo off timing off linesize 1000 trimspool on trim on
long 2000000 longchunksize 2000000
select
DBMS_SQLTUNE.REPORT_SQL_MONITOR(
   sql_id=>'&sql_id',
```

All in one

```
set lines 1000 pages 9999
COL instance_number FOR 9999 HEA 'Inst';
COL start_time for a20
COL plan_hash_value HEA 'Plan|Hash Value';
col sql_id for a15
```

```
COL instance_number FOR 9999 HEA 'Inst';
COL end_time HEA 'End Time';
COL plan_hash_value HEA 'Plan|Hash Value';
col sql_id for a15
COL executions_total FOR 999,999 HEA 'Execs|Total';
```

 TOP CPU Used

```
COL instance_number FOR 9999 HEA 'Inst';
COL end_time HEA 'End Time';
COL plan_hash_value HEA 'Plan|Hash Value';
col sql_id for a15
col SQL_PROFILE for a30
```

sql_id

```
col sql_text for a50
select sql_id,to_char (LAST_LOAD_TIME, 'dd-mon-yyyy hh24:mi:ss')
LAST_LOAD_TIME, exact_matching_signature, SQL_TEXT from
v$sqlarea where UPPER(sql_text) like '%DUMMY%' order by
UPPER(sql_text);
col table_name for a40
col owner for a30
select distinct owner, table_name, STALE_STATS, last_analyzed,
stattype_locked
```

PGA/TEMP (current)

```
      starting_time,
      end_time,
 (EXTRACT(HOUR FROM run_time) * 3600
                    + EXTRACT(MINUTE FROM run_time) * 60
sql_adaptive_plan_resolved
, COUNT(DISTINCT sql_exec_id), COUNT(*)
FROM dba_hist_active_Sess_history
WHERE sql_id = '4dszd9dysry0c'
select
SQL_ID,PLAN_HASH_VALUE,ELAPSED_TIME,PHYSICAL_READ_B
YTES,PHYSICAL_WRITE_BYTES,DISK_READS,DIRECT_WRITES,B
UFFER_GETS,ROWS_PROCESSED from v$sqlstats where
sql_id='&sql_id';
col sql_text for a60
SELECT rows_processed "Total Rows Processed",
       ROUND((SYSDATE - TO_DATE(first_load_time, 'yyyy-mm-dd
hh24:mi:ss')) * 24 * 60, 1) "Total Time (Min)",
       TRUNC(rows_processed /((SYSDATE -
TO_DATE(first_load_time, 'yyyy-mm-dd hh24:mi:ss')) * 24 * 60))
  select instance_number
inst_id,SESSION_ID,USER_ID,PROGRAM,sql_id,SQL_CHILD_NUMB
ER,sql_plan_hash_value,to_char (sql_exec_start, 'dd-Mon-yyyy
hh24:mi:ss') sql_exec_start from
      dba_hist_active_sess_history   where
select sess_io.inst_id,
       sess_io.sid,
    sesion.sql_id,
       sess_io.block_gets,
where sesion.sql_hash_value = sqltext.hash_value   and
sesion.sql_address    = sqltext.address   and sesion.username is not


 WITH snaps
     AS (SELECT /*+  materialize */
               dbid, SNAP_ID
           FROM dba_hist_snapshot s
          WHERE (begin_interval_time BETWEEN sysdate-&1 AND
sysdate))
select * from (
SELECT t.*, row_number () over (order by impact_secs desc ) seq#
FROM (
```

Wait Ratio

```
SELECT METRIC_NAME, VALUE
  FROM V$SYSMETRIC
  WHERE METRIC_NAME IN ('Database CPU Time Ratio',
  'Database Wait Time Ratio') AND
  INTSIZE_CSEC =
  (select max(INTSIZE_CSEC) from V$SYSMETRIC);
```

```
SELECT NVL(a.event, 'ON CPU') AS event,
       COUNT(*) AS total_wait_time
FROM   v$active_session_history a
WHERE  a.sample_time > SYSDATE - 5/(24*60) -- 5 mins
GROUP BY a.event
ORDER BY total_wait_time DESC;
```

```
select
    event,
    time_waited "time_waited(s)",
    case when time_waited = 0 then
        0
```

```
set lines 750 pages 9999
col OBJECT_NAME for a35
select * from (
SELECT
```

```
" @?/rdbms/admin/utlxplp.sql
(or)
 SELECT * FROM TABLE(DBMS_XPLAN.DISPLAY(",",'+COST
+BYTES -PREDICATE'));
```

```
SELECT * FROM
TABLE(DBMS_XPLAN.DISPLAY_CURSOR('&sql_id',&childnumber,'A
LLSTATS LAST +PEEKED_BINDS +PROJECTION +ALIAS
+OUTLINE +PREDICATE +COST +BYTES'));

(OR)

select * from
table(dbms_xplan.display_cursor('&sql_id',&childnumber,'ADVANCED')
);

(OR)

set pagesize 0 echo off timing off linesize 1000 trimspool on trim on
long 2000000 longchunksize 2000000
select
DBMS_SQLTUNE.REPORT_SQL_MONITOR(
   sql_id=>'&sql_id',
   report_level=>'ALL',
   type=>'TEXT')
from dual;
```

```
select spid,p.pid, s.sid, s.serial#, p.program from v$session s,
v$process p where paddr=addr and p.pid=30849 order by p.pid;

select spid "host-pid",p.pid, s.sid, s.serial#, p.program, s.machine from
gv$session s, gv$process p where paddr=addr and s.sid=&sid order by
p.pid;
select s.sid, s.serial#, s.username,
       to_char(s.logon_time,'DD-MON HH24:MI:SS') logon_time,
       p.pid oraclepid, p.spid "ServerPID", s.process "ClientPID",
       s.program clientprogram, s.module, s.machine, s.osuser,
select sys_context ( 'USERENV' , 'SID' ) OwnSID from dual;
select sys_context('USERENV','SESSION_USER')
current_user,sys_context('USERENV','SESSION_SCHEMA')
current_schema from dual;
```

```
alter session set events 'immediate crash';
```

```
select 'alter system kill session ' || '''' || sid || ',' || serial# ||',@'|| inst_id ||
'''' || ' immediate;' from gv$session where sid='&sid';
```

```
select ' alter system kill session '''||sid||''','''||serial#||''',''@'||inst_id||'''
immediate; '
from gv$session where username in ('SCHEMA1','SCHEMA2') and
logon_time < sysdate -1 and status='INACTIVE';
```

```
ps -ef | grep "oracleinstname (LOCAL=NO)"
```

```
SELECT a.sid, a.username, b.xidusn rollback_seg_no,
b.used_urec undo_records, b.used_ublk undo_blocks
FROM gv$session a, gv$transaction b
WHERE a.saddr = b.ses_addr;
```

```
SELECT Logon_time,
        (SELECT ROUND (VALUE / 1024 / 1024, 2)
           FROM v$sesstat
SELECT o.inst_id,o.sid,
       o.sql_text,
       o.address,
       o.hash_value,
       o.user_name,
       s.schemaname,
       o.sql_id
  FROM gv$open_cursor o, gv$session s
 WHERE o.saddr = s.saddr AND o.sid = s.sid AND (O.SID = &sid)
 and o.inst_id=s.inst_id;
"ID",B.BEGIN_INTERVAL_TIME,B.END_INTERVAL_TIME,A.RESOU
RCE_NAME,
CURRENT_UTILIZATION "CURRENT",MAX_UTILIZATION "MAX"
FROM WRH$_RESOURCE_LIMIT A, WRM$_SNAPSHOT B
WHERE A.RESOURCE_NAME LIKE '%session%'
AND A.SNAP_ID=B.SNAP_ID
SELECT SQL_ID, SQL_FULLTEXT, PLAN_HASH_VALUE, PARSING_
```

```
select
       count(*),
       CASE WHEN state != 'WAITING' THEN 'WORKING'
            ELSE 'WAITING'
       END AS state,
```

node

```
select inst_id, sql_id, address, hash_value, plan_hash_value, sql_text
from gv$sqlarea where sql_text like 'SELECT COUNT(*) FROM
V$SESSION A WHERE A.AUDSID IN%';
select * from gv$sqlarea where sql_text like '%XLA_AE_LINES_GT%';
select inst_id, sql_id, address, hash_value, plan_hash_value, sql_text
from gv$sqlarea where sql_id = 'amd3xmgb8cnuz';
-- On each INST_ID, do these:
alter session set events '5614566 trace name context forever';
exec sys.dbms_shared_pool.purge('00000006999BD648,
1622326877','c');
alter session force parallel query;

alter session enable parallel DML;
/*+ PARALLEL */
/*+ PARALLEL, 8 */
/*+ NOPARALLEL */
select sql_text
from v$sqlarea
where address = (select sql_address from v$session where sid in
(select SID
from v$transaction,dba_rollback_segs,v$session
where SADDR=SES_ADDR and
XIDUSN=SEGMENT_ID and
flag=7811));
```

```
select xidusn, xidsqn from v$transaction;
```

```
declare
   v_sql CLOB;
begin
   select sql_text into v_sql from dba_hist_sqltext where
sql_id='5273fz2cqkk80';
   sys.dbms_sqldiag_internal.i_create_patch(
      sql_text  => v_sql,
      hint_text => 'DYNAMIC_SAMPLING(4)',
      name      => '5273fz2cqkk80_patch');
end;
```

check

```
  select NAME,SIGNATURE ,STATUS,FORCE_MATCHING from dba_s
```

check

```
select SQL_HANDLE, PLAN_NAME, ENABLED, ACCEPTED, FIXED
,sql_text     from dba_sql_plan_baselines;
```

```
select * from
table(dbms_xplan.display_sql_plan_baseline(plan_name=>'SQL_PLA
N_b7pnq6yp8da6a29d0d9b7'));
```

All in one script

```
set linesize 180
set trimspool on

column  plan_name format a32
column  signature format 999,999,999,999,999,999,999
column  category  format a10
column  hint format a70 wrap word

break on plan_name skip 1 on signature skip 1 on opt_type skip 1 on


variable x number
begin
:x :=
    dbms_spm.load_plans_from_cursor_cache(
BEGIN
 FOR i IN (SELECT address, hash_value
 FROM gv$sqlarea WHERE sql_id = '&sql_id.')
 LOOP
 SYS.DBMS_SHARED_POOL.PURGE(i.address||','||i.hash_value, 'C');
 END LOOP;
END;
/
REM Flushes one cursor out of the shared pool. Works on 11g+
REM To create DBMS_SHARED_POOL, run the DBMSPOOL.SQL
script.
REM The PRVTPOOL.PLB script is automatically executed after
DBMSPOOL.SQL runs.
REM These scripts are not run by as part of standard database
creation.
SPO flush_cursor_&&sql_id..txt;
PRO *** before flush ***
SELECT inst_id, loaded_versions, invalidations, address, hash_value
```

Copy Dev explain plan to Prod
----

https://avdeo.com/2012/06/20/fixing-sql-plans-the-hard-way-part-1/

```
accept HINTED_SQL_ID prompt 'Enter good SQL ID:- '
accept CHILD_NO prompt 'Enter child number of good SQL:- '
accept BAD_SQL_ID prompt 'Enter bad SQL ID to be fixed:- '
accept PLAN_HASH_VALUE prompt 'Enter bad SQL plan_hash_value:-
'
set pagesize 0
set line 9999
set verify off;
set heading off;
set feedback off;
set echo off;
set pagesize 0
prompt '====================== OUTPUT
======================'
```

```
-----------------------------------------------------------------------------------------
------------------------------------------------------
---
---   one script to Identify all performace issue related to sql query
---
---    Author      Sugi
```

```
select
   stat_start.snap_id,
   to_char(snap.begin_interval_time,'YYYY-MM-DD HH24:MI') as
begin_hour,
   -- DB time is in units of centiseconds in
DBA_HIST_SYSSTAT.VALUE
   round( (stat_end.value - stat_start.value)/100 , 0) as
seconds_per_hour,
   -- also assumes hourly snapshots, hence divided by 3600
   round( (stat_end.value - stat_start.value)/(100*3600) , 1) as aas
from
```

```
column sample_minute format a16
select
   to_char(round(sub1.sample_time, 'MI'), 'YYYY-MM-DD HH24:MI') as
sample_minute,
   round(avg(sub1.on_cpu),1) as cpu_avg,
   round(avg(sub1.waiting),1) as wait_avg,
   round(avg(sub1.active_sessions),1) as act_avg,
   round(
```

```
column sample_hour format a16
select
   to_char(round(sub1.sample_time, 'HH24'), 'YYYY-MM-DD HH24:MI')
as sample_hour,
   round(avg(sub1.on_cpu),1) as cpu_avg,
   round(avg(sub1.waiting),1) as wait_avg,
   round(avg(sub1.active_sessions),1) as act_avg,
   round(
(variance(sub1.active_sessions)/avg(sub1.active_sessions)),1) as
```

```
set lines 155
col execs for 999,999,999
col min_etime for 999,999.99
col max_etime for 999,999.99
col avg_etime for 999,999.999
col avg_lio for 999,999,999.9
col norm_stddev for 999,999.9999
col begin_interval_time for a30
col node for 99999
break on plan_hash_value on startup_time skip 1
select * from (
select sql_id, sum(execs), min(avg_etime) min_etime, max(avg_etime)
max_etime, stddev_etime/min(avg_etime) norm_stddev
```

https://orastory.wordpress.com/2009/12/29/plans-gone-awry-an-invashti

```
select to_char(sample_time,'HH24:MI:SS') time, session_id sid
, session_serial# s#, user_id usr, session_state state
, time_waited waited, blocking_session blk_by,
blocking_session_status blk_stat
, sql_id, sql_child_number child_id, sql_plan_hash_value hash
, sql_plan_operation plan_op
, event, p1text, p1, p2text, p2, p3text p3, wait_class
, top_level_sql_id top_id, sql_exec_start ex_start
, plsql_entry_object_id pl_obj, plsql_entry_subprogram_id pl_prog
from   dba_hist_active_sess_history
where sample_time between <start_time> and <end_time>
order by time desc;

select sql_plan_hash_value, COUNT(*), MIN(sql_exec_start),
MAX(sql_exec_start)
from   dba_hist_active_sess_history
where  sql_id = <sql_id>
group by sql_plan_hash_value;

select plan_hash_value, optimizer_cost, optimizer_mode,
optimizer_env_hash_value, buffer_gets_delta/executions_delta
from   dba_hist_sqlstat
where sql_id = <sql_id>
and   executions_delta > 0;

SELECT * FROM
TABLE(DBMS_XPLAN.DISPLAY_AWR(<sql_id>,<plan_hash_value>,<
db_id>,<format>));

select
   id      xms_id,
   lpad(' ',depth*1,' ')||operation || ' ' || options xms_plan_step,
   object_name     xms_object_name,
   cost        xms_opt_cost,
   cardinality xms_opt_card,
   bytes       xms_opt_bytes,
   optimizer   xms_optimizer
from
   dba_hist_sql_plan
where
   plan_hash_value = <plan_hash>
order by id

select
OBJ#,INTCOL#,SAVTIME,NULL_CNT,MINIMUM,MAXIMUM,DISTCN
T,DENSITY,LOWVAL,HIVAL,SAMPLE_DISTCNT,SAMPLE_SIZE
from WRI$_OPTSTAT_HISTHEAD_HISTORY
where OBJ#=925718 and INTCOL#=5
order by SAVTIME;
```

```
Script to submit SQL_ID to Oracle SQL Tuning Advisor (DBMS_SQLTUNE).
-- Usage: @tune_sql_id [SQL ID]
--
-- e.g. @tune_sql_id dnrtsnqgfqzsz
-- @tune_sql_id dnrtsnqgfqzsz 10
--
-- Requires ADVISOR system privilege to run tuning task. Requires CREATE ANY SQL PROFILE to apply a profile.
--
http://docs.oracle.com/cd/E11882_01/appdev.112/e40758/d_sqltun.htm#ARPLS68383
--
-- If DBMS_SQLTUNE finds an improved plan the recommendations section will include a note like this:
--
-- SQL Profile Finding (see explain plans section below)
-- -------------------------------------------------------------
-- A potentially better execution plan was found for this statement.
--
-- This script captures the recommendation and generates a PL/SQL block which it displays at the end, accompanied by a backout statement.
-- Copy and paste the generated statement onto the command line and execute it.
-- Execution plans will then include a note that the profile was applied. The profile can also be found in DBA_SQL_PROFILES.

var minutes number;
var recommendations clob
var plsql_call varchar2(200);
var plsql_backout varchar2(200);
```

<-- For cluster

for standalone -->

set pages 500
set linesize 750
column box format a30
column spid format a10
column username format a30
column sid format 9999
column serial for 999999
column status format a15
column username format a10
column sql_text format a80

overall -->

<-- ONLY EXECUTING

```
trimspool on trim on long 2000000 longchunksize 2000000
select
DBMS_SQLTUNE.REPORT_SQL_MONITOR(
   sql_id=>'&sql_id',
```

Missing statements in SQL Monitoring -->

Completed (or)_ERROR_

you can rewrite the sql using below hint

```
EXTRACTVALUE(XMLType(report_summary),'/report_repository_summary/sql/@sql_id') sql_id,

EXTRACTVALUE(XMLType(report_summary),'/report
```

TO_NUMBER(EXTRACTVALUE(XMLType(report_summary),'/report_repository_summary/sql/stats/stat[@name="elapsed_time"]'))

https://community.toadworld.com/platforms/oracle/w/wiki/11372.real-time-sql-monitoring-12c-part-ii

gv$sql_monitor

```
set lines 1000 pages 9999
column sid format 9999
column serial for 999999
column status format a15
column username format a20
```

```
set pages 50000 lines 32767
col OPNAME for a10
col SID form 9999
col SERIAL form 9999999
col PROGRAM for a10
```

```
COLUMN sid FORMAT 99999
COLUMN serial# FORMAT 9999999
COLUMN machine FORMAT A30
COLUMN progress_pct FORMAT 99999999.00
COLUMN elapsed FORMAT A10
```

Overall Waits

Time model

COLUMN username FORMAT A20 COLUMN sid FORMAT 9999 COLUMN serial# FORMAT 9999 COLUMN event FORMAT A40	select stat_name, value from V$SESS_TIME_MODEL where sid = &sid order by value desc;
Each Layer time spend	Explain Plan waiting steps
SELECT ROUND(elapsed_time /1000000) AS "Elapsed (s)", ROUND(cpu_time /1000000,3) AS "CPU (s)", ROUND(queuing_time /1000000,3) AS	col PLAN FOR a150 SELECT RPAD('(' || p.plan_line_ID || ' ' || NVL(p.plan_parent_id,'0') || ')',8) || '|' || RPAD(LPAD (' ', 2*p.plan_DEPTH) ||
v$sqlarea (ALL)	ASH
col machine for a12 trunc col username for a12 trunc col osuser for a8 trunc col cld for 990	column my_ser format 99999 column my_state format a30 column my_blkr format 999 select to_char(a.sample_time, 'HH24:MI:SS')
v$sql (sql_id)	

COL executions FOR A20;
COL rows_processed FOR A20;
COL buffer_gets FOR A20;
COL disk_reads FOR A20;	

Elapsed/CPU/Read/Write MB	Each Layer time spend
FROM (SELECT status, --username, sql_id,	"Elapsed (s)", ROUND(cpu_time /1000000,3) AS "CPU (s)", ROUND(queuing_time /1000000,3) AS
AWR data (More detailed)	SHORT Report
set lines 1000 pages 9999 COL instance_number FOR 9999 HEA 'Inst'; COL start_time for a20 COL plan_hash_value HEA 'Plan|Hash Value'; col sql_id for a15	column program format a30 column username format a30 column sql_id format a13 SELECT query_runs.*, ROUND ((end_time - start_time) * 24, 2) AS

COL instance_number FOR 9999 HEA 'Inst';
COL end_time HEA 'End Time';
COL plan_hash_value HEA 'Plan|Hash Value';
col sql_id for a15
col SQL_PROFILE for a30
	col avg_etime for 999,999
	col avg_lio for 999,999,999
	col avg_pio for 999,999,999
	col begin_interval_time for a30
	col node for 99999

Top I/O Used	Top DISK Read

set lines 1000 pages 9999
COL instance_number FOR 9999 HEA 'Inst';
COL end_time HEA 'End Time';
COL plan_hash_value HEA 'Plan|Hash Value';
col sql_id for a15
	COL instance_number FOR 9999 HEA 'Inst';
	COL end_time HEA 'End Time';
	COL plan_hash_value HEA 'Plan|Hash Value';
	col sql_id for a15
	col SQL_PROFILE for a30

sql_text	bind_variable

```
 set  lines 750 pages 9999
select sql_text  from dba_hist_sqltext where sql_id =
'&SQL_ID';

 set long 20000
```

```
col VALUE_STRING for a50
SELECT
NAME,POSITION,DATATYPE_STRING,VALUE_ST
RING FROM gv$sql_bind_capture WHERE
sql_id='&sql_id';
```

```
col index_name for a50
SELECT owner, index_name,
table_name,last_analyzed, sample_size, num_rows,
partitioned, global_stats
```

PGA/TEMP (History)	PGA

```
HH24:MI:SS';
alter session set
nls_timestamp_format='YYYY/MM/DD HH24:MI:SS';
```

```
col star for A10 head ''

accept seconds prompt "Last Seconds [60] : " default
60;
```

http://blog.go-faster.co.uk/2016/11/ash-analysis-detecting-and-profiling.html

```
select sql_text,rows_processed from v$sql
where USERS_EXECUTING>0;
```

```
select s.sid, s.serial#, p.spid, s.username,
s.program,
t.xidusn, t.used_ublk, t.used_urec, sa.sql_text from
v$process p,v$session s, v$sqlarea sa,
v$transaction t
```

```
select sql_id,
       starting_time,
       end_time,
  (EXTRACT(HOUR FROM run_time) * 3600
                    + EXTRACT(MINUTE FROM run_time)
```

```
select sess_io.inst_id,
       sess_io.sid,
       sess_io.block_gets,
       sess_io.consistent_gets,
V$SQLAREA b where a.sql_id=b.sql_id and
a.status='ACTIVE';
```

http://www.artoftuning.net/sql-plan-comparison/

Wait Percentage % (Only wait events-NO CPU)	Wait Percentage % (Only wait events-WITH CPU)
TOTAL_WAITS, round(100 * (TOTAL_WAITS / SUM_WAITS),2) PCT_TOTWAITS, ROUND((TIME_WAITED / 100),2) TOT_TIME_WAITED, round(100 * (TIME_WAITED / SUM_TIME),2) PCT_TIME COLUMN wait_class format a30 COLUMN event format a60 COLUMN total_waits format 999999 COLUMN total_us format 999999999 COLUMN pct_time format 99.99 COLUMN avg_us format 999999.99 SET echo on	col time_secs format 999,999.99 "Time (s)" col pct format 99.99 "Time|pct" set lines 750 pagesize 10000 SELECT wait_class time_cat, ROUND((time_secs), 2) time_secs, ROUND (time_waited_micro / 1000000) AS time_waited_secs, ROUND (time_waited_micro * 100 / SUM (time_waited_micro) OVER (),2) AS pct_time FROM (SELECT event, total_waits, time_waited_micro

SELECT NVL(a.event, 'ON CPU') AS event,
COUNT(*) AS total_wait_time
FROM v$active_session_history a
WHERE a.sample_time > SYSDATE - 60/2880 -- 30 mins
GROUP BY a.event
ORDER BY total_wait_time DESC;
	select * from (
	select
	WAIT_CLASS ,
	EVENT,
	count(sample_time) as EST_SECS_IN_WAIT
	from v$active_session_history
	where sample_time between sysdate - interval '1'

 <-- shared pool (current) From AWR -->	select event, time_waited "time_waited(s)", case when time_waited = 0 then 0

SELECT * FROM TABLE(DBMS_XPLAN.DISPLAY('','','ALLSTATS LAST +OUTLINE +PEEKED_BINDS +PROJECTION +ALIAS +PREDICATE +COST +BYTES'));	select expl.* from gv$sql sql, v$session ses, TABLE(dbms_xplan.display_cursor(sql.sql_id, sql.child_number,format=>'typical +predicate')) expl where ses.sql_address = sql.address and ses.sid = &&1

<-- shared pool (current)
From AWR -->
	select * from table(dbms_xplan.display_awr('&sql_id', null, null, 'ALLSTATS LAST'));
	SELECT * FROM table(DBMS_XPLAN.DISPLAY_AWR('&sql_id',&plan_hash_value));
	SELECT * FROM TABLE(DBMS_XPLAN.display_sql_plan_baseline(plan_name=>'SQL_PLAN_agz791au8s6jg30a4b3a6'));
	SELECT tf.* FROM DBA_HIST_SQLTEXT ht, table
	(DBMS_XPLAN.DISPLAY_AWR(ht.sql_id,null, null, 'ALL')) tf
	WHERE ht.sql_id='&sql_id';

<-- Killing a session (Good Query)	select vs.sid, vs.username, vs.osuser, vs.process fg_pid, vp.spid bg_pid from v$session vs, v$process vp
select distinct sid OwnSID from v$mystat;	select sys_context('USERENV', 'IP_ADDRESS') from dual;

```
select ' alter system kill session
'''||sid||','||serial#||',@'||inst_id||''' immediate; '    -- 4
hours
from gv$session where username in
('SCHEMA1','SCHEMA2','SCHEMA3') and
status='INACTIVE' and last_call_et > 4*60*60;
```

```
begin
 for x in (select
SID,Serial#,status,last_call_et,username from
v$session
where type='USER'
and status='INACTIVE'
and username = 'TESTER'
and last_call_et > 1200
            ) loop
        execute 'Alter System Kill Session '''|| x.Sid
                     || ',' || x.Serial# || '''';
    end loop;
end;
/
```

Then kill the old sessions using kill -9 spid

```
kill -9 `ps -ef|grep LOCAL=NO|grep oratst1|awk '{print
$2}'`
```

```
for i in `ps -ef |grep "oracledb10g1 (LOCAL=NO)"
|grep -v grep | awk '{print $2}'`
do
 echo  kill -9  $i
done
```

```
SELECT o.sid,
       o.sql_text,
       o.address,
       o.hash_value,
       o.user_name,
       s.schemaname,
       o.sql_id
  FROM v$open_cursor o, v$session s
 WHERE o.saddr = s.saddr AND o.sid = s.sid AND
(O.SID = &sid)
```

```
SCHEMA_NAME, ELAPSED_TIME FROM V$SQL WHERE UPPER(PARSING_SCHEMA_NAME) ='SYS' O
```

https://asktom.oracle.com/pls/asktom/f?p=100:11:::::P11_QUESTION_ID:7143624535091

```
select t.inst_id
       ,s.sid
      ,s.serial#
      ,s.username
      ,s.machine
      ,s.status
      ,s.lockwait
      ,t.used_ublk
      ,t.used_urec
      ,t.start_time
from gv$transaction t
inner join gv$session s on t.addr = s.taddr;
```

If it returns a value, it means there is a pending transaction.

Here is the query:

```
select dbms_transaction.step_id from dual;
```

https://avdeo.com/2012/12/14/oracle-sql-patch-i/

```
VAR signature NUMBER;
VAR signaturef NUMBER;
REM
DECLARE
sql_txt CLOB;
output   varchar2(1000);
h       SYS.SQLPROF_ATTR;
PROCEDURE wa (p_line IN VARCHAR2) IS
BEGIN
DBMS_LOB.WRITEAPPEND(sql_txt,
```

ENABLE/DISABLE/DROP

```
  EXEC
DBMS_SQLTUNE.ALTER_SQL_PROFILE('coe_527
3fz2cqkk80_3455548535','STATUS','DISABLED');

  exec
dbms_sqltune.drop_sql_profile('coe_5273fz2cqkk80_
3455548535');
```

sql Profiles for a sql_id

```
col name for a30
col task_exec_name for a16
col category for a10
col created for a30
col sql_text for a150
col signature for 999999999999999999999999
```

Drop

```
  DECLARE
 i NATURAL;
BEGIN
  i :=
dbms_spm.drop_sql_plan_baseline('SQL_b3d69637a
a86a8ca');
  dbms_output.put_line(i);
END;
/
```

load from cursor

```
  variable sqlid number;
execute :sqlid
:=DBMS_SPM.LOAD_PLANS_FROM_CURSOR_C
ACHE(sql_id=>'5qbbnv0abm2vx',
PLAN_HASH_VALUE=> 4197102931,
SQL_HANDLE => 'SQL_d3318f33dfac7bc2');
```

Taking SQL HINTS from memory

```
chr(9)||chr(9)||'"'||regexp_replace(extractvalue(value(d
), '/hint'),'"','""')||'",'
 from
 xmltable('/*/outline_data/hint'
 passing (
 select
 xmltype(other_xml) as xmlval
 from
 v$sql_plan

select hint from (
select s.sql_id, sd.obj_type,
row_number() over (partition by sd.signature,
sd.category order by sd.signature) row_num,
extractValue(value(t), '/hint') hint
```

Taking SQL HINTS from AWR

```
chr(9)||chr(9)||'"'||regexp_replace(extractvalue(value(
d), '/hint'),'"','""')||'",'
 from
 xmltable('/*/outline_data/hint'
 passing (
 select
 xmltype(other_xml) as xmlval
 from
-- v$sql_plan
```

http://intermediatesql.com/oracle/what-are-sql-profiles-and-why-do-we-need-them/

http://expertoracle.com/2015/07/08/flush-bad-sql-plan-from-shared-pool/

```
                                                          col "00" for 99.9
set pages 100                                             col "01" for 99.9
select INSTANCE_NUMBER,                                   col "02" for 99.9
SUM(value),                                               col "03" for 99.9
to_char(trunc(end_time,’HH’),’dd-mm-yyyy hh24:mi’)        col "04" for 99.9
as sample_time                                            col "05" for 99.9
from dba_hist_sysmetric_history                           col "06" for 99.9
where metric_name=’Average Active Sessions’               col "07" for 99.9
and end_time between sysdate -1 and sysdate               col "08" for 99.9
group by trunc(end_time,’HH’), INSTANCE_NUMBER            col "09" for 99.9
order by 3,1;                                             col "10" for 99.9
```

http://kerryosborne.oracle-guy.com/2008/10/unstable-plans/

gation/

http://www.williamrobertson.net/documents/automated-	Test it before using

```
set head off
set verify off
set echo off
set pages 1500
set linesize 100
```

http://allappsdba.blogspot.com/2012/04/queries-to-get-session-information.html

```
ALTER SESSION SET
"_SQLMON_MAX_PLAN"=4020;
ALTER SESSION SET
"_SQLMON_MAX_PLANLINES"=4000;
```

```
NO_XML_QUERY_REWRITE */
t.report_id, x1.sql_id, x1.plan_hash,
x1.sql_exec_id,
x1.elapsed_time/1000000 ELAP_SEC
FROM dba_hist_reports t
```

```
report_id,
key1
sql_id,
key2
sql_exec_
```

uro-
pagano.c
om/2015/
05/04/hist
orical-sql-

```
NO_XML_QUERY_REWRITE */
t.report_id, x1.sql_id, x1.plan_hash,
x1.sql_exec_id, x1.elapsed_time/1000000
ELAP_SEC
FROM dba_hist_reports t
```

```
SELECT SID, SERIAL#, OPNAME,
TARGET, SOFAR, TOTALWORK,
UNITS,
TO_CHAR(START_TIME,'DD/MON/YYY
Y HH24:MI:SS') START_TIME,
```

```
column username format a20
Select s.username, s.Sid, s.serial#,
S.Sql_Id, round((Sysdate-
Sql_Exec_Start)*24*60*60/60,0)
MINUTES, Sql_Text
```

Stats

select vsn.name, vst.value from v$sesstat vst, v$statname vsn where vsn.statistic# = vst.statistic# and vst.value != 0 and vst.sid = &sid		
https://blog.yannickjaquier.com/oracle/real-time-sql-monitoring.html		
AWR	Time based	

SELECT
s.snap_id,TO_CHAR(s.begin_interval_time, 'DD-MON HH24:MI')
snap_time,ss.sql_id,ss.plan_hash_value,
	s.sql_id,
	sum(case
	when begin_interval_time = to_date('14-nov-2017 1100','dd-mon-yyyy hh24mi')
		from
		(
		  select
		    sql_id,

Explain Plan waiting steps		

SELECT
    RPAD('(' || p.plan_line_ID || ' ' ||
NVL(p.plan_parent_id,'0') || ')',8) || '|' ||
    RPAD(LPAD (' ', 2*p.plan_DEPTH) ||
	https://blog.yannickjaquier.com/oracle/real-time-sql-monitoring.html	

AWR -- FULL Details --check	Current Memory	https://github.com/iusoltsev/sqlplus/blob/master/
SQL> @dba_hist_sqlstat "sql_id = '8d49sjc17xwuc' and snap_id between 86116 and 86260 and executions_delta > 0"	<== Advanced Queries (Takes time)	

col avg_et_secs justify right format 9999999.99
col cost justify right format 9999999999
col timestamp justify center format a25
col parsing_schema_name justify center
	FROM dba_hist_active_sess_history
	WHERE sample_time
	BETWEEN '16-AUG-18 12.00.00.000000000 AM'
	AND '20-AUG-18 01.00.00.000000000	

All in one	Rank Based	

dba_hist_sqltext t where t.sql_id = x.sql_id and rownum = 1) txt
FROM (
SELECT sn.snap_id
,   TO_CHAR(sn.end_interval_time,'DD-
	dba_hist_sqltext t where t.sql_id = x.sql_id and rownum = 1) txt
	  from (
	SELECT snap_id
	, sql_id	

TEMP

```
col star for A10 head ''

accept seconds prompt "Last Seconds [60] : " default 60;

exec dbms_stats.flush_database_monitoring_info;

select inserts,updates,deletes from
```

http://guyharrison.squarespace.com/opsg
samples/

```
select EVENT, WAIT_CLASS,
SUM(TOTAL_WAITS),round
(sum(TIME_WAITED_micro)/1000000,0)
as time_waited_secs  from
V$SYSTEM_EVENT
where wait_class != 'Idle' group by
EVENT, WAIT_CLASS
```

```
   sql_id,
   event,
   count(*),
   lpad(round(ratio_to_report(count(*))
over () * 100)||'%',10,' ') percent
from
   dba_hist_active_sess_history h,              <-- wait event for a specific snap_id
```

http://www.centroid.com/blog/monitoring-exadata-smart-scan

```
select * from table(dbms_xplan.display('ASH_PLAN_TABLE','4dszd9dysry0c',null,'dbid=123456 and plan_hash_value = 3412983073'));
```

```
select * from table(dbms_xplan.display('ASH_PLAN_TABLE','4dszd9dysry0c','+ADAPTIVE','dbid=2783210685 and plan_hash_value = 3412983073'));
```

https://hoopercharles.wordpress.com/2010/03/01/dbms_xplan-format-parameters/

http://blog.go-faster.co

```
select 'ALTER SYSTEM KILL SESSION
'''||SID||','||SERIAL#||'''IMMEDIATE;' from
v$session where username = 'A' and
STATUS = 'ACTIVE';
```

http://www.tek-tips.com/viewthread.cfm?qid=1572803

```
set linesize 30
spool kill_old_sess.sh
select '#!/bin/ksh' from dual;
select 'kill -9 ' || spid
from v$process p, v$SESSTAT
t,v$sess_io i ,v$session s
where i.sid=s.sid
        and p.addr=paddr(+)
```

```
RDER BY ELAPSED_TIME DESC
```

<= My own sql Patch

Enable/disable	Fix baseline of one sql_id to another	

DECLARE
  l_plans_altered PLS_INTEGER;
BEGIN
  l_plans_altered := DBMS_SPM.alter_sql_plan_baseline(
    sql_handle   => 'SYS_SQL_1447ba3a1d83920f',
    plan_name   => 'SYS_SQL_PLAN_1d83920fae82cf72',
	variable sqlid number;
	execute :sqlid :=DBMS_SPM.LOAD_PLANS_FROM_CURSOR_CACHE(sql_id=>'31pux6bymf1d4');
	
	SQL> select sql_handle, plan_name, enabled from dba_sql_plan_baselines;	

Taking SQL HINTS from PROFILES	**Taking SQL HINTS from PLAN_TABLE**	

select hint as outline_hints
   from (select p.name, p.signature, p.category, row_number()
         over (partition by sd.signature, sd.category order by sd.signature) row_num,
         extractValue(value(t), '/hint') hint
   from sqlobj$data sd, dba_sql_profiles p,
	regexp_replace(extractvalue(value(d), '/hint'),'''','''''') plan_hint
	        from
	        xmltable('/*/outline_data/hint'
	                passing (
	                        select
	                                xmltype(other_xml) as xmlval
	                        from plan_table
		https://jonathanlewis.w

http://www.alfredokriegdba.com/2015/02/

http://www.nocoug.org/download/2008-08/a-tour-of-the-awr-tables.nocoug-Aug-21-2008.abercrombie.html#script-aas-exact	http://www.nocoug.org/download/2008-08/a-tour-of-the-awr-tables.nocoug-Aug-21-2008.abercrombie.html#aas-definition

ɔ.uk/2016/11/ash-analysis-detecting-and-profiling.html

Capacity Planning

Database/Table/tablespace/segment Growth Rate History

Tablespace growth Rate

Database Growth Trend/Rate

List object growth over last N days, sorted by growth desc table size changes between two periods.

SGA Allocation

1 lakh records table with 5 column

Generate DB Load

No of logons to database per hour

No of logons to database per minute

DOP for parallel session

Maximum no of sessions in the database
Cpu Demand

Transaction per second

Tuning

CPU wait time

Format the Explain plan

Finding Current trace file

DB Sequential read / Contention problem

Size of the Shared Pool

Sharable memory Used by database

Library cahe hit ratio

Clearing Invalidation

SHARED_POOL_RESERVED_SIZE

Keeping Large Objects in SGA

Tuning the Data Dictionary Cache

Sizing the User Global Area

TUNING DB BUFFER CACHE

how many blocks are used by each object

Calculating the Hit Ratio for Multiple Pools

Sizing the Redo Log Buffer

Sizing the SGA For Java

Multiple I/O Slaves

Multiple DBWR Processes

I/O tuning

SORTING

Diagnosing Contention for Latches

Diagnosing Contention for Manual Rollback Segment Header

Sizing Transaction Rollback Data

Space Requirement For Undo Retention

Monitoring and detecting lock contention

dispatcher processes

INDEX

Using Oracle Blocks Efficiently

Space Recover & performance due to defragmentation

Explain Plan

SQL Trace

Locking the Memory

Get top 10 hot segment

See hot data files (from a single block read time judge)

View pga proposed

Seesion Based Hit Ratio

FREE Buffer Waits

Top 10 CPU users

Top 10 users reading Data

enq: TX - row lock contention

ORA-10631: SHRINK clause should not be specified for this object

Top 10 CPU consumers in last 5 minutes

Top 10 waiting sessions in last 5 minutes

CPU Cores value from db

CPU Time + wait time = Total Time

CPU by OS/DB/Demand

Other DB Tuning

Avg Sql/Database Response Time

Sid waiting for ?

Sql statement running by SID

Current sessions problem

Where the particular SID was
 spending time when executing

What Class is Causing
Waits

system statistics

Metric values by %

AWR Snapshot list

Checking i/o Latency

Response time analysis

CPU Time + wait Time

Create manual awr snapshot

Cmd line sql monitor

top 100 sql queries based on CPU, elapsed time for the past 7 days in Oracle

Do we have multiple plan hash values for the same SQL ID

Oracle cost calculations for sql query

Monitoring Disk activity

Wait event for the sessions

oracle_which_sql_caused_wait_event

Single block read time

Performance tuning

Reading tkprof output

Wait event solution & concept

tools

TUNING

sql profiles related to sql_id

histograms

STATISTICS

Table statistics
Index statistics
schema Statistics
Database statistics

Statistics collection for Dictionary

When was last analyzed

History of rowcount in table / History of table statistics

Statistics collection for all schema

Stale statistics

automatic stats collection enabled or not

automatic stats collection for database formula (List stale more than 10%)

Automatic Stats Gathering Job - Parameters

Restore Old Statistics for DB/Schema/Table/Fixed Obj

Collecting historgram for a table

Histogram details for a table

statistics collection for schema including histogram

optimizer fraud- Manual stats setting

Statistics difference report

statistics of objects of a specific sql id

sql_profile details

sql base lines details

Top 10 statements

Top CPU consuming Session in last 5 minutes

Top Waiting Session in last 5 minutes

Top Waiting Event in last 5 minutes

Top SESSION by CPU usage, wait time and IO time in last 5 minutes

Top SQL by CPU usage, wait time and IO time in last 5 minutes

Top 10 sql by elapsed time

TOP 10 BY BUFFER GETS:

TOP 10 BY PHYSICAL READS:

TOP 10 BY EXECUTIONS:

TOP 10 BY PARSE CALLS:

TOP 10 BY SHARABLE MEMORY

TOP 10 BY VERSION COUNT:

ASH Check for current issue

everything about last 1 hour

SQL Tuning Advisor

CPU usage debugging

Does DB looking for More CPU's

CPU Debug : resmgr:cpu quantum

Optimizer

optimizer override

Optimizer Evolution - CERN

Manual Creation of sql profile & sqlpatch(Manual hint for apps sql statements who cant change the code)

Advisors

advisor recommendation for a taskid

Automatic sql tuning advisor recommendation summary

Table Fragmentation

Jobs/ Scheduler

Maintenance window time/Date

Command to change the default maintenance window

History of default maintenance job run status

Default jobs enabled or not

auot stats collection job details

Others

If we want to save every second ASH data to disk,By default, it save 1 second snapshot out of 10 seconds.

Cpu Starvation Calculation

AWR Related

Quick Cross-Period AWR Comparison

comparing top SQL in AWR across periods:

Load Profile query

Here is a technique I use when troubleshooting runaway query. It is based on da
tremendous value and potentially a good alternative to 10046 traces. The b
“statistics_level=all” mode. But it is OK to do that at session level, even in prod
statistics_level=all and recreate ru

– Find SQL_ID / Child_
– Find plan_hash_value fron
– Assuming the query is using bind variables, capture bind variables from v$sql_
bind v
– In a separate sessio
– Start executing SQL in this session. If there were bind variables,
– If the query is absolutely hopeless, cancel executio
– Find SQL_ID of just executed query by looking in v$session.prev_sql_id. If sq
same Plan_hash_va
– Examine plan statistics from v$sql_plan_statistics. The beauty of plan statisti
interpreting plan statistics. Once becoming familiar with them and having a dece
in numbers. 10-15 seconds of execution should be enough for the query

This recipe looks rather tedious, and it really is, unless the tool is used that does a
runaway queries (which are rare anyway) but simply expensive queries with a lo
cases or when in doubts. Most often, problems are sim

http://db.geeksinsight.com/2012/10/15/scripts-databasetabletablespace-growth-re

Weekly

```
with
a as (
select name,ts#,block_size
from v$tablespace,dba_tablespaces
where name = tablespace_name
```

```
select * from (select c.TABLESPACE_NAME,c.segment_name "Object
Name",b.object_type,
sum(space_used_delta) / 1024 / 1024 "Growth (MB)"
from dba_hist_snapshot sn,
dba_hist_seg_stat a,
column "Percent of Total Disk Usage" justify right format 999.99
column "Space Used (MB)" justify right format 9,999,999.99
column "Total Object Size (MB)" justify right format 9,999,999.99
set linesize 150
set pages 80
```

45% of ram size

EXAMPLE:

RAM SIZE = 484 MB

So 45 % should be = 216 MB

split the 216 MM for SGA,PGA,BACKGROUNG PROCESSES

Fixed background process requires = 40 MB

SGA = 160 MB

PGA = 16 MB

Table size May be 100 MB

```
set serveroutput on
prompt
-- This makes it possible to locate this session in
-- order to kill it later.
BEGIN
  dbms_output.put_line(' ');
  dbms_output.put_line('SID is ' || sys_context('USERENV','SID'));
END;
```

```
fgrep "14-MAR-2014" listener.log | fgrep "establish" | \
awk '{print $1 " "  $2 }' | awk -F: '{print $1 }'    | \
sort | uniq -c
```

```
fgrep "14-MAR-2014 03:" listener.log | fgrep "establish" | \
awk '{print $1 " "  $2 }' | awk -F: '{print $1 ":" $2 }'    | \
sort | uniq -c
```

PRALLEL_THREADS_PER_CPU * CPU_COUNT (number of CPU cores) * ACTI

```
SELECT A.SNAP_ID,A.INSTANCE_NUMBER
"ID",B.BEGIN_INTERVAL_TIME,B.END_INTERVAL_TIME,A.RESOURCE_NA
ME,
CURRENT_UTILIZATION "CURRENT",MAX_UTILIZATION "MAX"
FROM WRH$_RESOURCE_LIMIT A, WRM$_SNAPSHOT B
WHERE A.RESOURCE_NAME LIKE '%session%'
AND A.SNAP_ID=B.SNAP_ID
AND A.INSTANCE_NUMBER= B.INSTANCE_NUMBER
AND ((A.INSTANCE_NUMBER=1
AND B.BEGIN_INTERVAL_TIME > (SELECT STARTUP_TIME FROM
```

http://datavirtualizer.com/oracle-cpu-time/

```
WITH hist_snaps
AS (SELECT instance_number,
snap_id,
round(begin_interval_time,'MI') datetime,
(  begin_interval_time + 0 - LAG (begin_interval_time + 0)
OVER (PARTITION BY dbid, instance_number ORDER BY snap_id)) * 86400
diff_time
FROM dba_hist_snapshot), hist_stats
AS (SELECT dbid,
instance_number,
snap_id,
stat_name,
VALUE - LAG (VALUE) OVER (PARTITION BY
dbid,instance_number,stat_name ORDER BY snap_id)
delta_value
FROM dba_hist_sysstat
WHERE stat_name IN ('user commits', 'user rollbacks'))
SELECT datetime,
ROUND (SUM (delta_value) / 3600, 2) "Transactions/s"
FROM hist_snaps sn, hist_stats st
WHERE     st.instance_number = sn.instance_number
AND st.snap_id = sn.snap_id
AND diff_time IS NOT NULL
GROUP BY datetime
ORDER BY 1 desc;
```

```
select metric_name, round(value,2) from v$sysmetric
where metric_name in ('Database CPU Time Ratio', 'Database Wait Time Ratio')
and intsize_csec = (select max(INTSIZE_CSEC) FROM V$SYSMETRIC);
```

```
COLUMN QUERY_PLAN FORMAT A60
SELECT        LPAD (' ', 2 * LEVEL)
          || operation
          || ' '
          || options
          || ' '
          || object_name query_plan
     FROM plan_table
SET LINESIZE 100
COLUMN trace_file FORMAT A60

SELECT s.sid,
       s.serial#,
       pa.value || '/' || LOWER(SYS_CONTEXT('userenv','instance_name')) ||
 from user_tables
 where table_name =<tablename>;

select table_name,round((num_rows*avg_row_len/1024),2)||'kb' "size"
 from user_tables
```

You set the size of the shared pool with the SHARED_POOL_SIZE initialization parameter.
It defaults to 8,388,608 bytes (8 MB).

```
SELECT SUM(250 * users_opening)  FROM v$sqlarea;


SQL> SELECT 250 * value bytes_per_user
 FROM v$sesstat s, v$statname n
 WHERE s.statistic# = n.statistic#
 AND n.name = 'opened cursors current'
 AND s.sid = 15


select sum(pins) "Executions", sum(reloads)
 "Cache Misses", sum(reloads)/sum(pins)
 from v$librarycache;
```

```
SQL> select count(*) from hr.employees;
SQL> select namespace,pins,reloads,invalidations  from v$librarycache;
SQL> ANALYZE TABLE hr.employees COMPUTE STATISTICS;

SQL> select count(*) from hr.employees;
SQL> select namespace,pins,reloads,invalidations  from v$librarycache;
```

• Find those PL/SQL objects that are not kept in the
library cache:

```
select * from v$db_object_cache
 where sharable_mem > 10000
 and (type='PACKAGE' or type='PACKAGE BODY' or
 type='FUNCTION' or type='PROCEDURE')
 and KEPT='NO';
```

• Pin large packages in the library cache:

```
SQL> EXECUTE dbms_shared_pool.keep(‘package_name');
```

Keep the ratio of the sum of GETMISSES to the sum of GETS less than 15%:

```
SQL> select parameter, gets, getmisses from v$rowcache;
```

UGA space used by your connection:

```
SQL> select SUM(value) ||'bytes' "Total session memory"
 from V$MYSTAT, V$STATNAME
 where name = 'session uga memory'
 and v$mystat.statistic# = v$statname.statistic#;
```

UGA space used by all Oracle Shared Server users:

```
SQL> select SUM(value) ||'bytes' "Total session memory"
 from V$SESSTAT, V$STATNAME
 where name = 'session uga memory'
 and v$sesstat.statistic# = v$statname.statistic#;
```

Maximum UGA space used by all users:

```
SQL> select SUM(value) ||'bytes' "Total max memory"
 from V$SESSTAT, V$STATNAME
 where name = 'session uga memory max'
 and v$sesstat.statistic# = v$statname.statistic#;
```

Dynamic Buffer Cache Advisory Parameter

```
SQL> show parameter DB_CACHE_ADVICE

NAME                                 TYPE        VALUE
------------------------------------ ----------- ------------------------------
db_cache_advice                      string      ON
```

Three values are allowed: OFF, ON, and READY.

Using V$DB_CACHE_ADVICE

```
-------------------------
SELECT
size_for_estimate,buffers_for_estimate,estd_physical_read_factor,estd_physical
_reads
FROM V$DB_CACHE_ADVICE
WHERE name = 'DEFAULT'
AND block_size = ( SELECT value FROM V$PARAMETER
WHERE name = 'db_block_size')
AND advice_status = 'ON';
```

During peak running times, use the following query to calculate how many blocks are used by each object:

```
SQL> SELECT owner#, name, count(*) blocks
 FROM v$cache
 GROUP BY owner#, name;
```

KEEP Buffer Pool Guidelines

- Tuning goal: Keeping blocks in memory
- Size: Holds all or nearly all blocks
- Tool: ANALYZE ... ESTIMATE STATISTICS

```
SQL> ANALYZE TABLE hr.countries ESTIMATE STATISTICS;
SQL> SELECT table_name, blocks
 FROM dba_tables
 WHERE owner = 'HR'
 AND table_name = 'COUNTRIES';

TABLE_NAME BLOCKS
---------- ----------
COUNTRIES 14
```

```
SELECT name,
1 - (physical_reads / (db_block_gets +
consistent_gets)) "HIT_RATIO"
 FROM sys.v$buffer_pool_statistics
 WHERE db_block_gets + consistent_gets > 0;
```

Sizing the Redo Log Buffer

- Adjust the LOG_BUFFER parameter
- Default value: OS-specific, generally 500k

Using Dynamic Views to Analyze Redo Log Buffer Efficiency

LOG_BUFFER
LOG_CHECKPOINT_INTERVAL
LOG_CHECKPOINT_TIMEOUT

```
SQL> select sid, event, seconds_in_wait, state
 from v$session_wait
 where event = 'log buffer space%';

SID EVENT SECONDS_IN_WAIT STATE
----- ------ --------------- ---------
log buffer space 110 WAITING
```

```
SQL> SELECT name, value
 FROM v$sysstat
 WHERE name = 'redo buffer allocation retries';
```

```
SQL> select name, value
 from v$sysstat
 where name='redo log space requests';
SQL> SELECT name, value
 FROM v$sysstat
 WHERE name IN ('redo buffer allocation retries',
 'redo entries');
SQL> select event, total_waits, time_waited,
average_wait
 from v$system_event
 where event like 'log file switch completion%';
```

– Increase the size of the redo log files.

```
SQL>select event, total_waits, time_waited, average_wait
 from v$system_event
 where event like 'log file switch (check%';
```

– Check the frequency of checkpoints and set the appropriate values for LOG_CHECKPOINT_INTERVAL and LOG_CHECKPOINT_TIMEOUT.

– Check the size and number of redo log groups

```
SQL> select event, total_waits, time_waited,
average_wait
 from v$system_event
 where event like 'log file switch (arch%';
```

– Regulate archiving speed.

The LGWR process starts a new ARCn process whenever the current number of ARCn processes is insufficient to handle the workload. If you anticipate a heavy workload for archiving, such as during bulk loading of data, specify multiple archiver processes with the LOG_ARCHIVE_MAX_PROCESSES initialization parameter.

- DB_BLOCK_CHECKSUM is set to TRUE, and therefore adds performance overhead.

- SHARED_POOL_SIZE:

– 8 KB per loaded class

– 50 MB for loading large JAR files

- Configure Oracle Shared Server
- JAVA_POOL_SIZE

– 20 MB default

– 50 MB for medium-sized Java application

```
SELECT TOTAL_WAITS FROM V$SYSTEM_EVENT WHERE EVENT = 'FREE
BUFFER WAITS';

col name for a70
SQL> SELECT phyrds,phywrts,d.name
 FROM v$datafile d, v$filestat f
 WHERE d.file#=f.file# order by d.name;
```

```
SQL> select name, value from v$sysstat where name = 'sorts (rows)';
NAME VALUE
----------------- ------------------------------
sorts (rows) 639330
SQL> select disk.value "Disk", mem.value "Mem",
 (disk.value/mem.value)*100 "Ratio"
 from v$sysstat mem, v$sysstat disk
 where mem.name = 'sorts (memory)'
 and disk.name = 'sorts (disk)';

Disk Mem Ratio
--------- --------- ---------
23 206 11.165049
```

Increase the Shared pool & buffer cache accordingly

```
SQL> SELECT class, count FROM v$waitstat  WHERE class LIKE '%undo%';
or
SQL> SELECT sum(value) FROM v$sysstat WHERE name IN ('db block gets',
'consistent gets');
or
SQL> SELECT sum(waits)* 100 /sum(gets) "Ratio",
 sum(waits) "Waits", sum(gets) "Gets"
 FROM v$rollstat;
```

```
SELECT s.username, t.used_ublk, t.start_time
 FROM v$transaction t, v$session s
 WHERE t.addr = s.taddr;
```

```
SQL> SELECT s.username, t.used_ublk, t.start_time
 FROM v$transaction t, v$session s
 WHERE t.addr = s.taddr;
```

Undo Space = (UNDO_RETENTION * (Undo Blocks Per Second * DB_BLOCK_SIZE)) + DB_BLOCK_SIZE

```
SELECT (RD * (UPS * OVERHEAD) + OVERHEAD) AS "Bytes"
FROM (SELECT value AS RD FROM v$parameter
WHERE name = 'undo_retention'),
(SELECT (SUM(undoblks) / SUM( ((end_time - begin_time) * 86400)))
AS UPS FROM v$undostat),
(SELECT value AS Overhead FROM v$parameter
WHERE name = 'db_block_size');
```

```
V$LOCK
V$LOCKED_OBJECT
DBA_WAITERS
DBA_BLOCKERS
SQL> SELECT owner, object_id, object_name, object_type,
 v$lock.type
 FROM dba_objects, v$lock
 WHERE object_id = v$lock.id1 and object_name = table_name;
```

```
SELECT network , status ,
 SUM(OWNED) Clients,
 SUM(busy)*100/(SUM(busy)+SUM(idle)) "Busy Rate"
 FROM v$dispatcher group by network,status;
```

```
SELECT d.network network, d.name disp, s.username oracle_user,
s.sid sid,s.serial# serial#, p.username os_user,
p.terminal terminal, s.program program
FROM v$dispatcher d, v$circuit c, v$session s, v$process p
WHERE d.paddr = c.dispatcher(+)
AND c.saddr = s.saddr(+)
AND s.paddr = p.addr (+)
order by d.network, d.name, s.username;
```

As a general guideline, you should create indexes on tables that are often queried for less than 5% of the table's rows

Monitoring Index Space (Rebilding Indexes)

- To collect usage statistics regarding an index:

```
SQL> ANALYZE INDEX EMP_NAME_IX VALIDATE STRUCTURE;
```

- To view statistics collected:

```
SQL> SELECT name, (DEL_LF_ROWS_LEN/LF_ROWS_LEN) * 100
 AS wastage
 FROM index_stats;
```

- Rebuild indexes with wastage greater than 20%:

```
SQL> ALTER INDEX EMP_NAME_IX REBUILD;
```

- To coalesce indexes (alternative to REBUILD):

```
SQL> ALTER INDEX EMP_NAME_IX COALESCE;
```

```
SELECT INDEX_NAME, USED FROM V$OBJECT_USAGE;
```

To display segments with less than 10% free blocks:

```
SQL> SELECT owner, table_name, blocks, empty_blocks
 FROM dba_tables
 WHERE empty_blocks / (blocks+empty_blocks) < .1;
OWNER TABLE_NAME BLOCKS EMPTY_BLOCKS
------ ---------- ---------- ------------
HR EMPLOYEES 1450 50
HR COUNTRIES 460 40
```

To avoid dynamic allocation:

```
SQL> ALTER TABLE hr.employees ALLOCATE EXTENT;
Table altered.
```

Recovering Space

Below the high-water mark:

- Use the Export and Import utilities to:
  - Export the table
  - Drop or truncate the table
  - Import the table

Or, use the Alter Table Employees Move;
command to move the table

- Above the high-water mark, use the Alter Table Employees Deallocate Unused; command.

```
SQL> Explain plan for  select last_name from hr.employees;
```

- At the session level:

```
SQL> alter session set SQL_TRACE = {true|false};
SQL> execute DBMS_SESSION.SET_SQL_TRACE  ({true|false});
SQL> execute DBMS_SYSTEM.SET_SQL_TRACE_IN_SESSION (session_id,
serial_id, {true|false});
```

On some operating systems, the DBA can lock the SGA into real memory by setting the LOCK_SGA initialization parameter to TRUE, so it is never paged out to disk. Obviously, the Oracle server performs better if the entire SGA is kept in real memory. This should be used only on systems that have sufficient memory to hold all the SGA pages without degrading performance in other areas.

```
col owner for a10
col objct_name for a50
select * from
(Select
ob.owner, ob.object_name, sum (b.tch) Touchs
from x$bh b, dba_objects ob
where b.obj = ob.data_object_id
and b.ts#> 0
group by ob.owner, ob.object_name
order by sum (tch) desc)
where rownum <= 10 ;
```

```
SELECT t.file_name,
t.tablespace_name,
round (s.singleblkrdtim / s.singleblkrds, 2) AS CS,
s.READTIM,
s.WRITETIM
FROM v$filestat s, dba_data_files t
WHERE s.file# = t.file_id and rownum <= 10 order by cs desc ;
```

```
SELECT (SELECT ROUND (value/1024/1024, 0) FROM v$parameter
WHERE name = 'pga_aggregate_target') "Current Mb"
, ROUND (pga_target_for_estimate/1024/1024, 0) "Projected Mb"
, ROUND (estd_pga_cache_hit_percentage) "%"
FROM v$pga_target_advice
ORDER BY 2;
```

```
select Username,
OSUSER,
Consistent_Gets,
Block_Gets,
Physical_Reads,
100*( Consistent_Gets + Block_Gets - Physical_Reads)/
( Consistent_Gets + Block_Gets ) "Hit Ratio %"
from V$SESSION,V$SESS_IO
where V$SESSION.SID = V$SESS_IO.SID
and ( Consistent_Gets + Block_Gets )>0
and username is not null
order by Username,"Hit Ratio %";
```

increase DBWn process by setting db_writer_process parameter

```
select rownum as rank, a.*
from (
   SELECT v.sid, program, v.value / (100 * 60) CPUMins
   FROM v$statname s , v$sesstat v, v$session sess
  WHERE s.name = 'CPU used by this session'
    and sess.sid = v.sid
    and v.statistic#=s.statistic#
    and v.value>0
  ORDER BY v.value DESC) a
where rownum < 11
  ;
```

```
select rownum as rank, a.*
from (
   SELECT v.sid, program, v.value
   FROM v$statname s , v$sesstat v, v$session sess
  WHERE s.name = 'consistent gets'
    and sess.sid = v.sid
    and v.statistic#=s.statistic#
    and v.value>0
  ORDER BY v.value DESC) a
where rownum < 11   ;
```

For which SQL currently is waiting on:

```
---------------------------------------------------------
select sid, sql_text from v$session s, v$sql q where sid in (select sid from
v$session where state in ('WAITING') and wait_class != 'Idle' and event='enq:
TX - row lock contention' and (q.sql_id = s.sql_id or q.sql_id = s.prev_sql_id));
```

```
SELECT dt.owner, dt.table_name,
(CASE
WHEN NVL(ind.cnt, 0) < 1 THEN 'Y'
ELSE 'N'
END) AS can_shrink
FROM dba_tables dt,
(SELECT table_name, COUNT(*) cnt
FROM dba_indexes di
WHERE index_type LIKE 'FUNCTION-BASED%'
GROUP BY table_name) ind
WHERE dt.table_name = ind.table_name(+)
AND dt.table_name NOT LIKE 'AQ$%'
AND dt.table_name NOT LIKE 'BIN$%'
AND dt.owner = 'CSS_DATA'
ORDER BY 1, 2;
```

```
select * from
(
select session_id, session_serial#, count(*)
from v$active_session_history
where session_state= 'ON CPU' and
 sample_time > sysdate - interval '5' minute
group by session_id, session_serial#
order by count(*) desc
)
where rownum <= 10;
```

```
select * from
(
select session_id, session_serial#,count(*)
from v$active_session_history
where session_state='WAITING'  and
 sample_time >  sysdate - interval '5' minute
group by session_id, session_serial#
order by count(*) desc
)
where rownum <= 10;

SELECT MIN (val)
  FROM (SELECT TO_NUMBER (p.VALUE) AS val
          FROM SYS.v_$parameter p
         WHERE UPPER (p.NAME) = 'CPU_COUNT'
        UNION ALL
        SELECT o.VALUE AS val
          FROM SYS.v_$osstat o
         WHERE UPPER (o.stat_name) = 'NUM_CPUS'
            OR UPPER (o.stat_name) = 'NUM_CPU_CORES');

SELECT   mymodule "Module", SUM (cpu_time) "CPU Time", SUM (wait_time)
"Wait
Time",
         SUM (cpu_time) + SUM (wait_time) "Total Time"
    FROM (SELECT a.module mymodule,
                 (CASE (session_state)
                     WHEN 'ON CPU'
                        THEN wait_time / 100
                  END
                 ) cpu_time,
                 (CASE (session_state)
                     WHEN 'WAITING'
                        THEN time_waited / 100
                  END
                 ) wait_time
            FROM dba_hist_active_sess_history a, dba_hist_snapshot b
           WHERE b.end_interval_time > sysdate-10
             AND a.snap_id = b.snap_id
             AND a.user_id NOT IN (0, 5)
             AND a.instance_number = b.instance_number)
GROUP BY mymodule
  HAVING SUM (cpu_time) + SUM (wait_time) > 0
ORDER BY 2 DESC
```

```
select
                decode(n.wait_class,'User I/O','User I/O',
                                   'Commit','Commit',
                                   'Wait')                          CLASS,
                sum(round(m.time_waited/m.INTSIZE_CSEC,3))                AAS
          from  v$waitclassmetric  m,
                v$system_wait_class n
          where m.wait_class_id=n.wait_class_id
            and n.wait_class != 'Idle'
          group by  decode(n.wait_class,'User I/O','User I/O', 'Commit','Commit',
'Wait')
         union
            select 'CPU_ORA_CONSUMED'                                   CLASS,
                   round(value/100,3)                                   AAS
            from v$sysmetric
            where metric_name='CPU Usage Per Sec'
              and group_id=2
         union
           select 'CPU_OS'                                        CLASS ,
                   round((prcnt.busy*parameter.cpu_count)/100,3)          AAS
           from
             ( select value busy from v$sysmetric where metric_name='Host CPU
Utilization (%)' and group_id=2 ) prcnt,
             ( select value cpu_count from v$parameter where name='cpu_count' )
parameter
         union
            select
              'CPU_ORA_DEMAND'                                        CLASS,
              nvl(round( sum(decode(session_state,'ON CPU',1,0))/60,2),0) AAS
            from v$active_session_history ash
            where SAMPLE_TIME > sysdate - (60/(24*60*60));
```

```
select to_char(begin_time,'hh24:mi') time, round( value * 10, 2) "Response Time
(ms)"
     from v$sysmetric
     where metric_name='SQL Service Response Time';
```

```
SELECT sid, serial# , username, event,blocking_session, seconds_in_wait, sql_i

SELECT sql_text FROM v$sql WHERE sql_id= 'g3xry817zpwk3';

SELECT wait_class_id, wait_class#, wait_class, total_waits, time_waited FROM v


SELECT stat_name, value FROM v$sess_time_model WHERE sid=130;


SELECT wait_class_id, wait_class#, wait_class, total_waits, time_waited FROM v

select * from sys.aux_stats$

SELECT metric_name, AVG(value), metric_unit FROM v$sysmetric WHERE met


SELECT   snap_id, snap_level, instance_number,
         to_char(begin_interval_time, 'yyyy/mm/dd hh24:mi:ss') begin_interval_time,
         to_char(end_interval_time, 'yyyy/mm/dd hh24:mi:ss') end_interval_time,
         to_char(flush_elapsed) flush_elapsed,
         to_char(startup_time, 'yyyy/mm/dd hh24:mi:ss') startup_time,
         error_count
FROM     DBA_HIST_SNAPSHOT
WHERE    dbid = :dbid
order by 1
```

```
Select Inst_Id, EVENT, TOTAL_WAITS, TIME_WAITED ,
Round(100*Total_Waits/Time_Waited) Rate,
Round(10* Time_Waited/Total_Waits,1) Latency
From Gv$System_Event
Where Event Like '%db file sequential read%'
And Inst_Id In (1,2,3,4,5)
Order By 1;
```

```
select *
from
(select
     sql_text,
     sql_id,
     elapsed_time,
     cpu_time,
     user_io_wait_time
  from
     sys.v_$sqlarea
  order by 5 desc)
where rownum < 6;
```

```
select metric_name, round(value,2) from v$sysmetric
where metric_name in ('Database CPU Time Ratio', 'Database Wait Time Ratio')
and intsize_csec = (select max(INTSIZE_CSEC) FROM V$SYSMETRIC);
```

```
BEGIN
DBMS_WORKLOAD_REPOSITORY.CREATE_SNAPSHOT ();
END;
/
```

```
SELECT *
     FROM
       (SELECT status,
         --username,
         sql_id,
         sql_exec_id,
         TO_CHAR(sql_exec_start,'dd-mon-yyyy hh24:mi:ss') AS sql_exec_start,
         ROUND(elapsed_time/1000000)                    AS "Elapsed (s)",
         ROUND(cpu_time    /1000000)                    AS "CPU (s)",
         buffer_gets,
         ROUND(physical_read_bytes /(1024*1024)) AS "Phys reads (MB)",
         ROUND(physical_write_bytes/(1024*1024)) AS "Phys writes (MB)"
       FROM v$sql_monitor
       ORDER BY elapsed_time DESC
       )
     WHERE rownum<=20;
```

```
(select s.sql_id, RANK() OVER (ORDER BY
(max(s.CPU_TIME_TOTAL/s.executions_total)) DESC) cpu_rank,
RANK() OVER (ORDER BY
select
  SQL_ID
, PLAN_HASH_VALUE
, sum(EXECUTIONS_DELTA) EXECUTIONS
, sum(ROWS_PROCESSED_DELTA) CROWS
, trunc(sum(CPU_TIME_DELTA)/1000000/60) CPU_MINS
, trunc(sum(ELAPSED_TIME_DELTA)/1000000/60)  ELA_MINS
from DBA_HIST_SQLSTAT
where SQL_ID in (
'&sqlid')
group by SQL_ID , PLAN_HASH_VALUE
order by SQL_ID, CPU_MINS;
```

http://uralural.blogspot.com/2007/07/oracle-cbo-chooses-wrong-index-although.ht

```
set line 132
set pagesize 33

ttitle " ***** Database: "db", DataFile's Disk Activity (As of:" tdate " ) *****"

select substr(df.file#,1,2) "ID",
rpad(substr(name,1,52),52,'.') "File Name",
rpad(substr(phyrds,1,10),10,'.') "Phy Reads",
rpad(substr(phywrts,1,10),10,'.') "Phy Writes",
```

```
SELECT sid,
       username,
       event,
       total_waits,
       100 * round((total_waits / sum_waits),2) pct_of_total_waits,
       time_wait_sec,
       total_timeouts,
       average_wait_sec,
       max_wait_sec
```

```
select
   event,
   sql_id,
   snap_id,
   dbid,
   instance_number,
   sample_time,
   session_id,
   session_serial#,
   user_id,
   blocking_session,
   program,
   (
      select
         sql_text
      from
         dba_hist_sqltext
      where
         sql_id = dba_hist_active_sess_history.sql_id
   ) sql_text
from
   dba_hist_active_sess_history
where
   program = 'some.exe'
and
   sample_time between
      to_timestamp('START_TIME', 'YYYY-MM-DD HH24:MI:SS.FF3') and
      to_timestamp('END_TIME', 'YYYY-MM-DD HH24:MI:SS.FF3')
and
   event = 'some wait event';
```

```
Select Inst_Id, EVENT, TOTAL_WAITS, TIME_WAITED ,
Round(100*Total_Waits/Time_Waited) Rate,
Round(10* Time_Waited/Total_Waits,1) Latency
From GV$System_Event
Where Event Like '%db file sequential read%'
And Inst_Id In (1)
Order By 1;
```

http://gavinsoorma.com/2012/11/ash-and-awr-performance-tuning-scripts/

http://hourim.wordpress.com/2012/09/14/tuning-by-tkprof-a-case-study/

https://sites.google.com/site/embtdbo/wait-event-documentation

http://www.oraclerealworld.com/best-oracle-performance-tools/

```
select distinct
p.name sql_profile_name,
s.sql_id
from
dba_sql_profiles p,
DBA_HIST_SQLSTAT s
where
p.name=s.sql_profile;
```

```
col data_type for a20
col column_name for a20
select table_name,column_name,data_type,num_distinct,num_nulls,
sample_size, histogram,num_buckets
from dba_tab_columns where table_name = '&TABLE_NAME' AND
OWNER='&OWNER';
```

http://nadvi.blogspot.com/2011/10/gather-optimizer-statistics-11g_26.html

```
exec DBMS_STATS.GATHER_TABLE_STATS (ownname => '&OWNER' , tabna
exec  dbms_stats.gather_index_stats ( ownname => '&OWNER', indname => '&I
exec DBMS_STATS.GATHER_SCHEMA_STATS (ownname => '&OWNER' , est
EXEC DBMS_STATS.GATHER_DATABASE_STATS(ESTIMATE_PERCENT =>
exec dbms_stats.gather_schema_stats ('SYS');
exec dbms_stats.gather_database_stats (gather_sys=>TRUE);
exec dbms_stats.gather_dictionary_stats;
exec dbms_stats.gather_fixed_objects_stats;
```

Table

```
set lines 1000 set pages 9999
col table_name for a40
col owner for a30
col partition_name for a40
col subpartition_name for a40
select distinct owner, table_name,partition_name,subpartition_name,
num_rows,blocks,stale_stats, TO_CHAR(last_analyzed,'dd-mon-yyyy
hh24:mi:ss') AS last_analyzed, stattype_locked
  from dba_tab_statistics
  where owner='&owner' and table_name in ('&tab1','&tab2','&tab3') order by
2,3,4;
 select
table_name,STATUS,NUM_ROWS,SAMPLE_SIZE,to_char(LAST_ANALYZED,'
DD-MON-YYYY hh24:mi:ss') from dba_tables
  where table_name in ('&tab1','&tab2','&tab3') and owner='&schema';
```

```
select n.object_name as
T,o.ROWCNT,o.BLKCNT,to_char(o.SAVTIME,'dd/mm/yyyy HH:MI:SS') as time
from dba_objects n,sys.WRI$_OPTSTAT_TAB_HISTORY o where
o.obj#=n.object_id and n.object_name='&Object_name' and n.owner='&owner'
order by time;
select 'execute DBMS_STATS.GATHER_SCHEMA_STATS('''||username||''') ;'
from dba_users  where username <> 'SYS';
```

```
col TABLE_NAME for a30
col PARTITION_NAME for a20
col SUBPARTITION_NAME for a20
select
OWNER,TABLE_NAME,PARTITION_NAME,SUBPARTITION_NAME,NUM_ROWS,LAST_ANALYZED from dba_TAB_STATISTICS where
STALE_STATS='YES';
SELECT CLIENT_NAME, STATUS FROM DBA_AUTOTASK_CLIENT WHERE
CLIENT_NAME='auto optimizer stats collection';
```

```
SELECT TABLES.OWNER, TABLES.TABLE_NAME,
ROUND((DELETES + UPDATES + INSERTS)/NUM_ROWS*100)
PERCENTAGE
FROM DBA_TABLES TABLES, DBA_TAB_MODIFICATIONS MODIFICATIONS
WHERE TABLES.OWNER = MODIFICATIONS.TABLE_OWNER
AND TABLES.TABLE_NAME = MODIFICATIONS.TABLE_NAME AND
NUM_ROWS > 0
AND ROUND ( (DELETES + UPDATES + INSERTS) / NUM_ROWS * 100) >=
10
ORDER BY 3 desc
/
SET ECHO OFF
SET TERMOUT ON
SET SERVEROUTPUT ON
SET TIMING OFF
DECLARE
  v1  varchar2(100);
  v2  varchar2(100);
  v3  varchar2(100);
  v4  varchar2(100);
```

DBMS_STATS.RESTORE_DICTIONARY_STATS - Used to restore data dictionary stats

DBMS_STATS.RESTORE_FIXED_OBJECTS_STATS - Used to restore fixed object stats

DBMS_STATS.RESTORE_SCHEMA_STATS - Used to restore schema stats

DBMS_STATS.RESTORE_SYSTEM_STATS - Used to restore system stats

DBMS_STATS.RESTORE_TABLE_STATS - Used to restore table stats

```
exec dbms_stats.gather_table_stats('DBUSER01','TABLE01',
estimate_percent=>33, cascade => TRUE, method_opt => 'FOR ALL INDEXED
COLUMNS SIZE 10');
```

```
COL owner FOR A30
COL table_name FOR A30
COL column_name FOR A30
col LAST_ANALYZED for a30
SELECT OWNER,table_name,column_name, num_distinct,
num_buckets,NUM_NULLS,DENSITY,to_char(LAST_ANALYZED, 'dd-mon-yyyy
hh24:mi:ss') last_analyzed ,GLOBAL_STATS, histogram
      FROM DBA_TAB_COL_STATISTICS
     WHERE owner='&OWNER' and  table_name = '&TABLE_NAME';
begin
   dbms_stats.gather_schema_stats(
      ownname          => 'SCOTT',
      estimate_percent => dbms_stats.auto_sample_size,
      method_opt       => 'for all columns size auto',
      degree           => 7
   );
end;
/

begin
 dbms_stats.set_table_stats('TESTER'
 , 'TEST1'
 , numrows=>10000
 , numblks=>174
 , avgrlen=>110);
end;
/
select * from table(dbms_stats.diff_table_stats_in_history(
                         ownname => '&OWNER',
                         tabname => upper('&tabname'),
                         time1 => systimestamp,
                         time2 => to_timestamp('&time2','yyyy-mm-dd:hh24:mi:ss'),
                         pctthreshold => 0));
```

```
set lines 300 set pages 300
col table_name for a40
col owner for a30
select distinct owner, table_name, STALE_STATS, last_analyzed,
stattype_locked
  from dba_tab_statistics
  where (owner, table_name) in
  (select distinct owner, table_name
          from dba_tables
          where ( table_name)
          in ( select object_name
                  from gv$sql_plan
                  where upper(sql_id) = upper('&sql_id') and object_name is not null))
  --and STALE_STATS='YES'
/
SELECT NAME, SQL_TEXT, CATEGORY, STATUS
FROM   DBA_SQL_PROFILES;

SELECT sql_handle, plan_name, enabled, accepted
FROM   dba_sql_plan_baselines
WHERE sql_text NOT LIKE '%dba_sql_plan_baselines%';



SELECT session_id,
       COUNT(*)
FROM   v$active_session_history
WHERE  session_state = 'ON CPU'
       AND sample_time > sysdate - ( 5 / ( 24 * 60 ) )
GROUP  BY session_id
ORDER  BY COUNT(*) DESC;
SELECT session_id,
       COUNT(*)
FROM   v$active_session_history
WHERE  session_state = 'WAITING'
       AND sample_time > sysdate - ( 5 / ( 24 * 60 ) )
GROUP  BY session_id
ORDER  BY COUNT(*) DESC;
SELECT event,
       COUNT(*)
FROM   v$active_session_history
WHERE  session_state = 'WAITING'
       AND sample_time > sysdate - ( 5 / ( 24 * 60 ) )
GROUP  BY event
ORDER  BY COUNT(*) DESC;
```

```
select
ash.session_id,
ash.session_serial#,
ash.user_id,
ash.program,
sum(decode(ash.session_state,'ON CPU',1,0)) "CPU",
sum(decode(ash.session_state,'WAITING',1,0)) -
sum(decode(ash.session_state,'WAITING',
decode(en.wait_class,'User I/O',1, 0 ), 0)) "WAITING" ,
sum(decode(ash.session_state,'WAITING',
decode(en.wait_class,'User I/O',1, 0 ), 0)) "IO" ,
sum(decode(session_state,'ON CPU',1,1)) "TOTAL"
from v$active_session_history ash,
v$event_name en
where en.event# = ash.event# AND SAMPLE_TIME >  SYSDATE - (5/(24*60))
group by session_id,user_id,session_serial#,program
order by sum(decode(session_state,'ON CPU',1,0));
SELECT ash.sql_id,
    SUM(DECODE(ash.session_state, 'ON CPU', 1, 0))     "CPU",
    SUM(DECODE(ash.session_state, 'WAITING', 1, 0))
         - SUM( DECODE(ash.session_state, 'WAITING',
DECODE(en.wait_class, 'User I/O', 1, 0), 0)) "WAIT",
    SUM(DECODE(ash.session_state, 'WAITING',  DECODE(en.wait_class,
'User I/O', 1, 0),  0))     "IO",
    SUM(DECODE(ash.session_state, 'ON CPU', 1,   1))     "TOTAL"
FROM   v$active_session_history ash,
    v$event_name en
WHERE  sql_id IS NOT NULL AND SAMPLE_TIME >  SYSDATE - (5/(24*60))
    AND en.event# = ash.event#
GROUP  BY sql_id
ORDER  BY SUM(DECODE(session_state, 'ON CPU', 1, 0)) DESC;
```

```
select
   sql_id,
   child_number,
   sql_text,
   elapsed_time
from
   (select
      sql_id,child_number,
      sql_text,
      elapsed_time,
      cpu_time,
      disk_reads,
   rank ()
   over
      (order by elapsed_time desc)
   as
      sql_rank
   from
      gv$sql)
where
   sql_rank < 10;


SET LINESIZE 1000
SET PAGESIZE 100
SELECT * FROM
(SELECT SUBSTR(SQL_TEXT,1,40) SQL,
BUFFER_GETS, EXECUTIONS, BUFFER_GETS/EXECUTIONS
"GETS/EXEC",
HASH_VALUE,ADDRESS
FROM V$SQLAREA
WHERE BUFFER_GETS > 10000
ORDER BY BUFFER_GETS DESC)
WHERE ROWNUM <= 10;

 SELECT * FROM
(SELECT SUBSTR(SQL_TEXT,1,40) SQL,
DISK_READS, EXECUTIONS, DISK_READS/EXECUTIONS "READS/EXEC",
HASH_VALUE,ADDRESS
FROM V$SQLAREA
WHERE DISK_READS > 1000
ORDER BY DISK_READS DESC)
WHERE ROWNUM <= 10;
```

```
SELECT * FROM
(SELECT SUBSTR(SQL_TEXT,1,40) SQL,
EXECUTIONS, ROWS_PROCESSED, ROWS_PROCESSED/EXECUTIONS
"ROWS/EXEC",
HASH_VALUE,ADDRESS
FROM V$SQLAREA
WHERE EXECUTIONS > 100
ORDER BY EXECUTIONS DESC)
WHERE ROWNUM <= 10;

SELECT * FROM
(SELECT SUBSTR(SQL_TEXT,1,40) SQL,
PARSE_CALLS, EXECUTIONS, HASH_VALUE,ADDRESS
FROM V$SQLAREA
WHERE PARSE_CALLS > 1000
ORDER BY PARSE_CALLS DESC)
WHERE ROWNUM <= 10;

SELECT * FROM
(SELECT SUBSTR(SQL_TEXT,1,40) SQL,
SHARABLE_MEM, EXECUTIONS, HASH_VALUE,ADDRESS
FROM V$SQLAREA
WHERE SHARABLE_MEM > 1048576
ORDER BY SHARABLE_MEM DESC)
WHERE ROWNUM <= 10;

SELECT * FROM
(SELECT SUBSTR(SQL_TEXT,1,40) SQL,
VERSION_COUNT, EXECUTIONS, HASH_VALUE,ADDRESS
FROM V$SQLAREA
WHERE VERSION_COUNT > 20
ORDER BY VERSION_COUNT DESC)
WHERE ROWNUM <= 10;
```

http://orababy.blogspot.com/2013/08/active-session-history-queries.html

```
DECLARE
  l_sql_tune_task_id VARCHAR2(100);
BEGIN
  l_sql_tune_task_id := DBMS_SQLTUNE.create_tuning_task (
                          sql_id => '4pgqpkv31u4nn',
                          scope => DBMS_SQLTUNE.scope_comprehensive,
                          time_limit => 2100,
                          task_name => '4pgqpkv31u4nn_tuning_task',
                          description => 'Tuning task for statement d4pgqpkv31u4nn.');
  DBMS_OUTPUT.put_line('l_sql_tune_task_id: ' || l_sql_tune_task_id);
END;
/

EXEC DBMS_SQLTUNE.execute_tuning_task(task_name =>
'4pgqpkv31u4nn_tuning_task');

select task_name, status from dba_advisor_log where
task_name='4pgqpkv31u4nn_tuning_task';

SET SERVEROUTPUT ON
SET LONG 100000;
set longchunksize 1000;
SET PAGESIZE 1000
SET LINESIZE 200
SELECT DBMS_SQLTUNE.report_tuning_task('4pgqpkv31u4nn_tuning_task')
AS recommendations FROM dual;
SET PAGESIZE 24
```

```
  cpu/60 AS cpu,
  bcpu/60 AS bcpu,
  DECODE(SIGN((cpu+bcpu)/60-cpu_ora_consumed), -1, 0, ((cpu+bcpu)/60-
cpu_ora_consumed)) AS cpu_ora_wait,
  scheduler/60 AS scheduler,
  uio/60 AS uio,
  sio/60 AS sio,
```

```
with AASSTAT as (
           select
                 decode(n.wait_class,'User I/O','User I/O',
                                     'Commit','Commit',
                                     'Wait')                         CLASS,
                 sum(round(m.time_waited/m.INTSIZE_CSEC,3))                AAS
           from  v$waitclassmetric  m,
```

https://jonathanlewis.wordpress.com/2014/12/09/parse-time-2/

```
select ksppinm name,
       ksppstvl value
from   sys.x$ksppi x,
       sys.x$ksppcv y
where (x.indx = y.indx)
and   ksppinm = '_optimizer_aggr_groupby_elim';

alter session set optimizer_features_enable = '11.2.0.4';

select ksppinm name,
       ksppstvl value
from   sys.x$ksppi x,
       sys.x$ksppcv y
where (x.indx = y.indx)
and   ksppinm = '_optimizer_aggr_groupby_elim';

alter session set "_optimizer_aggr_groupby_elim"=true;
```

Optimizer Evolution

- Oracle 8i – histograms
- Oracle 9i – bind variable peeking
- Oracle 10g – Automatic Optimizer Statistics Collection,Dynamic Sampling
- Oracle 11gR1 – Adaptive Cursor Sharing, Extended statistics (correlated columns)
- Oracle 11gR2 – Cardinality Feedback
- Oracle 12c – Adaptive Query Optimization

```
declare
v_sql_text CLOB;
BEGIN
select sql_text into v_sql_text from v$sql where sql_id = 'bu18sp8k0wcvv';
sys.dbms_sqldiag_internal.i_create_patch(
sql_text=>v_sql_text,
hint_text=>'full(@SEL$1 test_dba_tables)',
 name=>'tst_patch');
END;
/
exec dbms_sqldiag.DROP_SQL_PATCH('tst_patch');
```

Advisor names
Enabled or not

Job History

Parameters

EXEC_ID for sql tuning advisors (History of Job run status)

Automatic Sql advisor Report

```
SELECT a.command AS type,
f.message AS findings,
a.message AS recommendations,
t.message AS rationale
FROM dba_advisor_actions a,
dba_advisor_recommendations r,
dba_advisor_findings f,
dba_advisor_rationale t
WHERE a.task_id = &task_id
AND a.task_id = r.task_id
AND a.rec_id = r.rec_id
AND a.task_id = t.task_id
AND a.rec_id = t.rec_id
AND f.task_id = r.task_id
AND f.finding_id = r.finding_id;
```

```
set lines 750 pages 9999
set long 9999
select dbms_sqltune.report_auto_tuning_task(
  (select min(execution_name) from dba_advisor_findings
    where task_name like 'SYS_AUTO_SQL%'),
  (select max(execution_name) from dba_advisor_findings
    where task_name like 'SYS_AUTO_SQL%')
) from dual;
```

```
select
   table_name,round((blocks*8),2) "size (kb)" ,
   round((num_rows*avg_row_len/1024),2) "actual_data (kb)",
   (round((blocks*8),2) - round((num_rows*avg_row_len/1024),2))
"wasted_space (kb)"
from
   dba_tables
where  owner='&OWNER' and table_name='&TABLE_NAME' and
   (round((blocks*8),2) > round((num_rows*avg_row_len/1024),2))
order by 4 desc;
```

https://tinky2jed.wordpress.com/technical-stuff/oracle-stuff/changing-the-oracle-da

```
select window_name, repeat_interval, duration from dba_scheduler_windows;

EXECUTE
DBMS_SCHEDULER.SET_ATTRIBUTE('WEEKNIGHT_WINDOW','repeat_inter
val','freq=daily;byday=MON,TUE,WED,THU,FRI;byhour=05;byminute=0;
bysecond=0');
EXECUTE
DBMS_SCHEDULER.SET_ATTRIBUTE('WEEKEND_WINDOW','repeat_interva
l',' freq=daily;byday=SAT;byhour=07;byminute=0;bysecond=0');

select client_name, job_status, job_start_time, job_duration from dba_autotask_jo

select client_name, status, window_group from dba_autotask_client;

select actual_start_date,run_duration,job_name,status,additional_info from
dba_scheduler_job_run_details where job_name like 'ORA$AT_OS%' order by
log_date;

select * from dba_scheduler_job_run_details where job_name like
'ORA$AT_OS%' order by log_date;
```

```
alter system set “_ash_disk_filter_ratio”=1.
```

https://blog.dbi-services.com/sockets-cores-virtual-cpu-logical-cpu-hyper-threadin
https://orastory.wordpress.com/page/2/

```
WITH subq_snaps AS
(SELECT dbid              dbid
 ,     instance_number     inst
 ,     snap_id            e_snap
 ,     lag(snap_id) over (partition by instance_number, startup_time order by
snap_id) b_snap
 ,     TO_CHAR(begin_interval_time,'D') b_day
 ,     TO_CHAR(begin_interval_time,'DD-MON-YYYY HH24:MI') b_time
 ,     TO_CHAR(end_interval_time,'HH24:MI')   e_time
 ,    ((extract(day    from (end_interval_time - begin_interval_time))*86400)
```

```
SELECT x.*, (SELECT sql_text from dba_hist_sqltext t where t.sql_id = x.sql_id
and rownum = 1) txt
FROM (
SELECT sn.snap_id
,      TO_CHAR(sn.end_interval_time,'DD-MON-YYYY HH24:MI') dt
,      st.sql_id
,      st.instance_number
,      st.parsing_schema_name
,      st.plan_hash_value
,      SUM(st.fetches_delta) fch
,      SUM(rows_processed_delta) rws
,      SUM(executions_delta)     execs
,      ROUND(SUM(elapsed_time_delta)/1000/1000)   elp
,
ROUND(SUM(elapsed_time_delta)/1000/1000/NVL(NULLIF(SUM(executions_d
elta),0),1),2)   elpe
,      ROUND(SUM(cpu_time_delta)/1000/1000)       cpu
,      SUM(buffer_gets_delta)    gets
,      ROUND(SUM(iowait_delta)/1000/1000)         io
,      ROUND(SUM(clwait_delta)/1000/1000)         cl
,      ROUND(SUM(ccwait_delta)/1000/1000)         cc
,      ROUND(SUM(apwait_delta)/1000/1000)         ap
,      ROUND(SUM(plsexec_time_delta)/1000/1000)   pl
,      ROUND(SUM(disk_reads_delta))         disk_reads
,      ROUND(SUM(direct_writes_delta))        direct_writes
,      ROW_NUMBER() over (PARTITION BY sn.dbid, sn.snap_id,
st.instance_number
                         ORDER BY SUM(elapsed_time_delta) desc) rn
FROM   dba_hist_snapshot sn
,      dba_hist_sqlstat  st
WHERE  st.dbid            = sn.dbid
AND    st.snap_id         = sn.snap_id
AND    sn.instance_number = st.instance_number
```

```
col short_name  format a20            heading 'Load Profile'
col per_sec     format 999,999,999.9   heading 'Per Second'
col per_tx      format 999,999,999.9   heading 'Per Transaction'
set colsep '   '

select lpad(short_name, 20, ' ') short_name
     , per_sec
     , per_tx from
    (select short_name
```

ta from v$sql_plan_statistics already mentioned in your article. In my opinion, plan statistics have
iggest obstacle to a wide adoption is the horrible slowness caused by high CPU overhead in
uction system. So, the execution of the query is staged in a separate session where we can set
n environment. Here is the full sequence of actions:

number of runway SQL from v$session;
n v$sql using SQL_ID and CHILD_NUMBER;
_bind_capture. There can be problems capturing bind variables, for brevity I will skip details. If no
ariables, skip this step;
n “alter session set statistics_level=all”;
, set same values and, even more importantly, same datatypes as in Capture step;
n (send break) after some time, even 10-15 secs should be sufficient;
l text was not changed, the SQL_ID should be same as in step 1. Very important: it should have
alue, otherwise revise previous steps;
cs is that it provides data even on partial execution. It can be a separate article dedicated to the
nt viewer, one should be able to see absolutely clear where the problem is. No guess work, it’s all
to get to the steps where it will be running for lo-o-ong time, they can be easily identified.

all that things. The point is, once I was free from all that routine, I was able to troubleshoot not only
ot of gets/execution. I should add that it makes sense to use this technique in really complicated
nple and a quick look at the execution plan is enough to spot the cause.

http://db.geeksinsight.com/2012/10/15/scripts-databasetabletablespace-growth-report-using-awr/

Hourly

```
set linesize 750 pages 9999
set numwidth 20
COL NAME FOR A30
col SNAP_ID for 9999999

set serveroutput on

 Declare
   v_BaselineSize    number(20);
   v_CurrentSize    number(20);
```

OS occupies memory of
----

20 % of RAM SIZE --> for windows

10 % of RAM SIZE --> for UNIX/LINUX

For merging files

```
cat *.xml >> file.log
```

VE_INSTANCE_COUNT (number of active instances) http://www.oracle.com/technetwork/articles/schumacher-analysis-099313.html

```
col STAT_NAME for a20
col VALUE_DIFF for 9999,999,999
col STAT_PER_MIN for 9999,999,999
set lines 200 pages 1500 long 99999999
col BEGIN_INTERVAL_TIME for a30
col END_INTERVAL_TIME for a30
set pagesize 40
set pause on


select hsys.SNAP_ID,
       hsnap.BEGIN_INTERVAL_TIME,
       hsnap.END_INTERVAL_TIME,
           hsys.STAT_NAME,
           hsys.VALUE,
           hsys.VALUE - LAG(hsys.VALUE,1,0) OVER (ORDER BY hsys.SNAP_ID) AS
"VALUE_DIFF",
           round((hsys.VALUE - LAG(hsys.VALUE,1,0) OVER (ORDER BY hsys.SNAP_ID)) /
           round(abs(extract(hour from (hsnap.END_INTERVAL_TIME -
hsnap.BEGIN_INTERVAL_TIME))*60 +
           extract(minute from (hsnap.END_INTERVAL_TIME - hsnap.BEGIN_INTERVAL_TIME)) +
           extract(second from (hsnap.END_INTERVAL_TIME -
hsnap.BEGIN_INTERVAL_TIME))/60),1)) "STAT_PER_MIN"
from dba_hist_sysstat hsys, dba_hist_snapshot hsnap
 where hsys.snap_id = hsnap.snap_id
 and hsnap.instance_number in (select instance_number from v$instance)
 and hsnap.instance_number = hsys.instance_number
 and hsys.STAT_NAME='user commits'
 order by 1;
```

Instead of creating new table u can try

```
alter table table_name move;
```

It will remove the contention

In a test environment, you can measure shareable memory by selecting the number of open cursors for a test user. You multiply the resulting value by the total number of users:

Ideally, your application should have a library cache as large as the sum of the numbers above, plus a small allowance for dynamic SQL.

This is useful when calculating shared pool size

- Reloads should be less than 1% of the pins:
- If the reloads-to-pins ratio is greater than 1%, increase the value of the SHARED_POOL_SIZE parameter.

When Invalidations Occur ?

when a table, sequence, synonym, or view is re-created or altered or dropped, or
a procedure or package specification is recompiled, all dependent shared SQL areas are invalidated.

Additional V$SQL_PLAN columns not found in PLAN_TABLE:
- ADDRESS: Cursor parent handle address
- HASH_VALUE: Parent statement hash value in library cache
- CHILD_NUMBER: Number using this execution plan
- DEPTH: Level of the operation in the tree
- CPU_COST: CPU cost of the operation as estimated by the cost-based optimizer. If using the rule-based optimizer, this column is null.
- IO_COST: Cost of the operation as estimated by the cost-based optimizer. If using the rule-based optimizer, this column is Null.
- TEMP_SPACE: Space usage of sort or hash-join estimated by cost-based optimizer

- REQUEST_MISS = 0 or not increasing
- FREE_MEMORY = > 50% of the SHARED_POOL_RESERVED_SIZE minimum

How to Keep Objects ?
Use the supplied DBMS_SHARED_POOL package and the KEEP procedure to keep objects.
To create the package, run the dbmspool.sql script. The prvtpool.plb script is automatically executed at the end of the previous one. These scripts are not run by catproc.sql.
Use the UNKEEP procedure to remove pinned objects from the shared pool.

First, the DBA determines the change in the hit ratio as buffers are added or removed. As a general rule, increase buffer cache size if:

- The cache hit ratio is less than 90%
- There is adequate memory for other processes without inducing additional page faults

Hit ratio is affected by data access methods:

- Full table scans
- Data or application design
- Large table with random access
- Uneven distribution of cache hits

The V$SESSION_WAIT view indicates through the Log Buffer Space event if there are any waits for space in the log buffer because the session is writing data into the log buffer faster than LGWR can write it out.

Note: The V$SYSSTAT view displays another statistic, Redo Log Space Requests:

• The Redo Buffer Allocation Retries value should be near
0; the number should be less than 1% of redo entries.

– In the V$SYSTEM_EVENT view, check the number of occurrences of the event Log File Switch Completion, which identifies the log file switch waits because of log switches.

– In the V$SYSTEM_EVENT view, check the number of occurrences of the event Log File Switch (Checkpoint Incomplete), which identifies the log file switch waits because of incomplete checkpoints.

– In the V$SYSTEM_EVENT view, check the number of the occurrences of the event Log File Switch (Archiving Needed), which identifies the log file switch waits because of the archiving issue.

- Turn asynchronous I/O on or off with:

– DISK_ASYNCH_IO
– TAPE_ASYNCH_IO

Multiple DBWn processes can be deployed with
DB_WRITER_PROCESSES (DBW0 to DBW9).

Tuning DBWn I/O
Tune the DBWn by looking at the value of the
following event:
FREE BUFFER WAITS

Consider increasing the DBWn processes, if you see a high number of free_buffer_waits after querying the V$SYSTEM_EVENT view as in the following syntax:

Note: The last two columns contain 0 unless the TIMED_STATISTICS parameter is set to TRUE.
Use the following query to monitor these values:

Avoiding Sorts

Avoid sort operations whenever possible:

- Use NOSORT to create indexes.
- Use UNION ALL instead of UNION.
- Use index access for table joins.
- Create indexes on columns referenced in the ORDER BY clause.
- Select the columns for analysis.
- Use ESTIMATE rather than COMPUTE for large objects.

Create indexes on columns that are frequently referenced with ORDER BY statements.

```
SQL> select disk.value "Disk", mem.value "Mem",
 (disk.value/mem.value)*100 "Ratio"
 from v$sysstat mem, v$sysstat disk
 where mem.name = 'sorts (memory)'
 and disk.name = 'sorts (disk)';

Disk Mem Ratio
--------- --------- ---------
23 206 11.165049
```

- The number of waits for any class should be less than 1% of the total number of requests.
- If not, create more rollback segments.

- Deletes are expensive.TRUNCATE instead, performance is improved.
- Inserts use minimal rollback space.
- Updates use rollback space, depending on the
number of columns.
- Index maintenance adds rollback.

Using Less Rollback

- Export / Import operations

– Import
– Set COMMIT = Y
– Size the set of rows with the BUFFER keyword
– Export: Set CONSISTENT=N

- SQL*Loader operations: Set the commit intervals with ROWS

You can use the following query to set the UNDO_RETENTION parameter and size the undo tablespace:

You can increase the overall number of locks available for an instance by increasing the values of the DML_LOCKS and ENQUEUE_RESOURCES parameters. This may be necessary in a parallel server configuration.

V$LOCK
V$LOCKED_OBJECT
DBA_WAITERS
DBA_BLOCKERS

To find the table name that corresponds to a particular resource ID 1 of the V$LOCK view:

Query the V$DISPATCHER view to determine the usage for selected dispatcher processes.
You identify contention for dispatchers by checking:

A query to report the dispatcher, session, and process mapping using shared servers:

Maintenance Considerations

In a data warehousing environment, data is usually maintained by way of bulk inserts and updates. Index maintenance is deferred until the end of each DML operation. For example, if you insert 1,000 rows, then the inserted rows are placed into a sort buffer, and then the updates of all 1,000 index entries are batched. (This is why SORT_AREA_SIZE must be set properly for good performance with inserts and updates on bitmap indexes.) Thus, each bitmap segment is updated only once per DML operation, even if more than one row in that segment changes.

Typically, if 15% of the index data is deleted, then you
should consider rebuilding the index.

Identifying Unused Indexes

- To start monitoring the usage of an index:

```
ALTER INDEX HR. EMP_NAME_IX MONITORING USAGE;
```

- To stop monitoring the usage of an index:

```
ALTER INDEX HR. EMP_NAME_IX NOMONITORING USAGE;
```

- To query the usage of the index:

- Larger extents can have a small performance benefit because the Oracle server can read one large extent from disk with fewer multiblock reads than would be required to read many small extents. To avoid partial multiblock reads, set the extent size to a multiple of 5 × DB_FILE_MULTIBLOCK_READ_COUNT. Multiply by five because the Oracle server tries to allocate extents on five-block boundaries. By matching extent sizes to the I/O and space allocation sizes, the performance cost of having many extents in a segment is minimized. However, for a table that never has a full table scan operation, it makes no difference in terms of query performance whether the table has one extent or multiple extents.

The High-Water Mark

- Reset by the TRUNCATE command
- Never reset by DELETE statements

The High-Water Mark
Space above the high-water mark can be reclaimed at the table level by using the following command:
ALTER TABLE <table_name> DEALLOCATE UNUSED...

Move the table

After the table is moved, all indexes are marked unusable, and must be rebuilt.

The DB_BLOCK_SIZE Parameter
The database block size:

- Is defined by the DB_BLOCK_SIZE parameter
- Is set when the database is created
- Is the minimum I/O unit for data file reads
- Is 2 KB or 4 KB by default, but up to 64 KB is
allowed
- Cannot be changed easily
- Should be an integer multiple of the OS block size
- Should be less than, or equal to, the OS I/O size

- Query plan_table to display the execution plans:
– Query PLAN_TABLE directly
– Use script utlxpls.sql (Hide Parallel Query information)
– Use script utlxplp.sql (Show parallel Query information)

Hot Segments under Specific Schema & type=table

```
 select * from
(Select
ob.owner, ob.object_name, sum (b.tch) Touchs
from x$bh b, dba_objects ob
where b.obj = ob.data_object_id
and b.ts#> 0 and ob.object_type='TABLE' and ob.owner='&owner'
group by ob.owner, ob.object_name
order by sum (tch) desc)
where rownum <= 10 ;
```

The blocking session is:

```
select blocking_session, sid, serial#, wait_class, seconds_in_wait from v$session where
blocking_session is not NULL order by blocking_session;
```

http://www.oaktable.net/content/oracle-cpu-time

```
with epsilon
as
(select avg(average - STANDARD_DEVIATION ) m1,
        avg(average +  STANDARD_DEVIATION ) m2
from dba_hist_sysmetric_summary
where metric_name='User Calls Per Sec')
select avg(a.average -  a.STANDARD_DEVIATION) "A - Good",
       avg(a.average) "Average",
       avg(a.average + a.STANDARD_DEVIATION)  "B - Bad"
from dba_hist_sysmetric_summary a,
dba_hist_sysmetric_summary b,
epsilon e
where a.metric_name='SQL Service Response Time'
and b.metric_name='User Calls Per Sec'
and a.snap_id = b.snap_id
and b.average between e.m1 and e.m2
/
```

```
d FROM v$session WHERE state = 'WAITING' AND wait_class!= 'Idle' ;
```

```
v$session_wait_class WHERE sid=161;
```

```
v$system_wait_class;
```

```
tric_unit LIKE '\%%' ESCAPE '\' GROUP BY metric_name, metric_unit;
```

Latency should be always within 5-10 ms ..If it exceeds then it is a issue http://blog.yannickjaquier.com/oracle/real-time-sql-monitoring.html

```
me => '&TABLE_NAME',cascade => true, estimate_percent => dbms_stats.auto_sample_size,me
NDEX_NAME',estimate_percent => dbms_stats.auto_sample_size, degree  => 8);
imate_percent => dbms_stats.auto_sample_size ,method_opt=>'FOR ALL COLUMNS SIZE AUTC
 DBMS_STATS.AUTO_SAMPLE_SIZE, METHOD_OPT => 'FOR ALL COLUMNS SIZE AUTO', D
```

Partition

```
set lines 1000 pages 9999
col table_owner for a30
col table_name for a50
col partition_name for a50
select  table_owner,table_name, partition_name, global_stats, TO_CHAR(last_analyzed,'dd-mon-yyyy hh24:mi:ss') AS last_analyzed, num_rows
   from dba_tab_partitions
   where table_name='&partition_name'
   and table_owner='&table_owner'
   order by 1, 2, 4 desc nulls last;
```

```
set serveroutput on

DECLARE
ObjList dbms_stats.ObjectTab;
BEGIN
dbms_stats.gather_database_stats(objlist=>ObjList, options=>'LIST STALE');
FOR i in ObjList.FIRST..ObjList.LAST
LOOP
dbms_output.put_line(ObjList(i).ownname || '.' || ObjList(i).ObjName || ' ' || ObjList(i).ObjType || ' ' ||
ObjList(i).partname);
END LOOP;
END;
/
```

```
COL owner FOR A30
COL table_name FOR A30
COL column_name FOR A30

SELECT
  owner
  table_name,
  column_name, count(*)
FROM
  dba_tab_histograms
WHERE owner = '&OWNER'
and table_name='&TABLE_NAME'
GROUP BY owner,table_name, column_name
ORDER BY count(*) ASC
/
```

```
select blocks allocated_blks,
 count( distinct substr(t.rowid,1,8)
              ||substr(t.rowid,15,4)) used
      from user_segments e,
      &tab_name t
      where e.segment_name=upper ('&tab_name')
      and e.segment_type='TABLE'
      group by e.blocks;
```

```
SELECT *
FROM
TABLE(DBMS_XPLAN.display_sql_plan_baseline(plan_name=>'SYS_SQLPROF_015fdc7edace
0003'));
```

http://rohitsinhago.blogspot.com/2009/05/tracing-other-sessions.html

https://alexzeng.wordpress.com/2012/09/25/valuate-oracle-awr-automatic-workload-repository/

```
  SELECT sql_id,child_number,sql_text, elapsed_time
 FROM (SELECT sql_id, child_number, sql_text, elapsed_time, cpu_time,
             disk_reads,
             RANK () OVER (ORDER BY elapsed_time DESC) AS elapsed_rank
        FROM gv$sql)
WHERE elapsed_rank <= 10;
```

```
exec
dbms_sqltune.alter_sql_profile('SYS_SQLPROF_0236d8f37a490001','STATUS','DISABLED');

exec dbms_sqltune.alter_sql_profile('SQL_PROF_FOR_DELETE','STATUS','ENABLED');

EXEC DBMS_SQLTUNE.DROP_SQL_PROFILE('SYS_SQLPROF_02384d2b5e320000');
```

https://blog.yannickjaquier.com/oracle/visualizing-active-session-history-ash-to-produce-grid-contr https://blogs.oracle.com/optimizer/setting-a-session-parameter-overrides-ofe

https://prasanthkothuri.files.wordpress.com/2015/06/who-changed-my-plan.pdf

https://prasanthkothuri.wordpress.com/2015/06/08/sql-patch-fast-way-to-add-a-hint-without-changi

```
SELECT * FROM dba_advisor_definitions;
SELECT client_name, status, max_duration_last_30_days FROM DBA_AUTOTASK_CLIENT;
  SELECT job_start_time, job_status, job_duration,WINDOW_NAME FROM
DBA_AUTOTASK_JOB_HISTORY
     WHERE client_name='sql tuning advisor'
     AND job_start_time >= SYSDATE -7
     ORDER BY job_start_time DESC;
```

```
SET lines 750 pages 9999
col DESCRIPTION FOR a150
col parameter_name for a50
col parameter_value FOR a15
SELECT parameter_name,parameter_value,is_default,description
     FROM DBA_ADVISOR_PARAMETERS
     WHERE task_name='SYS_AUTO_SQL_TUNING_TASK'
     ORDER BY parameter_name;
set lines 750 pages 9999
col execution_name for a15
col TASK_NAME for a30
  col ERROR_MESSAGE for a100

  SELECT execution_name,task_id,task_name,TO_CHAR(execution_start,'dd-mon-yyyy
hh24:mi:ss') AS execution_start,TO_CHAR(execution_end,'dd-mon-yyyy hh24:mi:ss') AS
execution_end,
     status,error_message    FROM DBA_ADVISOR_EXECUTIONS
     WHERE task_name='SYS_AUTO_SQL_TUNING_TASK'
     ORDER BY execution_id DESC;
SET LONG 999999999
SELECT
DBMS_AUTO_SQLTUNE.REPORT_AUTO_TUNING_TASK('&EXEC_ID_MIN','&EXEC_ID_MAX
','TEXT','ALL','SUMMARY') FROM dual;
```

(or) Most Recent

```
SET LONG 999999999
SELECT
DBMS_AUTO_SQLTUNE.REPORT_AUTO_TUNING_TASK(NULL,NULL,'TEXT','ALL','SUMMA
RY') FROM dual;
```

https://timurakhmadeev.wordpress.com/2012/02/21/load-profile/

http://www.jamesmorle.com/the-oracle-wait-interface-is-useless-sometimes-pt/

https://anandmandilwar.com/2016/11/09/collection-of-useful-script-for-oracle-apps-dba/

https://www.programering.com/a/MTM3QDNwATM.html http://db.geeksinsight.com/2012/10/15/scripts-databasetabletablespace-growth-report-using-awr/

https://grepora.com/2016/05/25/oracle-tps-evaluating-transaction-per-second/

http://dbaclass.com/article/find-user-commits-per-minute-oracle-database/

The V$SHARED_POOL_RESERVED fixed table can also indicate when the value for show parameter db_cache_size;

- Initial parameter values:

– SGA_MAX_SIZE = 128M,
- DB_CACHE_SIZE = 96M
– SHARED_POOL_SIZE = 32M

```
 col owner for a30
 col object_name for a50
SELECT Rownum AS Rank,
       Seg_Lio.*
  FROM (SELECT St.Owner,
               St.Obj#,
               St.Object_Type,
               St.Object_Name,
               St.VALUE,
               'LIO' AS Unit
         FROM V$segment_Statistics St
        WHERE St.Statistic_Name = 'logical reads'
        ORDER BY St.VALUE DESC) Seg_Lio
```

https://blog.pythian.com/do-you-know-if-your-database-slow/ https://workwiththebest.intraway.com/blog-post/how-to-quickly-tell-if-your-oracle-database-is-running-slow-or-not/	https://www.oracle.com/technetwork/articles

```
Select /*+Parallel(X 10) */ Snap_Id,
Round(Sum(Singleblkrds)/1000000,1)
Megreads, Round(Sum(Singleblkrdtim)*10) Ms
From Dba_Hist_Filestatxs X
Where Snap_Id > &awr_id
Group By Snap_Id),
--
S2 As
(Select Snap_Id, Megreads - Lag(Megreads,1)
Over(Order By Snap_Id) Totrds,
Ms- Lag(Ms,1) Over (Order By Snap_Id) Tot_Ms
From S1 )
```

For over a period using awr snap_id

It should be less than < 8 ms

```
thod_opt=>'FOR ALL COLUMNS SIZE AUTO', granularity => 'ALL', degree => 8);

)', degree => 8);
EGREE => 8, CASCADE => TRUE);
```

Subpartition

```
set lines 1000 pages 9999
col table_owner for a30
col table_name for a50
col subpartition_name for a50
select  table_owner,table_name, subpartition_name,
global_stats, TO_CHAR(last_analyzed,'dd-mon-yyyy
hh24:mi:ss') AS last_analyzed, num_rows
    from dba_tab_subpartitions
    where table_name='&subpartition_name'
    and table_owner='&table_owner'
    order by 1, 2, 4 desc nulls last;
```

(Doc ID 457666.1)

https://community.toadworld.com/platforms/oracle/w/wiki/5477.recovering-previous-statistics

<-- displays the plan_name recommendation in details

This link is having performance tunign of CERN nuclear system , having manual sql profile creations

ng-the-code/

```
SET LINESIZE 80 PAGESIZE 1000 LONG 100000
SELECT
DBMS_SQLTUNE.REPORT_TUNING_TASK('SYS_AUTO_S
QL_TUNING_TASK') FROM DUAL;
```

/schumacher-analysis-099313.html

Database

Database Size

Table

Table size in MB

Each Table Size under a Schema

LOB Segment size in a TABLE Column wise

Table Partition size

Table subpartition size

Schema

Schema size in GB

Each Schema Size in MB

Tablespace

Tablespace Free

datafiles under tablespace

Max datafile limit formula

Temp Tablespace Usage

Db block size

What is the minimum size we can reduce for a datafile

How much size we can reduce a datafile

List the size of Oracle stored procedures

Top 10 large tables

INDEX larger than Table size for rebuilding

Index size,status,analyzed for a single table

Extent That going to get failed in Next Extent

Table fragmentation

LOB Space Allocations- Partition

CDB

```
select a.data_size+b.temp_size+c.redo_size+d.controlfile_size
"total_size in MB" from ( select sum(bytes)/1024/1024 data_size
from dba_data_files) a,( select nvl(sum(bytes),0)/1024/1024
temp_size from dba_temp_files ) b,( select sum(bytes)/1024/1024
redo_size from sys.v_$log ) c,
( select sum(BLOCK_SIZE*FILE_SIZE_BLKS)/1024/1024
controlfile_size from v$controlfile) d;
```

Without LOB

```
select   segment_name,sum(bytes)/(1024*1024) "TABLE_SIZE(MB)"
from dba_extents where  segment_type='TABLE' and
owner=upper('&OWNER') and segment_name =upper(
'&SEGMENT_NAME') group by segment_name;
   segment_name           table_name,
   sum(bytes)/(1024*1024) table_size_meg
from
COLUMN owner FORMAT A30
COLUMN table_name FORMAT A30
COLUMN column_name FORMAT A30
owner,segment_name,partition_name,segment_type,bytes/1024/1024
"MB" FROM dba_segments WHERE segment_type = 'TABLE
PARTITION' and segment_name= '&TABLE_NAME' and
owner,segment_name,partition_name,segment_type,bytes/1024/1024
"MB" FROM dba_segments WHERE segment_type = 'TABLE
SUBPARTITION' and segment_name= '&TABLE_NAME' and
```

```
SELECT  owner,Sum(bytes)/1024/1024/1024 AS total_size_gb
FROM dba_segments
WHERE owner =upper('&1') group by  owner;
select owner,sum(bytes/1024/1024) ||'M' from dba_segments group by
owner order by sum(bytes/1024/1024) desc;
```

```
col tablespace_name format a30
col TABLESPACE_NAME heading "Tablespace|Name"
col Allocated_size heading "Allocated|Size(GB)" form 99999999.99
col Current_size heading "Current|Size(GB)" form 99999999.99
col Used_size heading "Used|Size(GB)" form 99999999.99
col Available_size heading "Available|Size(GB)" form 99999999.99
col Pct_used heading "%Used (vs)|(Allocated)" form 99999999.99
select a.tablespace_name
        ,a.alloc_size/1024/1024/1024 Allocated_size
        ,a.cur_size/1024/1024/1024 Current_Size
select TABLESPACE_NAME,FILE_ID,FILE_NAME,BYTES/1024/1024
"size in
MB",AUTOEXTENSIBLE,MAXBYTES/1024/1024,ONLINE_STATUS
from dba_data_files where
TABLESPACE_NAME='&TABLESPACE_NAME';
```

```
ON MAXSIZE ' || to_char(round((bytes *
1.25)/(1024*1024*1024),0),9999) || 'G;'
  from dba_data_files
where bytes > maxbytes
order by tablespace_name;
SELECT   A.tablespace_name tablespace, D.mb_total,
         SUM (A.used_blocks * D.block_size) / 1024 / 1024 mb_used,
         D.mb_total - SUM (A.used_blocks * D.block_size) / 1024 / 1024
mb_free

column value new_val blksize
select value from v$parameter where name = 'db_block_size'
/
set linesize 1000 pagesize 0 feedback off trimspool on
with
 hwm as (
  -- get highest block id from each datafiles ( from x$ktfbue as we don't
need all joins from dba_extents )
  select /*+ materialize */ ktfbuesegtsn ts#,ktfbuefno
relative_fno,max(ktfbuebno+ktfbueblks-1) hwm_blocks
set verify off
column file_name format a50 word_wrapped
column smallest format 999,990 heading "Smallest|Size|Poss."
column currsize format 999,990 heading "Current|Size"
column savings format 999,990 heading "Poss.|Savings"
break on report
compute sum of savings on report

column value new_val blksize
select value from v$parameter where name = 'db_block_size'
/

select file_name,
column num_instances heading "Num" format 999
column type heading "Object Type" format a12
column source_size heading "Source" format 99,999,999
column parsed_size heading "Parsed" format 99,999,999
column code_size heading "Code" format 99,999,999
column error_size heading "Errors" format 999,999
column size_required heading "Total" format 999,999,999

select   * from (select  owner,segment_name, bytes/1024/1024 meg
from  dba_segments
   where segment_type = 'TABLE' and owner not in
('SYS','SYSTEM','DBSNMP','APEX_040200','MDSYS')
   order by   bytes/1024/1024 desc) where  rownum <= 10;

select si.owner,si.tablespace_name, i.index_name, t.table_name,
round(si.bytes/1024/1024,1) index_mb, round(st.bytes/1024/1024,1)
table_mb,
round(si.bytes/st.bytes*100,1) pct_larger
from dba_segments si, dba_indexes i, dba_tables t, dba_segments st
```

```
col owner for a30
col index_name for a30
col table_name for a30
col column_name for a30
col index_status for a20
 SELECT
segment_name,OWNER,segment_type,extents,max_extents,TABLES
PACE_NAME
 FROM dba_segments
       WHERE extents > max_extents - 10;
```

```
col table_name for a30
col total_size for a20
col actual_size for a20
select table_name,round(((blocks*8/1024)),2)||'MB' "TOTAL_SIZE",
round((num_rows*avg_row_len/1024/1024),2)||'Mb' "ACTUAL_SIZE",
round(((blocks*8/1024)-(num_rows*avg_row_len/1024/1024)),2) ||'MB'
"FRAGMENTED_SPACE",
(round(((blocks*8/1024)-
(num_rows*avg_row_len/1024/1024)),2)/round(((blocks*8/1024)),2))*1
```

```
select l.column_name, l.partition_name, l.lob_name,
l.lob_partition_name, s.bytes/1048576 "Size (MB)"
from dba_segments s, dba_lob_partitions l
where s.segment_name = l.lob_name
and s.owner='&OWNER'
and l.table_name ='&table_name'
and l.lob_partition_name = s.partition_name;
```

PDB

```
select a.data_size+b.temp_size "total_size in MB"
from ( select sum(bytes)/1024/1024 data_size
from dba_data_files) a,
( select nvl(sum(bytes),0)/1024/1024 temp_size
from dba_temp_files ) b;
```

WITH LOB Segments

```
COLUMN TABLE_NAME FORMAT A50
COLUMN OBJECT_NAME FORMAT A50
COLUMN OWNER FORMAT A30

COLUMN OBJECT_NAME FORMAT A50
COLUMN OWNER FORMAT A30

COLUMN owner FORMAT A30
COLUMN table_name FORMAT A30
COLUMN column_name FORMAT A30
```

```
SELECT  owner,tablespace_name,Sum(bytes)/1024/1024/1024 AS total_size_gb
FROM dba_segments
WHERE owner =upper('&1') group by  owner,tablespace_name;
```

```
column  pct_used format 999.9       heading "%|Used"
column  name    format a19      heading "Tablespace Name"
column  Kbytes   format 999,999,999    heading "KBytes"
column  used    format 999,999,999   heading "Used"
column  free    format 999,999,999  heading "Free"
column  largest    format 999,999,999  heading "Largest"
column  max_size format 999,999,999 heading "MaxPoss|Kbytes"
column  pct_max_used format 999.9       heading "%|Max|Used"
break   on report
compute sum of kbytes on report
```

```
alter tablespace USERTABLESPACE add datafile '+DATAGROUP' size 10G;
```

```
SELECT   A.tablespace_name tablespace, D.mb_total,
         SUM (A.used_blocks * D.block_size) / 1024 / 1024 mb_used,
         D.mb_total - SUM (A.used_blocks * D.block_size) / 1024 / 1024 mb_free
FROM     v$sort_segment A,
```

Good query runs faster than the next query below

```
Script to Command to resize
------------------------------------------
column value new_val blksize
select value from v$parameter where name = 'db_block_size'
/
column cmd format a75 word_wrapped

select 'alter database datafile '''||file_name||''' resize ' ||
ceil( (nvl(hwm,1)*&&blksize)/1024/1024 ) || 'm;' cmd
from dba_data_files a,
( select file_id, max(block_id+blocks-1) hwm
from dba_extents
group by file_id ) b
```

http://oracletechdba.blogspot.com/2014/08/oracle-to-find-over-sizedfragmented.htm

```
select table_name,avg_row_len,round(((blocks*16/1024)),2)||'MB' "TOTAL_SIZE",
    round((num_rows*avg_row_len/1024/1024),2)||'Mb' "ACTUAL_SIZE",
    round(((blocks*16/1024)-(num_rows*avg_row_len/1024/1024)),2) ||'MB'
"FRAGMENTED_SPACE",
    (round(((blocks*16/1024)-
(num_rows*avg_row_len/1024/1024)),2)/round(((blocks*16/1024)),2))*100
"percentage"
    from all_tables WHERE table_name='&TABLE_NAME';
```

CDB WITH ALL PDB

```
(
select  c.con_id,nvl(p.name, 'CDB') name,
sum(bytes)/1024/1024 SIZE_MB from  cdb_data_files c,
v$pdbs p where c.con_id=p.con_id(+) GROUP BY
c.con_id,name
UNION
select  c.con_id,nvl(p.name, 'CDB') name ,
```

```
SELECT tablespace_name,
SUM (bytes) / (1024 * 1024) "FREE(MB)"
FROM dba_free_space where
tablespace_name=UPPER('&TABLESPACE_NAME')
GROUP BY tablespace_name;
```

```
select b.Total_MB,
       b.Total_MB - round(a.used_blocks*8/1024)
Current_Free_MB,
       round(used_blocks*8/1024)
```

object that use high watermark

```
-----------------------------------------------
select
              file_name,
              segment_type,
              owner||'.'||segment_name segment,
              block_id,
```

l

```
col Table_name for a40
select table_name,
round(size_kb/1024) SIZE_MB,
round(actual_data_kb/1024) ACTUAL_DATA_MB,
round(wasted_space_kb/1024) WASTED_SPACE_MB
from (
select
   table_name,round((blocks*8),2) size_kb,
   round((num_rows*avg_row_len/1024),2)
```

Only Segment Based

```
 select sum(bytes)/1024/1024/1024 "DATABASE SIZE GB" from dba_segments;
```

```
select * from dba_tablespace_usage_metrics;
SELECT ts.tablespace_name, "File Count",
TRUNC("SIZE(MB)", 2) "Size(MB)",
TRUNC(fr."FREE(MB)", 2) "Free(MB)",
TRUNC("SIZE(MB)" - "FREE(MB)", 2) "Used(MB)",
df."MAX_EXT" "Max Ext(MB)",
(fr."FREE(MB)" / df."SIZE(MB)") * 100 "% Free"
FROM (SELECT tablespace_name,
SUM (bytes) / (1024 * 1024) "FREE(MB)"
FROM dba_free_space
GROUP BY tablespace_name) fr,
```

http://stelliosdba.blogspot.com/2012/03/ora-03297-file-contains-used-data.html

http://blog.

```
select
   table_name,round((blocks*8),2) "size (kb)" ,
   round((num_rows*avg_row_len/1024),2) "actual_data (kb)",
   (round((blocks*8),2) - round((num_rows*avg_row_len/1024),2)) "wasted_space (kb)"
from
   dba_tables
where  owner='&OWNER' and table_name='&TABLE_NAME' and
   (round((blocks*8),2) > round((num_rows*avg_row_len/1024),2))
order by 4 desc;
```

dbi-services.com/resize-your-oracle-datafiles-down-to-the-minimum-without-ora-03297/

find services belongs to pdb

pdb name for a service

find cdb name from pdb name

Close and opne PDB's

connect to PDB directly

registering the service on CRS

Relocate services

MAX_PDB_STORAGE

PDB & CDB Name

```
COLUMN name FORMAT A30
SELECT name, pdb FROM   v$services ORDER BY name;
```

```
crsctl status resource -t|grep -iw $CHANGE_ME |grep ora.*.svc|cut -f2 -d. |
while read line ; do
echo -e "\033[0;31mDatabase Name: \033[0;0m"$line
echo -e "\033[0;31mNodes: \033[0;0m"`crsctl stat res ora.$line.db -p | grep
"GEN_USR_ORA_INST_NAME@SERVERNAME" | cut -f 2 -d'@'`;
ORACLE_HOME=`crsctl stat res ora.$line.db -p | grep -iw ORACLE_HOME | cut -d'=' -f2` ; export
ORACLE_HOME
echo -e "\033[0;31mPDB: \033[0;0m"`$ORACLE_HOME/bin/srvctl config  service  -d $line -s
$CHANGE_ME | grep -i "Pluggable database name" | cut -f2 -d:`;
```

```
ps -ef | grep tns | grep SCAN | awk {'print $9'} |
while read line ; do
lsnrctl status $line | grep -i pdb_name -A 2
done
```

```
alter pluggable database <PDB_NAME> close immediate instances=ALL;
alter pluggable database <PDB_NAME> open instances=ALL;
alter pluggable database <PDB_NAME> close instances=('INST1');
```

```
sid (or) . oraenv --> to cdb then
connect to pdb :   export TWO_TASK=PDB1
```

```
collabn1:/home/oracle[RAC1]$ srvctl add service -d RAC -pdb PDB -s svctest -r RAC1 -a RAC2 -P
BASIC
collabn1:/home/oracle[RAC1]$ srvctl start service -d RAC -s svctest
srvctl status service -d <db_name> -s <service>
srvctl relocate service -d RAC12C -s <service> -oldinst <INST1> -newinst <INST2>
```

```
select PROPERTY_NAME,PROPERTY_VALUE,DESCRIPTION,CON_ID FROM cdb_properties
WHERE property_name = 'MAX_PDB_STORAGE';
```

```
select name CDB_NAME,(select name from v$pdbs) PDB_NAME,open_mode,database_role,(SELECT
to_char(sysdate,'DD-MON-YYYY HH24:MI:SS') FROM dual) "Current_time_db" ,(select
INSTANCE_NAME from v$instance) INSTANCE_NAME,(select HOST_NAME from v$instance )
HOST_NAME from v$database;
```

```
export CHANGE_ME=servicenamepls
crsctl status resource -t|grep -i $CHANGE_ME |grep ora.*.svc|cut -f2 -d. |
while read line ; do
echo -e "\033[0;31mDatabase Name: \033[0;0m"$line
echo -e "\033[0;31mNodes: \033[0;0m"`srvctl config database -d $line | grep
"Configured nodes" | cut -f2 -d:`;
echo -e "\033[0;31mPDB: \033[0;0m"`srvctl config  service  -d $line -s
$CHANGE_ME | grep -i "Pluggable database name" | cut -f2 -d:`;
done ; unset CHANGE_ME
```

```
srvctl add service -db RAC12C -service TAFSRV -preferred RAC12C1 -available
RAC12C2 \
-tafpolicy BASIC -policy AUTOMATIC -failovertype SELECT -failovermethod BASIC \
-failoverretry 5 -pdb DEMOPDB -verbose
srvctl start service -d RAC12C -s TAFSRV
```

```
col DB_UNIQUE_NAME format a10
set head on;
set lines 5000;
set tab off;
set trimspool on;
COL BYTES FORMAT 999,999,999,999,999
COL MAXBYTES FORMAT 999,999,999,999,999
COL ZBYTES  FORMAT 999,999,999,999,999
col FILE_NAME format a115
```

<---- input the service name

```
col DB_UNIQUE_NAME format a10
set lines 5000;
set tab off;
set trimspool on;
COL BYTES FORMAT 999,999,999,999,999
COL MAXBYTES FORMAT 999,999,999,999,999
COL ZBYTES FORMAT 999,999,999,999,999
col FILE_NAME format a115

col total_size format 999,999,999,999,999
col zmax_pdb_storage format
```

Rman Tuning

RMAN Examples & scenarios

Rman script to run in background

Get the latest SCN by using the below command.

Clone Progress

% completed - Good query

Rman throughput speed -good query

RMAN backup speed history

Running RMAN info (Good Query)

To see RMAN Config in sqlplus

last 7 days backup report (Best)

Short RMAN report

RMAN Backup Report

RMAN output Log File Viewing

Yesterdays Backup output / Status

RMAN Backup Available or not

Available Backup Size

Obsolete BACKUP SIZE

Day wise backup size

RMAN Currently running or not

Files Need Recovery

How long we can go back using RMAN Backup

Block Recovery

Object under corrupted block

if the v$database_block_corruption is not showing anything then (refer the link. That is good)

Skip the corrupted block

extract usefull data from corrupted table refer this link for block corruption , if RMAN block recover command is not working

RMAN catalog version

Unregister from catalog

To see the dbid from catalog

SBT - RMAN interaction

SET NEWNAME FOR DATAFILE

ALTER DATABASE RENAME FILE

Backup Failure report

Datafile report

RMAN catalog report (Completion time is older than 7 days)

Oldest backup available

Oldest backup Available in rman catalog (this gives result of all datafile oldest backup)

Oldest backup Available in rman catalog (this gives result of just time stamp)

backup throughput

restore/duplicate throughput

Backup not deleting the archivelog
ORA-19625: error identifying file
ORA-17503: ksfdopn:2 Failed to open file
ORA-15012: ASM file '+BKUP/thread_1_seq_49222.1018.892640727'
does not exist

ORA-01113: file 1 needs media recovery
ORA-01110: data file 1: '/OraData/SIAMST/siamst_system.dbf

history of my database incarnations

%FRA Used

Tracing RMAN

restoring missing archive log file

standby archivelog deletion command

Backup the backup sets from ASM disk to TAPE http://www.oraclecoursebooks.com/books/oracle10g_rman/12_10g_rman/oracle10g_rman

https://web.stanford.edu/dept/itss/docs/oracle/10gR2/backup.102/b14191/rcmdupdb006.ht

```
oracle@hostname$ cat rman_db1_clone.sh
nohup rman target sys/pwd@db1 catalog user/pwd@rcat_db auxiliary /
cmdfile='rman_db1_clone.rcv' log='rman_db1_clone.log' &
oracle@hostname$ cat rman_db1_clone.rcv
run{
allocate auxiliary channel aux1 type 'sbt_tape' parms
'ENV=(TDPO_OPTFILE=/usr/tivoli/tsm/client/oracle/bin64/tdpo.opt.host123)';
allocate auxiliary channel aux2 type 'sbt_tape' parms
'ENV=(TDPO_OPTFILE=/usr/tivoli/tsm/client/oracle/bin64/tdpo.opt.host123)';
allocate auxiliary channel aux3 type 'sbt_tape' parms
'ENV=(TDPO_OPTFILE=/usr/tivoli/tsm/client/oracle/bin64/tdpo.opt.host123)';
SET NEWNAME FOR DATAFILE  1 to '/db/db1/d1/DB2SYS.DBF';
SET NEWNAME FOR DATAFILE  2 to '/db/db1/d2/DB2UNDOT1A.DBF';
SET NEWNAME FOR DATAFILE  3 to '/db/db1/d1/DB2TLST1A.DBF';
SET NEWNAME FOR DATAFILE  4 to '/db/db1/d2/DB2USRT1A.DBF';
SET NEWNAME FOR DATAFILE  5 to '/db/db1/d13/DB2RAPT7R.DBF';
SET NEWNAME FOR DATAFILE  6 to '/db/db1/d12/DB2RAPI7O.DBF';
SET NEWNAME FOR DATAFILE  7 to '/db/db1/d14/DB2RAPT7I.DBF';
duplicate target database to db2;
}
```

```
select max(first_change#) chng
 from v$archived_log
 /
```

```
COLUMN sid FORMAT 999
COLUMN serial# FORMAT 9999999
COLUMN machine FORMAT A30
COLUMN progress_pct FORMAT 99999999.00
COLUMN elapsed FORMAT A10
COLUMN remaining FORMAT A10
SELECT sl.opname,s.sid,
       s.serial#,
       s.machine,
       TRUNC(sl.elapsed_seconds/60) || ':' || MOD(sl.elapsed_seconds,60) elapsed,
       TRUNC(sl.time_remaining/60) || ':' || MOD(sl.time_remaining,60) remaining,
       ROUND(sl.sofar/sl.totalwork*100, 2) progress_pct
FROM   v$session s,
       v$session_longops sl
WHERE  s.sid     = sl.sid
AND    s.serial# = sl.serial#;
```

```
SELECT SID, SERIAL#, CONTEXT, SOFAR, TOTALWORK,opname ,
ROUND(SOFAR/TOTALWORK*100,2) "%_COMPLETE", Time_remaining
FROM V$SESSION_LONGOPS
WHERE OPNAME LIKE 'RMAN%'
AND OPNAME NOT LIKE '%aggregate%'
AND TOTALWORK != 0
AND SOFAR != TOTALWORK;
```

```
set linesize 126
column Pct_Complete format 99.99
column client_info format a25
column sid format 999
column MB_PER_S format 999.99
select s.client_info,
l.sid,
l.serial#,
l.sofar,
l.totalwork,
```

```
select to_char(START_TIME,'MM-DD-YYYY HH24:MI:SS') as
START_TIME,to_char(END_TIME,'MM-DD-YYYY HH24:MI:SS') as
END_TIME,ELAPSED_SECONDS/60/60,INPUT_BYTES/1024/1024/1024/1024 as
INPUT_TB,OUTPUT_BYTES/1024/1024/1024/1024 as
OUTPUT_TB,INPUT_BYTES_PER_SEC/1024/1024 as
```

```
SELECT p.SPID, s.sid, s.serial#, sw.EVENT, sw.SECONDS_IN_WAIT AS
SEC_WAIT, sw.STATE, CLIENT_INFO
FROM V$SESSION_WAIT sw, V$SESSION s, V$PROCESS p
WHERE s.client_info LIKE 'rman%'
AND s.SID=sw.SID
AND s.PADDR=p.ADDR;
```

```
select * from v$rman_configuration;
```

```
COL STATUS FORMAT a25
COL HRS    FORMAT 999.99
COL start_time  format a15
COL end_time  format a15
COL in_size  FORMAT a10

SELECT INPUT_TYPE,
       STATUS,
       TO_CHAR(START_TIME,'mm/dd/yy hh24:mi') start_time,
       TO_CHAR(END_TIME,'mm/dd/yy hh24:mi')   end_time,
       INPUT_BYTES_DISPLAY in_size,
       round(ELAPSED_SECONDS/3600,3) HRS
FROM V$RMAN_BACKUP_JOB_DETAILS
where sysdate - start_time <= 7;
```

```
col start_time         heading 'Started'      format a20
col backup_type        heading 'Backup Type'  format a12
col time_taken_display heading 'Elapsed|Time' format a10
col elapsed_min        heading 'Run|Min'      format 999
```

```
col start_time  for a30
col TIME_TAKEN_DISPLAY for a10
select
  j.session_recid, j.session_stamp,
  to_char(j.start_time, 'yyyy-mm-dd hh24:mi:ss') start_time,
  to_char(j.end_time, 'yyyy-mm-dd hh24:mi:ss') end_time,
  (j.output_bytes/1024/1024) output_mbytes, j.status, j.input_type,
  decode(to_char(j.start_time, 'd'), 1, 'Sunday', 2, 'Monday',
                                     3, 'Tuesday', 4, 'Wednesday',
                                     5, 'Thursday', 6, 'Friday',
                                     7, 'Saturday') dow,
  j.elapsed_seconds, j.time_taken_display,
```

```
select output
from GV$RMAN_OUTPUT
where session_recid = &SESSION_RECID
  and session_stamp = &SESSION_STAMP
order by recid;
```

```
select output
from v$rman_output
where session_recid = (select max(session_recid) from v$rman_status)
order by recid ;
```

```
SELECT
     bs.recid                      bs_key
    , bp.tag                                                                       tag
  , DECODE(backup_type
          , 'L', 'Archived Redo Logs'
          , 'D', 'Datafile Full Backup'
          , 'I', 'Incremental Backup'  )                 backup_type
  , bs.incremental_level
incremental_level
  , DECODE(   bp.status
```

```
 select  round(sum(bytes/1024/1024/1024),2) bsize  from  v$backup_piece  where
STATUS='A';
SELECT SUM(BYTES/1024/1024/1024) FROM V$BACKUP_FILES where
OBSOLETE='YES' and FNAME is not null and ((FILE_TYPE='PIECE' and
BACKUP_TYPE='BACKUP SET') or BACKUP_TYPE='COPY')
```

```
         , decode(backup_type, 'L', 'Archive Log', 'D', 'Full', 'Incremental') backup_type
         , bsize "Size MB"
    from (select trunc(bp.completion_time) ctime
           , backup_type
           , round(sum(bp.bytes/1024/1024),2) bsize
       from v$backup_set bs, v$backup_piece bp
       where bs.set_stamp = bp.set_stamp
      and bs.set_count  = bp.set_count
COLUMN SID FORMAT 999
COLUMN SPID FORMAT 9999

SELECT s.SID, p.SPID, s.CLIENT_INFO

SELECT * FROM V$RECOVER_FILE;

 select TO_CHAR(min(bs.completion_time),'DD/MM/YYYY HH24:MI') completion_time
from v$backup_set bs, v$backup_piece bp  where
 bs.set_stamp = bp.set_stamp
 AND bs.set_count = bp.set_count and bs.backup_type='D'
 AND bp.status='A';

blockrecover datafile 8 block 13;

SELECT owner, segment_name, segment_type
FROM dba_extents
WHERE file_id = 1
AND 16516 BETWEEN block_id AND block_id + blocks - 1;


RMAN> validate tablespace corrupt;

DBMS_REPAIR.SKIP_CORRUPT_BLOCKS (
     SCHEMA_NAME => 'SCOTT',
     OBJECT_NAME => 'DEPT',
     OBJECT_TYPE => dbms_repair.table_object,
     FLAGS => dbms_repair.skip_flag);
END;
/
```

Querying scott's tables using the DBA_TABLES view shows that SKIP_CORRUPT is enabled for table scott.dept.

```
SELECT OWNER, TABLE_NAME, SKIP_CORRUPT FROM DBA_TABLES


);
PL/SQL procedure successfully completed.
SQL> insert into MANA.TAXINQUIRY_LOG_REPAIR ( select * from
MANA.TAXINQUIRY_LOG );
```

Note: if any error while creating table (SCRIPT to skip ORA-8103 ORA-1410 ORA-1578 ORA-600 [kdsgrp1] when reading a TABLE (Doc ID 1527738.1)) this metalink note will help

https://docs.oracle.com/cd/B10501_01/server.920/a96521/repair.htm

```
select * from schemaname.rcver;
```

```
RUN
{
  SET DBID 3668200963;
  UNREGISTER DATABASE DB_NAME NOPROMPT;
}
```

```
select DBID, NAME ,STATUS from RC_DATABASE_INCARNATION where name like
'%IKB%'
```

```
COLUMN EVENT FORMAT a10
COLUMN SECONDS_IN_WAIT FORMAT 999
COLUMN STATE FORMAT a20
COLUMN CLIENT_INFO FORMAT a30

SELECT p.SPID, EVENT, SECONDS_IN_WAIT AS SEC_WAIT,
       sw.STATE, CLIENT_INFO
FROM V$SESSION_WAIT sw, V$SESSION s, V$PROCESS p
WHERE sw.EVENT LIKE 's%bt%'
       AND s.SID=sw.SID
       AND s.PADDR=p.ADDR
;
```

```
select 'SET NEWNAME FOR DATAFILE ' || FILE# || ' TO ''' || '/u03/oradata/&1/' ||
substr(name,instr(name,'/',-
1)+1) || ''';' from v$datafile;
```

```
SELECT 'SQL "ALTER DATABASE RENAME FILE '''''|| MEMBER ||''''''' ||chr(10)||'to ''''' ||
member || ''''''' ;' FROM
V$LOGFILE;
```

```
select 'V92 - ' ||DB NAME,dbid,min(backuptype_db) DBBKP,min(backuptype_arch)
ARCBKP
from (
select a.name DB,dbid,
decode(b.bck_type,'D',max(b.completion_time),'I',
max(b.completion_time)) BACKUPTYPE_db,
decode(b.bck_type,'L',
max(b.completion_time)) BACKUPTYPE_arch
from v92.rc_database a,v92.bs b
where a.db_key=b.db_key
and b.bck_type is not null having max(start_time) < sysdate - 7
group by a.name,dbid,b.bck_type
) group by db,dbid
union
select 'V102 - ' ||DB NAME,dbid,min(backuptype_db) DBBKP,min(backuptype_arch)
ARCBKP
from (
select a.name DB,dbid,
decode(b.bck_type,'D',max(b.completion_time),'I',
max(b.completion_time)) BACKUPTYPE_db,
decode(b.bck_type,'L',
max(b.completion_time)) BACKUPTYPE_arch
from v102.rc_database a,v102.bs b
where a.db_key=b.db_key and a.name not in ('P001SLT','P1WEBAPP')
and b.bck_type is not null having max(start_time) < sysdate - 7
group by a.name,dbid,b.bck_type
) group by db,dbid
union
select 'V105 - ' ||DB NAME,dbid,min(backuptype_db) DBBKP,min(backuptype_arch)
ARCBKP
from (
select a.name DB,dbid,
decode(b.bck_type,'D',max(b.completion_time),'I',
set feedback     off
set verify       off
set serveroutput on size 1000000

column db    new_value _DB noprint;
column dbkey new_value _DBKEY noprint;

select 'rman'||'&&1' db from dual;
select dbinc_key dbkey from &_DB..rc_database;


/* cleanup work if you see duplicate entries due to multiple incarnations */
--delete from &_DB..dbinc where dbinc_key != (select dbinc_key from
&_DB..rc_database);
--delete from &_DB..dfatt where dbinc_key != (select dbinc_key from
&_DB..rc_database);
--delete from &_DB..bdf   where dbinc_key != (select dbinc_key from
&_DB..rc_database);
--delete from &_DB..df    where dbinc_key != (select dbinc_key from
&_DB..rc_database);
--delete from &_DB..al    where dbinc_key != (select dbinc_key from
&_DB..rc_database);
--delete from &_DB..tsatt where dbinc_key != (select dbinc_key from
&_DB..rc_database);
--delete from &_DB..ts    where dbinc_key != (select dbinc_key from
&_DB..rc_database);


-- was forced to create a new table due to the index on the
-- original table cause invalid packages when dropping

create table &_DB.BDF nologging as select file#, COMPLETION_TIME, create_scn from
&_DB..BDF where DBINC_KEY = &_DBKEY;


select 'V92 - ' ||DB NAME,dbid,min(backuptype_db) DBBKP,min(backuptype_arch)
ARCBKP
from (
select a.name DB,dbid,
decode(b.bck_type,'D',max(b.completion_time),'I',
max(b.completion_time)) BACKUPTYPE_db,
decode(b.bck_type,'L',
max(b.completion_time)) BACKUPTYPE_arch
from v92.rc_database a,v92.bs b
where a.db_key=b.db_key
and b.bck_type is not null having max(start_time) < sysdate - 7
group by a.name,dbid,b.bck_type
) group by db,dbid
```

V92

v102

v104

v112

v121

```
                bdf.completion_time,
                df.file#,
                fname,
                tag
    FROM v112.dbinc dbi,
        v112.bdf,
        v112.ts,
        v112.df,
```

```
select DB NAME,dbid,NVL(TO_CHAR(min(backuptype_db),'DD/MM/YYYY
HH24:MI'),'01/01/0001:00:00') DBBKP,
NVL(TO_CHAR(min(backuptype_arch),'DD/MM/YYYY HH24:MI'),'01/01/0001:00:00')
ARCBKP
from (
select a.name DB,dbid,
decode(b.bck_type,'D',min(b.completion_time),'I', min(b.completion_time))
BACKUPTYPE_db,
decode(b.bck_type,'L',min(b.completion_time)) BACKUPTYPE_arch
```

```
SELECT
  'BACKUP THROUGHPUT',
  round(SUM(v.value/1024/1024),1) mbytes_sofar,
  round(SUM(v.value     /1024/1024)/nvl((SELECT MIN(elapsed_seconds)
  FROM v\$session_longops
  WHERE OPNAME LIKE 'RMAN: aggregate output'
  AND SOFAR            != TOTALWORK
  AND elapsed_seconds IS NOT NULL
  ),SUM(v.value     /1024/1024)),2) mbytes_per_sec,
  n.name
FROM gv\$sesstat v,
  v\$statname n,
  gv\$session s
WHERE v.statistic#=n.statistic#
AND n.name = 'physical read total bytes'
AND v.sid = s.sid
AND v.inst_id=s.inst_id
AND s.program like 'rman@%'
GROUP BY 'BACKUP THROUGHPUT',n.name;
```

```
SELECT
  'DUPLICATE/RESTORE THROUGHPUT',
  round(SUM(v.value/1024/1024),1) mbytes_sofar,
  round(SUM(v.value     /1024/1024)/nvl((SELECT MIN(elapsed_seconds)
  FROM v\$session_longops
  WHERE OPNAME LIKE 'RMAN: aggregate input'
  AND SOFAR            != TOTALWORK
  AND elapsed_seconds IS NOT NULL
  ),SUM(v.value     /1024/1024)),2) mbytes_per_sec,
  n.name
FROM gv\$sesstat v,
  v\$statname n,
  gv\$session s
WHERE v.statistic#=n.statistic#
AND n.name = 'physical write total bytes'
AND v.sid = s.sid
AND v.inst_id=s.inst_id
AND s.program like 'rman@%'
GROUP BY 'DUPLICATE/RESTORE THROUGHPUT',n.name;
```

```
run {
allocate channel c1 type 'SBT'
parms'ENV=(TDPO_OPTFILE=/opt/tivoli/tsm/client/oracle/bin64/tdpo.opt)';
allocate channel c2 type 'SBT'
parms'ENV=(TDPO_OPTFILE=/opt/tivoli/tsm/client/oracle/bin64/tdpo.opt)';
backup archivelog all skip inaccessible delete input;
release channel c1;
release channel c2;
}
```

```
F.GROUP#
  2  AND G.STATUS = 'CURRENT';

MEMBER
--------------------------------------------------------------------------------
/OraRedo/RedoLogFiles/siamst_log01.dbf
```

```
SELECT lpad(' ',2*(level-1))
  || TO_CHAR(DBINC_KEY) AS DBINC_KEY,
  db_key,
  db_name,
  TO_CHAR(reset_time,'YYYY-MM-DD HH24:MI:SS'),
  dbinc_status
FROM rman.dbinc
  START WITH PARENT_DBINC_KEY IS NULL
  CONNECT BY prior DBINC_KEY   = PARENT_DBINC_KEY ;
```

```
SELECT
  ROUND((A.SPACE_LIMIT / 1024 / 1024 / 1024), 2) AS FLASH_IN_GB,
  ROUND((A.SPACE_USED / 1024 / 1024 / 1024), 2) AS FLASH_USED_IN_GB,
  ROUND((A.SPACE_RECLAIMABLE / 1024 / 1024 / 1024), 2) AS
FLASH_RECLAIMABLE_GB,
  SUM(B.PERCENT_SPACE_USED)  AS PERCENT_OF_SPACE_USED
FROM
  V$RECOVERY_FILE_DEST A,
  V$FLASH_RECOVERY_AREA_USAGE B
GROUP BY
  SPACE_LIMIT,
  SPACE_USED ,
  SPACE_RECLAIMABLE ;
```

```
rman rcvcat rman/rman@catalog target user/passwd@target debug trace trace_file_name
```

```
run
{
set archivelog destination to '/ora_backup/rman/arch/';
restore archivelog from logseq=8619 until logseq=8632 thread=2;
}
```

```
run{
CONFIGURE ARCHIVELOG DELETION POLICY TO NONE;
crosscheck archivelog all;
crosscheck backup;
delete noprompt expired backup;
delete noprompt expired archivelog all;
delete noprompt obsolete;
delete noprompt archivelog all completed before 'sysdate - 1/24';
}
```

```
RMAN> crosscheck backup;
RMAN> BACKUP DEVICE TYPE sbt BACKUPSET ALL DELETE INPUT;
```

```
select sl.sid, sl.opname,
       to_char(100*(sofar/totalwork), '990.9')||'%'
pct_done,
       sysdate+(TIME_REMAINING/60/60/24)
done_by
  from v$session_longops sl, v$session s
 where sl.sid = s.sid
   and sl.serial# = s.serial#
   and sl.sid in (select sid from v$session where
module like 'backup%' or module like 'restore%' or
```

```
set lines 750 pages 9999
col start_time         heading 'Started'      format a30
col end_time           heading 'End'          format a30
col time_taken_display heading 'Elapsed|Time'
format a10
```

<-- Very good query

```
      r.command_id
backup_name
  ,              TO_CHAR(r.start_time, 'Dy mm/dd/yyyy
HH24:MI:SS')              start_time
  ,              TO_CHAR(r.end_time, 'Dy mm/dd/yyyy
HH24:MI:SS')              end_time
  ,             r.time_taken_display
elapsed_time
  , DECODE(   r.status
           , 'COMPLETED'
           ,   r.status
           , 'RUNNING'
```

First execute the above query & get RECID & STAMP ID

```
select object_type,mbytes_processed, start_time,
end_time,status
from v$rman_status
where session_recid = (select max(session_recid)
from v$rman_status)
and operation !='RMAN'
order by recid;
```

may work

Master Note for Handling Oracle Database Corruptio

http://www.pythian.com/blog/oracle-free-block-corrup

https://gruffdba.wordpress.com/2012/11/28/oracle-da https://alexzeng.wordpress.com/2008/09/17/how-to-r

http://www.adp-gmbh.ch/ora/admin/scripts/rman_backup_history.html

Just test once before running

```
SELECT DISTINCT dbi.db_name,
                   bdf.completion_time,
                   df.file#,
                   fname,
                   bp.tag
SELECT DISTINCT dbi.db_name,
                   bdf.completion_time,
                   df.file#,
                   bdf.completion_time,
                   df.file#,
                   fname,
                   bdf.completion_time,
                   df.file#,
SELECT DISTINCT dbi.db_name,
                   bdf.completion_time,
                   df.file#,
                   fname,
                   tag
     FROM v121.dbinc dbi,
          v121.bdf,
```

Customized from the Above query

This is a GOOD Query for the TIMESTAMP

https://blogs.oracle.com/sysdba/entry/rman_backup_ http://technopark02.blogspot.com/2007/09/oracle-fixi

```
select * from v$flash_recovery_area_usage;
```

e

```
col dbsize_mbytes      for 99,999,990.00 justify right head "DBSIZE_MB"
col input_mbytes       for 99,999,990.00 justify right head "READ_MB"
col output_mbytes      for 99,999,990.00 justify right head "WRITTEN_MB"
col output_device_type for a10           justify left head "DEVICE"
col complete           for 990.00        justify right head "COMPLETE %"
col compression        for 990.00        justify right head "COMPRESS|% ORIG"
col est_complete       for a20           head "ESTIMATED COMPLETION"
col recid              for 9999999       head "ID"

select recid
```

CF: Number of controlfile backups included in the backup set
DF: Number of datafile full backups included in the backup set
I0: Number of datafile incremental level-0 backups included in the backup set
I1: Number of datafile incremental level-1 backups included in the backup set
L: Number of archived log backups included in the backup set
OUT INST: Instance where the job was executed and the output is available

http://www.pythian.com/blog/viewing-rma-jobs-status-and-output/

n Issues (Doc ID 1088018.1)

tion-test-case/

tabase-corruption-identify-and-correct/

```
CREATE OR REPLACE FORCE VIEW VSEIRCAT_SOX_RPT
(
   DBINC_KEY,
   DB_NAME,
   COMPLETION_TIME,
```

we have to do a full backup, Any backup prior to missing archives is no longer valid.... 1:51 PM

before that we have to .... 1:51 PM

backup archivelog all skip inaccessible delete input; 1:52 PM

and 1:52 PM

change archivelog all validate.... 1:52 PM

the first will allow the backup of all the archives and skip the missing 1:52 PM

the second will allow the backups to continue normally going forward by marking the archive backups as validated even though the are missing. 1:53 PM

The fact that we had hard errors on the disk that were not correctable is odd.

```
      r.command_id
backup_name
  ,
TO_CHAR(r.start_time,
'mm/dd/yyyy HH24:MI:SS')
start_time
  ,
TO_CHAR(r.end_time,
'mm/dd/yyyy HH24:MI:SS')
end_time
  ,
r.time_taken_display
```

restore point check
drop restorepoint

flashback on /off
FRA Usage

Is row movement enabled for a table?

finding Dependent constraint.(or) ORA-02266: unique/primary keys in table referenced by enabled foreign keys

FRA size

TIME stamp to SCN

SCN to Time stamp

Findng Current SCN

v$flash_recovery_area_usage incorrect archivelog count

```
select name, scn, time, guarantee_flashback_database,DATABASE_INCARNATION# from v$restore_point;
drop   restore point  PSUCHECK;

select flashback_on from v$database;
 alter database flashback off;
  alter database flashback on;
  select * from v$flash_recovery_area_usage;


select
   table_name,row_movement from dba_tables
where
     owner='ODB'
and
   table_name = 'AC_ACTUAL_FLIGHTS';
SELECT CONSTRAINT_NAME, OWNER,TABLE_NAME FROM DBA_CONSTRAINTS
WHERE R_CONSTRAINT_NAME IN
( SELECT CONSTRAINT_NAME
FROM dba_constraints
WHERE
OWNER=UPPER('&user_dependent_from') AND
TABLE_NAME=UPPER('&object_dependent_from')  )
AND STATUS='ENABLED';
set lines 100
col name format a60
select name
, floor(space_limit / 1024 / 1024) "Size MB"
, ceil(space_used  / 1024 / 1024) "Used MB"
from v$recovery_file_dest
order by name
/

select timestamp_to_scn(to_timestamp('25/07/2012 16:32:30','DD/MM/YYYY HH24:MI:SS')) as scn from dual;

select scn_to_timestamp(176195435) as timestamp from dual;

select current_scn from v$database ;
```

```
RUN {
change archivelog all crosscheck;
report obsolete orphan;
report obsolete;
crosscheck backup;
crosscheck copy;
crosscheck backup of controlfile;

delete noprompt expired backup;
delete noprompt expired archivelog all;
delete noprompt expired backup of controlfile;
delete force noprompt expired copy;
delete force noprompt obsolete orphan;
delete force noprompt obsolete;
}
```

```
----------

Enter value for user_dependent_from: FRODO
Enter value for object_dependent_from: CITY

CONSTRAINT_NAME     OWNER    TABLE_NAME
  ZZZ              FRODO      COUNTRY
```

EXP , How long will it take ?

copy one table from one db to another

Estimate Dumpfile Size

ORA-02266: unique/primary keys in table referenced by enabl
ORA-27054: NFS file system where the file is created or resides is not mounted with correct options

Export the table from previous time

user privs for impdp

what expdp/impdp actualy executing

schema backup

Scan the table using exp without taking dumpfile

DBA _resumable

Querying **V$SESSION_LONGOPS & V$SESSION** views

Querying V$SESSION_LONGOPS & V$DATAPUMP_JOB

Querying all the related views with a single query:-

```
set lines 750 pages 9999
 col  job_name for a30
 col STATE for a10
 col sql_text for a100
 col message for a100
 col job_mode for a30

 SELECT x.job_name,
      b.state,
```

```
set arraysize 1000
copy from username/passwd@source_connect_string to username/passwd@target_connect_string insert
table_target using select * from table_source;
```

Suppose if the table exists then

```
COPY FROM username1/passwd1@PROD to username2/passwd2@SANDBOX
    INSERT TABLE_C (*) USING (SELECT * FROM TABLE_C WHERE COL_A = 4884);
```

```
 expdp system/******* SCHEMAS=CMX_ORS NOLOGFILE=y ESTIMATE_ONLY=y DIRECTORY=tmp_pump_dir
```

```
alter session set current_schema=CMFP;
select 'alter table '||a.owner||'.'||a.table_name||' disable constraint '||a.constraint_name||';'
    from all_constraints a, all_constraints b
    where a.constraint_type = 'R'
    and a.r_constraint_name = b.constraint_name
    and a.r_owner  = b.owner
    and b.table_name = 'CMFP_HRCHY';
```

```
ALTER SYSTEM SET EVENTS '10298 trace name context forever, level 32';
```

```
exp system/***** tables=schema.table_name file=exp_ix_str_store.dmp log=exp_ix_str_store.log query=\"where DI
```

```
set long 20000 longchunksize 20000 pagesize 0 linesize 1000 feedback off verify off trimspool on
column ddl format a1000

begin
   dbms_metadata.set_transform_param (dbms_metadata.session_transform, 'SQLTERMINATOR', true);
   dbms_metadata.set_transform_param (dbms_metadata.session_transform, 'PRETTY', true);
end;
/

variable v_username VARCHAR2(30);
```

```
SELECT DISTINCT dp.job_name, dp.session_type, s.inst_id, s.SID, s.serial#,
 s.username, s.inst_id, s.event, s.sql_id, q.sql_text,
 dj.operation, dj.state
 FROM gv$session s,
 dba_datapump_sessions dp,
 dba_datapump_jobs dj,
 gv$sql q
 WHERE s.saddr = dp.saddr
 AND dp.job_name = dj.job_name
 AND s.sql_id = q.sql_id
 AND s.inst_id IN (1, 2, 3)
ORDER BY s.inst_id;
```

expdp

```
nohup expdp parfile=/dbexports/EXP/dbname/schemaexpdp.par &
```

```
userid='/ as sysdba'
dumpfile=WRNAME_exp_dbname_SCOTT.dmp
logfile=WRNAME_exp_w665pr_SCOTT.log
```

```
exp tables=<XXX> file=/dev/null log=/tmp/export.log statistics=none volsize=0
```

```
select * from dba_resumable;
```

```
SELECT b.username, a.sid, b.opname, b.target,
          round(b.SOFAR*100/b.TOTALWORK,0) || '%' as "%DONE", b.TIME_REMAINING,
          to_char(b.start_time,'YYYY/MM/DD HH24:MI:SS') start_time
    FROM v$session_longops b, v$session a
    WHERE a.sid = b.sid      ORDER BY 6;
```

```
SELECT sl.sid, sl.serial#, sl.sofar, sl.totalwork, dp.owner_name, dp.state, dp.job_mode
    FROM v$session_longops sl, v$datapump_job dp
    WHERE sl.opname = dp.job_name
    AND sl.sofar != sl.totalwork;
```

```
select x.job_name,b.state,b.job_mode,b.degree
, x.owner_name,z.sql_text, p.message
, p.totalwork, p.sofar
, round((p.sofar/p.totalwork)*100,2) done
, p.time_remaining
from dba_datapump_jobs b
left join dba_datapump_sessions x on (x.job_name = b.job_name)
left join v$session y on (y.saddr = x.saddr)
left join v$sql z on (y.sql_id = z.sql_id)
left join v$session_longops p ON (p.sql_id = y.sql_id)
WHERE y.module='Data Pump Worker'
AND p.time_remaining > 0;
```

```
col username for a20
col opname for a50
col  message for a100
set lines 750 pages 9999
select username,opname,target_desc,sofar,totalwork,message from
V$SESSION_LONGOPS where message not like '%RMAN%' and
username='SYS';
```

```
COPY FROM username1/passwd1@//192.168.3.17:1521/PROD_SERVICE to
username2/passwd2@//192.168.4.17:1521/SANDBOX_SERVICE
    INSERT TABLE_C (*) USING (SELECT * FROM TABLE_C WHERE COL_A
= 4884);
```

```
alter system set event='10298 trace name context forever, level 32' scope=spfile;
```

```
BTIME\<to_date\(\'2014-05-27 00:01:08\'\,\'yyyy-mm-dd hh24:mi:ss\'\)\"
```

Schema imp dp important

SELECT b.username, a.sid, b.opname, b.target, round(b.SOFAR*100/b.TOTALWORK,0) || '%' as "%DONE", b.TIME_REMAINING, to_char(b.start_time,'YYYY/MM/DD HH24:MI:SS') start_time FROM v$session_longops b, ;	SYS_EXPORT_SCHEMA_01 will show the actual time required Rowid Range	http://www.dbaref.com/home/oracle-11g-new-features/r

monitoringdatapump

Reverse Engineering

ddl statement of all types

Last DDL Date

DDL

package ddl

Generating DDL of table

Generate DDL of DB_LINK

Sequence DDL

Tablespace DDL

Users DDL

DDL of all users (Except schema)

roles granted to user
system privs granted to user
other privs granted to user

A single Objects Grants (any type of Object)

Password DDL

Cannot reuse the password

ORA-28003: password verification for the specified password failed
ORA-20009: Error: You cannot change password.

Profile

Roles DDL

create a role and assign all privileges to the role

Roles comparision between databases

No of users have a particular role

ALL Privileges for a SINGLE user

List of Users/Roles having privilege on TABLE

Privileges granted by you to others

System privileges to roles and users

Roles & privs for a user

when was the password was changed for a user

NON Default database users

DDL of V$ Tables

Parallel DDL option

Source code of all invalid objects

create scripts of all tables of a particular schema

GET_DEPENDENT_DDL

http://www.orafaq.com/node/807

http://amit7oracledba.blogspot.com/2013/02/dbmsmetadatagetddl-package-how-to-get.html

```
select to_char(last_ddl_time,'DD-MON-YYYY HH24:MI:SS') from dba_objects where owner='&user' and object_name='&table_name';
set long 20000 longchunksize 20000 pagesize 0 linesize 1000
SELECT DBMS_METADATA.GET_DDL('&OBJECT_TYPE','&OBJECT_NAME','&OWNER') AAA FROM DUAL;
set pagesize 0 echo off timing off linesize 1000 trimspool on trim on long 2000000 longchunksize 2000000
SELECT DBMS_METADATA.GET_DDL('PACKAGE_SPEC','&OBJECT_NAME','&OWNER') AAA FROM DUAL;
```

```
set long 1000
select dbms_metadata.get_ddl('TABLE','EMP','SCHEMA')||'/' from dual;
```

```
SET LONG 1000
SELECT DBMS_METADATA.GET_DDL('DB_LINK',db.db_link,db.owner) from dba_db_links db;
' INCREMENT BY '||INCREMENT_BY||chr(10)||
' START WITH '||LAST_NUMBER||chr(10)||
' MINVALUE '||MIN_VALUE||chr(10)||
' MAXVALUE '||MAX_VALUE||chr(10)||
decode(CYCLE_FLAG,'N',' NOCYCLE','CICLE')||chr(10)||
decode(ORDER_FLAG,'N',' NOORDER','ORDER')||chr(10)||
```

```
select 'select dbms_metadata.get_ddl(''TABLESPACE'','''|| tablespace_name || ''') from dual;' from dba_tablespaces;
```

```
select dbms_metadata.get_ddl('TABLESPACE','&TABLESPACE_NAME') from dual;
```

```
set long 2000
select (case
        when ((select count(*)
               from   dba_users
               where  username = 'ODB') > 0)
        then  dbms_metadata.get_ddl ('USER', 'ODB')
        else  to_clob ('   -- Note: User not found!')
        end ) Extracted_DDL from dual
UNION ALL
select (case
        when ((select count(*)
```

```
  set long 20000 longchunksize 20000 pagesize 0 linesize 1000
  select dbms_metadata.get_ddl('USER',username)  FROM dba_users where username  not in ( select owner from dba_objects) and username like 'U%' OR username like 'A%';
```

```
select dbms_metadata.get_ddl('USER', '&USER') || '/' usercreate from dual;
SELECT DBMS_METADATA.GET_GRANTED_DDL('ROLE_GRANT','&USER') || '/' FROM DUAL;
SELECT DBMS_METADATA.GET_GRANTED_DDL('SYSTEM_GRANT','&USER') || '/' FROM DUAL;
SELECT DBMS_METADATA.GET_GRANTED_DDL('OBJECT_GRANT','&USER') || '/' FROM DUAL;
select DBMS_METADATA.GET_GRANTED_DDL('TABLESPACE_QUOTA', '&USER') '/' from dual;
  select 'grant ' || PRIVILEGE || ' on ' || OWNER || '.'  ||TABLE_NAME ||  ' to ' || GRANTEE || ';' from dba_tab_privs
where table_name='&Any_type_of_object_name' ;
select 'alter user ' || NAME ||' identified by values ''' || password || ''';' from user$;
declare
userNm varchar2(100);
userpswd varchar2(100);
begin
  userNm := upper('&TypeUserNameHere');
  select password into userpswd from sys.user$ where name = userNm;
  execute immediate ('ALTER PROFILE "FUNCTIONAL_USER" LIMIT
  PASSWORD_VERIFY_FUNCTION null
  PASSWORD_LIFE_TIME UNLIMITED
  PASSWORD_REUSE_TIME UNLIMITED
  PASSWORD_REUSE_MAX UNLIMITED');
  execute immediate ('alter user '||userNm||' identified by oct152014oct');
  execute immediate ('alter user '||userNm||' identified by values '''||userpswd||'''');
  execute immediate ('ALTER PROFILE "FUNCTIONAL_USER" LIMIT
  PASSWORD_VERIFY_FUNCTION PASSWDCOMPLEXVERIFICATION');

end;
/
SQL> ALTER PROFILE FUNCTIONAL_USER LIMIT PASSWORD_VERIFY_FUNCTION NULL;
Profile altered.
SQL> alter user trial identified by test;
User altered.
SQL> conn trial/test;
ALTER PROFILE "FUNCTIONAL_USER" LIMIT   PASSWORD_VERIFY_FUNCTION
```

```
select ' alter user '||username||' profile '||PROFILE||';' from dba_users;
set long 20000 longchunksize 20000 pagesize 0 linesize 1000 feedback off verify off trimspool on
column ddl format a1000

begin
   dbms_metadata.set_transform_param (dbms_metadata.session_transform, 'SQLTERMINATOR', true);
   dbms_metadata.set_transform_param (dbms_metadata.session_transform, 'PRETTY', true);
end;
/

variable v_role VARCHAR2(30);
```

```
create role L1_SUPPORT;
create role L2_SUPPORT;
create role L3_SUPPORT;

set pagesize 0
set echo off
set trimspool on
set linesize 120
set feedback off

spool grant.sql
DB link : dev to prod

CREATE DATABASE LINK "COMPARE" CONNECT TO DBSNMP IDENTIFIED BY mypwd USING 'destonation-db-name';
 select *  from DBA_TAB_PRIVS@compare T1 where not exists (select 1 from dba_tab_privs T2 where
```

```
select GRANTEE as users from dba_role_privs where GRANTED_ROLE=UPPER('&GRANTED_ROLE');
```

```
select 'GRANT '||granted_role||' to '||grantee||
       DECODE(ADMIN_OPTION, 'Y', ' WITH ADMIN OPTION;', ';')
  from dba_role_privs
 where grantee like upper('%&&uname%')
         UNION
select 'GRANT '||privilege||' to '||grantee||
       DECODE(ADMIN_OPTION, 'Y', ' WITH ADMIN OPTION;', ';')
  from dba_sys_privs
 where grantee like upper('%&&uname%')
         UNION
select 'grant ' || PRIVILEGE || ' on ' || OWNER || '.'  ||TABLE_NAME ||  ' to ' || GRANTEE || ';'
  from dba_tab_privs
 where grantee like upper('%&&uname%');
SELECT     grantee || ' Through role ' || granted_role ge, 'SELECT' priv
      FROM dba_role_privs
START WITH granted_role IN (SELECT grantee
                              FROM dba_tab_privs
                             WHERE PRIVILEGE = 'SELECT')
CONNECT BY PRIOR grantee = granted_role
UNION
SELECT     grantee || ' Through role ' || granted_role ge, 'UPDATE' priv
      FROM dba_role_privs
```

LISTING INFORMATION ABOUT PRIVILEGES

To see which table privileges are granted by you to other users.

```
SELECT * FROM USER_TAB_PRIVS_MADE
```

To see which table privileges are granted to you by other users

```
SELECT * FROM USER_TAB_PRIVS_RECD;
```

To see which column level privileges are granted by you to other users.

```
SELECT * FROM USER_COL_PRIVS_MADE
```

To see which column level privileges are granted to you by other users

```
SELECT * FROM USER_COL_PRIVS_RECD;
```

To see which privileges are granted to roles

```
SELECT * FROM USER_ROLE_PRIVS;
```

```
  lpad(' ', 2*level) || c "Privilege, Roles and Users"
from
  (
  /* THE PRIVILEGES */
    select
      null   p,
      name   c
    from
      system_privilege_map
    where
```

Granted Roles:

```
SELECT *   FROM DBA_ROLE_PRIVS
 WHERE GRANTEE = 'USER';
```

Privileges Granted Directly To User:

```
SELECT *
  FROM DBA_TAB_PRIVS
 WHERE GRANTEE = 'USER';
```

```
SELECT NAME, TO_CHAR (ptime, 'DD-MON-YYYY HH24:MI:SS') AS "LAST TIME CHANGED", ctime "CREATION TIME", ltime "LOCKED"
FROM USER$
WHERE ptime IS NOT NULL
ORDER BY ptime DESC;
select username from dba_users where username not in (
'ANONYMOUS',
'AURORA',
'$JIS',
'$UTILITY',
'$AURORA',
'$ORB',
'$UNAUTHENTICATED',
'CTXSYS',
'DBSNMP',
```

```
  set  long 10000
     select
              VIEW_DEFINITION
          from
              V$FIXED_VIEW_DEFINITION
         where
              view_name='GV$SQL_MONITOR';
```

```
alter session enable parallel DDL;
```

```
db where status='INVALID';
SELECT DBMS_METADATA.GET_DDL('PACKAGE_BODY',db.OBJECT_NAME,'db.owner')|| '/' from dba_objects
db where status='INVALID' and object_type='PACKAGE BODY';
SELECT DBMS_METADATA.GET_DDL(db.OBJECT_TYPE,db.OBJECT_NAME,db.OWNER)|| '/' from dba_objects
db where status='INVALID' and object_type not like '%PACKAGE%';


select 'select dbms_metadata.get_ddl(''TABLE'', '''||TABLE_NAME||''',''<schema>'') from dual;' FROM DBA_TABLES;

SELECT DBMS_METADATA.GET_DEPENDENT_DDL('REF_CONSTRAINT','<table_name>','<schema>') from dual;
```

http://asktom.oracle.com/pls/asktom/f?p=100:11:0::::P11_QUESTIC

The general syntax of GET_DDL is
GET_DDL(object_type, name, schema, version, model, transform).

```
set lines 1000 long 2000 pages 9999
select dbms_metadata.get_ddl('TABLE', table_name) from
user_tables where table_name in
```

```
')||'database link '||CHR(10)
||DECODE(U.NAME,'PUBLIC',Null, U.NAME||'.')|| L.NAME||chr(10)
||'connect to ' || L.USERID || ' identified by '
||L.PASSWORD||' using ''' || L.host || ''''
||chr(10)||';' TEXT
FROM sys.link$ L, sys.user$ U
```

```
select 'create tablespace ' || df.tablespace_name || chr(10)
 || ' datafile ''' || df.file_name || ''' size ' || df.bytes
 || decode(autoextensible,'N',null, chr(10) || ' autoextend on
maxsize '
 || maxbytes)
 || chr(10)
 || 'default storage ( initial ' || initial_extent
```

```
spool d1riskusercreate.sql
set pagesize 0
set escape on
select 'create user ' || U.username || ' identified ' ||
DECODE(password,
      NULL, 'EXTERNALLY',
      ' by values ' || '''' || password || ''''
      )
|| chr(10) ||
'default tablespace ' || default_tablespace || chr(10) ||
'temporary tablespace ' || temporary_Tablespace || chr(10) ||
select 'grant ' || GRANTED_ROLE || ' to ' || ROLE || ';' from
role_role_privs where role='&ROLE'
union
select 'grant ' || PRIVILEGE || ' to ' || ROLE || ';' from role_sys_privs
where role='&&ROLE'
union
select 'grant ' || PRIVILEGE || ' on ' || OWNER || '.'
||TABLE_NAME ||  ' to ' || GRANTEE || ';' from dba_tab_privs
where GRANTEE='&&ROLE' ;
```

```
set long 100000
set longchunksize 200
set heading off
set feedback off
set echo off
set verify off
undefine role

select (case
when ((select count(*)
from dba_roles
```

```
select * from DBA_TAB_PRIVS where grantee like
upper('%&enter_username%');
select * from DBA_ROLE_PRIVS where grantee like
upper('%&enter_username%');
select * from DBA_SYS_PRIVS where grantee like
upper('%&enter_username%');
select * from DBA_COL_PRIVS where grantee like
upper('%&enter_username%');
```

```
select
  case when level = 1 then own || '.' || obj || ' (' || typ || ')' else
  lpad (' ', 2*(level-1)) || obj || nvl2 (typ, ' (' || typ || ')', null)
  end
from
  (
  /* THE OBJECTS */
    select
      null        p1,
```

granted role permissions

```
select * from ROLE_ROLE_PRIVS where ROLE IN (select
granted_role from USER_ROLE_PRIVS where USERNAME
USER);
select * from ROLE_TAB_PRIVS  where ROLE IN (select
granted_role from USER_ROLE_PRIVS where
USERNAME=USER);
select * from ROLE_SYS_PRIVS  where ROLE IN (select
granted_role from USER_ROLE_PRIVS where
USERNAME=USER);
```

http://tech.padipa.net/generating-create-scripts-through-dbms_meta

N_ID:494205100346718343

```
set long 20000 longchunksize 20000 pagesize 9999 linesize 1000
feedback off verify off trimspool on
column ddl format a1000

begin
   dbms_metadata.set_transform_param
(dbms_metadata.session_transform, 'SQLTERMINATOR', true);
   dbms_metadata.set_transform_param
(dbms_metadata.session_transform, 'PRETTY', true);
end;
```

```
select dbms_metadata.get_ddl('USER', 'KPHU000') || '/' usercreate from dual;
SELECT
```

try this

Find locked Object

To find who is accessing the particular Table

Who is accessing the object

What a sid executed (sql_text of blocker)

To find Blocking GOOD query

Check who is blocking who in RAC, including objects

what sql_id ,sql_text are waiting due to blocking session

blocking session history

Reference Object

To Find Lock & Session - Metalink

Locked statistics Table

who is using my dblink

https://github.com/gwenshap/Oracle-DBA-Scripts/blob/master/locks.sql

```
 col username       format a40
 col sess_id        format a10
 col object format a50
 col mode_held      format a10
 select     oracle_username || ' (' || s.osuser || ')' username
,  s.sid || ',' || s.serial# sess_id
,  owner || '.' || object_name object
,  object_type
```

```
s.machine, o.object_name, t1.ctime
FROM
v$ACCESS    A,
v$LOCK     T1,
```

```
gv$access a, gv$session b
where a.object='&object_name'
and a.sid=b.sid
and a.inst_id=b.inst_id
```

All sql executed by a sid

```
       o.sql_text,
       o.address,
       o.hash_value,
       o.user_name,
```

```
col blocking_status for a100
 select s1.inst_id,s2.inst_id,s1.username || '@' || s1.machine
 || ' ( SID=' || s1.sid || ' )  is blocking '
 || s2.username || '@' || s2.machine || ' ( SID=' || s2.sid || ' ) ' AS
blocking_status , round(s2.seconds_in_wait/60)||' mins' Blocking_mins
```

```
machine, do.object_name as locked_object,id1, id2, lmode, request,
gv$lock.type
FROM gv$lock join gv$session on gv$lock.sid=gv$session.sid and
gv$lock.inst_id=gv$session.inst_id
join gv$locked_object lo on gv$lock.SID = lo.SESSION_ID and
```

```
 m.SID,
 sn.SERIAL#,
 m.TYPE,
 decode(LMODE,
  0, 'None',
select sample_time, session_state, blocking_session,
owner||'.'||object_name||':'||nvl(subobject_name,'-') obj_name,
   dbms_ROWID.ROWID_create (
      1,
      o.data_object_id,
      current_file#,
      current_block#,
      current_row#
   ) row_id
from dba_hist_active_sess_history s, dba_objects o
where user_id = 92
and sample_time between
   to_date('29-SEP-12 04.55.02 PM','dd-MON-yy hh:mi:ss PM')
   and
   to_date('29-SEP-12 05.05.02 PM','dd-MON-yy hh:mi:ss PM')
and event = 'enq: TX - row lock contention'
and o.data_object_id = s.current_obj#
order by 1,2;
SELECT p.table_name "Parent Table", c.table_name "Child Table",
p.constraint_name "Parent Constraint", c.constraint_name "Child
Constraint"      FROM user_constraints p      JOIN user_constraints c
ON(p.constraint_name=c.r_constraint_name)      WHERE
(p.constraint_type = 'P' OR p.constraint_type = 'U')      AND
c.constraint_type = 'R'      AND p.table_name = UPPER('&table_name');
```

```
0001_Gblocking.sql
---------------------------
set linesize 200
set pagesize 1000

column username format a10
column module format a50
column blocker format a7
column waiter format a7
column lmode format 9999
column request format 9999
column inst_id format 9999
column sid format 9999
col username format a6
col sid format 9999
col osuser format a14
col s# format 99999
col server format a10
col client format a10
col pname format a10

spool locking_information

select to_char(sysdate, 'DD-MON-YYYY HH24:MI:SS' ) current_time from
dual
/

prompt ########################
prompt # Blocking Information #
prompt ########################
select b.inst_id||'/'||b.sid blocker,
-- s.module,
w.inst_id||'/'||w.sid waiter,

select owner, table_name, stattype_locked from dba_tab_statistics where st
```

who is querying via dblink?
 Courtesy of Tom Kyte, via Mark Bobak
 this script can be used at both ends of the database link
 to match up which session on the remote database started
 the local transaction
 the GTXID will match for those sessions
 just run the script on both databases

```
Select /*+ ORDERED */
substr(s.ksusemnm,1,10)||'-'|| substr(s.ksusepid,1,10)      "ORIGIN",
substr(g.K2GTITID_ORA,1,35) "GTXID",
substr(s.indx,1,4)||'.'|| substr(s.ksuseser,1,5) "LSESSION" ,
s2.username,
substr(
   decode(bitand(ksuseidl,11),
      1,'ACTIVE',
      0, decode( bitand(ksuseflg,4096) , 0,'INACTIVE','CACHED'),
      2,'SNIPED',
      3,'SNIPED',
      'KILLED'
   ),1,1
) "S",
substr(w.event,1,10) "WAITING"
from  x$k2gte g, x$ktcxb t, x$ksuse s, v$session_wait w, v$session s2
where  g.K2GTDXCB =t.ktcxbxba
and   g.K2GTDSES=t.ktcxbses
and  s.addr=g.K2GTDSES
and  w.sid=s.indx
and s2.sid = w.sid;
```

```
COLUMN username FORMAT A20
COLUMN object_owner FORMAT A20
COLUMN object_name FORMAT A30
COLUMN locked_mode FORMAT A15

SELECT b.inst_id,
       b.session_id AS sid,
       NVL(b.oracle_username, '(oracle)') AS username,
```

```
gv$lock.sid sess, machine, do.object_name as
locked_object,id1, id2, lmode, request, gv$lock.type
FROM gv$lock join gv$session on
gv$lock.sid=gv$session.sid and
gv$lock.inst_id=gv$session.inst_id
join gv$locked_object lo on gv$lock.SID =
lo.SESSION_ID and gv$lock.inst_id=lo.inst_id
join dba_objects do on lo.OBJECT_ID =
```

blocker_sql

```
  from (select s.sid, txt.sql_text
          from gv$sqltext txt, gv$session s, gv$lock l
          where txt.address = s.sql_address
          and s.sid = l.sid
```

waiting_sql

```
  select sid as waiting_sid, sql_text as waiting_sql
  from (select s.sid, txt.sql_text
          from gv$sqltext txt, gv$session s, gv$lock lb,
gv$lock lw
          where txt.address = s.sql_address
```

```
set lines 750 pages 9999
col blocking_status for a100
SELECT s1.inst_id,s2.inst_id, 'Instance '||s1.INST_ID||'
'|| s1.username || '@' || s1.machine || ' ( SID=' || s1.sid ||
','|| s1.serial#||' '||s1.status||  ','||s1.sql_id||'  )
```

```
SELECT
   l1.sid || ' is blocking ' || l2.sid blocking_sessions
FROM
   v$lock l1, v$lock l2
WHERE
   l1.block = 1 AND
```

```
select OS_USER_NAME os_user,
 PROCESS os_pid,
 ORACLE_USERNAME oracle_user,
 l.SID oracle_id,
 decode(TYPE,
  'MR', 'Media Recovery',
col sql_id format a15
col inst_id format '9'
col sql_text format a50
col module format a10
col blocker_ses format '999999'
col blocker_ser format '999999'
SELECT distinct
       a.sql_id ,
       a.inst_id,
       a.blocking_session blocker_ses,
       a.blocking_session_serial# blocker_ser,
       a.user_id,
       s.sql_text,
       a.module
FROM  GV$ACTIVE_SESSION_HISTORY a,
      gv$sql s
where a.sql_id=s.sql_id
  and blocking_session is not null
  and a.user_id <> 0 --  exclude SYS user
  and a.sample_time > sysdate - 1
/
```

ORA-02266: unique/primary keys in table referenced by enabled foreign keys

SELECT * from dba_waiters;	INST_ID, sid,BLOCKING_SESSION,BLOCKING_instance from gv$sessio
https://github.com/gwenshap/Oracle-DBA-Scripts/blob/master/locks.sql http://www.oracle-	SELECT s.blocking_session, s.sid,

Trace enable

Trace disable

Finding own sid & serial number

Start session trace

Tracing other user's sessions 10g

tracing in 11g (use this)

Trace a module:

Trace a specific sql_id

Check active events in oracle database

Trace using Trigger

Other important traces

Tracing an entire database

Tkprof

DB HUNG - OS level

Database HUNG - db level

to see wait events and bind variables for a different session

10046 Trace

10046   Tracing existing process

For any sql_id that is in the shared pool (since 11gR2)

Login to db when hung

Tracing RMAN

Tracing Oracle process OS level

Tracing SQLPLUS at OS level

Tracing your own process using sqlplus

Oradebug utility

adrci

adrci create package

10132 Trace

Check this

parallel session trace

How to dump ASH info to file when db is hang

http://tinky2jed.wordpress.com/technical-stuff/oracle-stuff/what-is-the-correct-way-to-trace-a-session-in-oracle/
http://alexzeng.wordpress.com/2008/08/01/how-to-trace-oracle-sessions/
http://www.oracle-developer.net/display.php?id=516
https://mahmoudhatem.wordpress.com/2015/07/23/trcextprof-sql-the-raw-trace-file-10046-profiler-based-on-exterr

```
 ALTER SESSION SET EVENTS '10046 TRACE NAME CONTEXT FOREVER, LEVEL 12';
-or-
  EXEC SYS.DBMS_SYSTEM.SET_EV(SID,SERIAL#,10046,12,'');
-or-
  EXEC SYS.DBMS_SUPPORT.START_TRACE_IN_SESSION(SID,SERIAL#,WAITS=>TRUE,BINDS=>FALSE);
-or-
  EXEC
DBMS_MONITOR.SESSION_TRACE_ENABLE(SESSION_ID=>SID,SERIAL_NUM=>SERIAL#,WAITS=>TRUE,
BINDS=>TRUE)
-or-
  EXEC SYS.DBMS_SYSTEM.SET_SQL_TRACE_IN_SESSION(SID,SERIAL#,TRUE);

ALTER SESSION SET EVENTS '10046 TRACE NAME CONTEXT OFF';
-or-
  EXEC SYS.DBMS_SYSTEM.SET_EV(SID,SERIAL#,10046,0,'');
-or-
  EXEC SYS.DBMS_SUPPORT.STOP_TRACE_IN_SESSION(SID,SERIAL#);
-or-
  EXEC SYS.DBMS_SYSTEM.SET_SQL_TRACE_IN_SESSION(SID,SERIAL#,FALSE);

select username,inst_id, sid, serial# FROM gV$SESSION WHERE audsid = userenv('sessionid');


ALTER SESSION SET sql_trace = true;
ALTER SESSION SET sql_trace = false;

execute dbms_system.set_sql_trace_in_session(501, 44396, true);

execute dbms_system.set_sql_trace_in_session(501, 44396, false);
Look for trace file in USER_DUMP_DEST:
SQL> exec dbms_monitor.session_trace_enable(session_id=>3,serial_num=>5027,binds=>true,waits=>true);
PL/SQL procedure successfully completed.
SQL> exec dbms_monitor.session_trace_disable(session_id=>3,serial_num=>5027);
PL/SQL procedure successfully completed.

EXECUTE DBMS_MONITOR.SERV_MOD_ACT_TRACE_ENABLE(
     service_name=>'vasont.world', module_name=>'VasontU.exe',
     action_name=>DBMS_MONITOR.ALL_ACTIONS,waits=>TRUE,
     binds=>TRUE,instance_name=>NULL);
EXECUTE DBMS_MONITOR.SERV_MOD_ACT_TRACE_DISABLE(
  service_name=>'vasont.world',module_name=>'VasontU.exe');
alter system set events 'sql_trace[SQL:8kybysnu4nn34] plan_stat=all_executions,wait=true,bind=true';        (OR)
alter system set events 'sql_trace[sql: cjrha4bzuupzf]  level=12';



alter system set events 'sql_trace[SQL:8kybysnu4nn34] off';
```

```
declare
event_level number;
begin
  for i in 10000..10999 loop
   sys.dbms_system.read_ev(i,event_level);
   if (event_level > 0) then
      dbms_output.put_line('Event '||to_char(i)||' set at level '|| to_char(event_level));
   end if;
  end loop;
end;
/
```

```
create or replace trigger sqltrace
after logon on database
begin
  if user='XXXXX'
  then
    execute immediate 'alter session set events ''10046 trace name context forever, level 4''';
  end if;
end;
/
```

10132

```
ALTER SYSTEM SET sql_trace = true SCOPE=MEMORY;
ALTER SYSTEM SET sql_trace = false SCOPE=MEMORY;
```

```
$ tkprof orcl102_ora_3064.trc output.prf EXPLAIN=scott/tiger SYS=NO
tkprof ora10g_ora_5868.trc tracea.prf    sort=fchcpu,prscpu,execpu
```

```
$ ps -ef| egrep '4711|4713'
$ truss -aefo /tmp/truss.out -p <PID>
strace -f -o output.txt sqlplus / as sysdba
```

```
$ sqlplus /nolog

### SQL> set _prelim on

SQL> connect / as sysdba

SQL> oradebug setorapname reco

SQL> oradebug -g all hanganalyze 3

exec sys.dbms_system.set_ev([SID], [SERIAL], 10046, 12, '');
exec dbms_system.set_ev(si=>123, se=>1234, ev=>10046, le=>8, nm=>' ');


SQL> SELECT p.spid, s.sid, s.serial#
FROM v$session s, v$process p
WHERE s.paddr = p.addr AND p.spid = 24078
SPID SID SERIAL#
------------ ---------- ----------
24078 18 5
SQL> begin dbms_system.set_ev(18,5, 10046,12,''); end; -- trace on
-- collect trace information for approximately 15 minutes during the problem
SQL> begin dbms_system.set_ev(18, 5, 10046,0,''); end; -- trace off
```

Find SID of that session

Find its OS pid from the following (This does not work for Windows)

```
select p.PID,p.SPID,s.SID from v$process p,v$session s where s.paddr = p.addr and s.sid = &SESSION_ID;
oradebug setospid spid??? (spid from above query)
oradebug unlimit
oradebug event 10046 trace name context forever,level 12;
execute DBMS_SQLDIAG.DUMP_TRACE(p_sql_id=>'&sid',
 p_child_number=>0,
 p_component=>'Optimizer',
 p_file_id=>'TRACE_10053');
```

```
sqlplus -prelim / as sysdba

SQL*Plus: Release 10.2.0.2.0 - Production on Mon Dec 11 17:28:14 2006

SQL> oradebug setmypid
Statement processed.
SQL> oradebug unlimit
Statement processed.
SQL> oradebug dump systemstate 266
```

Metalink Note : 452358.1

```
rman rcvcat rman/rman@catalog target user/passwd@target debug trace trace_file_name

strace -fF -v -p 16311   -o  output.txt
strace /oracle/product/10.2.0.1/bin/sqlplus -V 2>&1 |less
```

```
 host strace -e trace=open -p #unix_pid & echo $! > .tmp.pid
```

oradebug is a debugging utility that is essentially undocumented and is intended for use by Oracle support analysts for various tasks one of which is that it can be used to set trace. oradebug is available from svrmgrl before Oracle 9i and from SQL*Plus after. The first step in using this tool is to find the OS PID or the Oracle PID of the process you want to analyse. You can do this as follows:

```
 SQL> connect system/manager@sans as sysdba
 Connected.
 SQL> col sid for 999999
 SQL> col serial# for 999999
 SQL> col spid for a8
 SQL> col username for a20
```

```
adrci> show homes
diag/tnslsnr/servername/listener
adrci> set home diag/tnslsnr/dbname/listener
adrci> purge -age 2880 -type ALERT
adrci> purge -age 2880 -type TRACE
```

So to purge all diagnostic information, including trace files, older than 2 days you would issue the following.

```
adrci> purge -age 2880
```

```
adrci> show homes
adrci> set homepath diag/rdbms/dbname/insatncename
adrci> show incident
adrci> ips pack incident 90737 in /tmp
```

For 11.2 and higher, issue "$GRID_HOME/bin/diagcollection.sh" as root on all nodes.
    For 10.2 and 11.1, issue "$CRS_HOME/bin/diagcollection.pl -crshome=$CRS_HOME --collect" as root on all

```
alter session set tracefile_identifier='My_trace';

alter session set events '10132 trace name context forever, level 12';

alter session set events '10132 trace name context off';

SQL> exec dbms_sqldiag.dump_trace(p_sql_id=>'<SQL_ID>', p_child_number=>0, p_component=>'Compiler',
p_file_id=>'CBO_TRACE_DEV');
SQL> exec dbms_sqldiag.dump_trace(p_sql_id=>'<SQL_ID>', p_child_number=>0, p_component=>'Compiler',
p_file_id=>'CBO_TRACE_PRD');

alter session set "_px_trace"=high,all;

SQL> oradebug setmypid
SQL> oradebug dump ashdump 5 — This will dump last 5 minute content
SQL> oradebug tracefile_name
```

al-tables-regexp/

select sys_context ('USERENV' , 'SID') OwnSID from dual;

You can also add an identifier to the trace file name for later identification:
ALTER SESSION SET sql_trace = true;
ALTER SESSION SET tracefile_identifier = mysqltrace;
http://www.orafaq.com/wiki/SQL_Trace

Get the SID & Serial number from Row 1

tkprof ora10g_ora_5868.trc tracea.prfs sort=fchcpu,prscpu,execpu

https://blog.dbi-services.com/enable-10046-tracing-for-a-specific-sql/
https://avdeo.com/tag/sql-tuning/
https://oracle-base.com/articles/misc/sql-trace-10046-trcsess-and-tkprof
http://www.juliandyke.com/Diagnostics/Trace/EnablingTrace.php

```
create or replace trigger trace_trigger_scott
    AFTER LOGON ON DATABASE
    WHEN (USER='SCOTT')
    declare
   stmt varchar2(100);
   hname varchar2(20);
   uname varchar2(20);
   begin
       select sys_context('USERENV','HOST'),
              sys_context('USERENV','SESSION_USER')
       into hname,uname from dual;
       stmt := 'alter session set tracefile_identifier='||hname||'_'||uname;
       EXECUTE IMMEDIATE stmt;
   EXECUTE IMMEDIATE 'alter session set sql_trace=true';
     end;
    /
```

and disable tracing when the user disconnects:

```
-----------------------------------------------
create or replace trigger trace_trigger_off
  BEFORE LOGOFF ON DATABASE
  when(user='SCOTT')
begin
  execute immediate 'alter session set sql_trace=false';
  end;
/
```

https://jonathanlewis.wordpress.com/2006/11/27/event-10132/

http://www.orafaq.com/wiki/TKProf
best

solaris
linux

```
connect / as sysdba
oradebug setospid 9834
oradebug unlimit
oradebug event 10046 trace name context forever,level 12
oradebug tracefile_name
--wait for 5 mins and then trace off
oradebug event 10046 trace name context off
Find SID of that session
Find its Orable pid from the following (This does work for Windows)
select p.PID,p.SPID,s.SID from v$process p,v$session s where s.paddr = p.addr and
s.sid = &SESSION_ID;
oradebug setorapid pid??? (pid from above query)
oradebug unlimit
oradebug event 10046 trace name context forever,level 12;
```

```
sqlplus '/ as sysdba'
oradebug setmypid
oradebug unlimit
oradebug hanganalyze 3
-- Wait one minute before getting the second hanganalyze
oradebug hanganalyze 3
oradebug tracefile_name
exit
```

for linux

```
SQL> column spid new_value unix_pid
SQL> select spid from v$process p join v$session s on p.addr=s.paddr and
s.sid=sys_context('userenv','sid');
SPID
------------------------
10914
```

```
SQL> host strace -e trace=open -p #unix_pid & echo $! > .tmp.pid
Process 10914 attached - interrupt to quit
```

```
SQL> declare
  p_file utl_file.file_type;
begin
  p_file := utl_file.fopen ('MYDIR', 'myfile2.txt', 'w');
end;
```

```
for i in `adrci exec="show homes"|grep listener`;do
 echo "adrci exec=\"set home $i;purge\""
 adrci exec="set home $i;purge";
 # maybe check $? here
done
```

```
select distinct sid OwnSID from v$mystat;
```

```
BEGIN
  for c in (select s.sid     sid,
                   s.serial# serial
            from   v$session s
            where  s.USERNAME = 'ASAP' and
                   s.status = ’ACTIVE’)
  loop
    dbms_support.start_trace_in_session(c.sid, c.serial);
  end loop;
END;
```

https://blog.dbi-services.com/oracle-is-hanging-dont-forget-hanganalyze-and-systemstate/

```
alter session set tracefile_identifier='mytrace_1089';
alter session set events '10046 trace name context forever,level
12:1089 trace name errorstack level 3';
select * from dual@DBLINK;
alter session set events '10046 trace name context off';
```

```
SQL> alter session set tracefile_identifier='trap_03150';
 SQL> alter session set events '10046 trace name context
forever,level 12:3150 trace name errorstack level 3';
```

Systemstate

```
sqlplus '/ as sysdba'
oradebug setmypid
oradebug unlimit
oradebug dump systemstate 266
oradebug dump systemstate 266
oradebug tracefile_name
exit
```

Invalid Objects

Details of a object

Recomplinig Invalid objects (nonpdb) --> for pdb refer next column

Script to recompile invalid objects

Compile errors

Dropping all objects under ur schema

How to recompile a pakage?

Objects count order by owner

row count of all tables in a schema

all Indexes for a table

Find Table_name from index_name
Disable Trigger

Disable constraint

Disable constraints

Dependency Objects

count of columns of a table in a schema

dependent objects

ORA-02292: integrity constraint (OWNER.CONSTRAINT_NAME) violated - child record found

BLOB/CLOB Objects tables in db

Find duplicate records in a column

Source code of all invalid objects

Dblinks

Select * from dba_db_links;

Drop Private db link

Rebuild / Fragmentation

Indexes requires rebuild
Rebuild Indexes

Index fragmentation

Index fragmentation other way

Unused indexes

Tables requires rebuild

TABLE Fragmentation / All Tables

re-organizing table

```
select owner, object_type, substr(object_name,1,30) object_name, status
from dba_objects where status='INVALID' order by object_type;
set lines 1000 pages 9999
col object_name for a50
col owner for a30
col object_type for a30
col STATUS for a30
select owner,object_name,object_type,CREATED,STATUS,TO_CHAR(LAST_DDL_TIME,'dd-mon-yyyy
hh24:mi:ss') AS LAST_DDL_TIME from dba_objects where object_name in
('&OBJECT_NAME1','&OBJECT_NAME2','&OBJECT_NAME3');
select 'alter '||object_type||' '||owner||'.'||object_name||' compile;'
from dba_objects
where status<>'VALID'
and object_type not in ('PACKAGE BODY','TYPE BODY','UNDEFINED','JAVA CLASS','SYNONYM')
union
select 'alter package '||owner||'.'||object_name||' compile body;'
from dba_objects
where status<>'VALID'
and object_type='PACKAGE BODY'
union
select 'alter type '||owner||'.'||object_name||' compile body;'
from dba_objects
where status<>'VALID'
set serveroutput on size 100000;
declare
 sqlstring varchar2(2000);
 rec_count integer;
begin
 dbms_utility.compile_schema('SCHEMA1');
 dbms_utility.compile_schema('SCHEMA2');
 select count(1) into rec_count from sys.all_objects
 where owner in ('SCHEMA1','SCHEMA2')
 and object_type in ('FUNCTION','VIEW','PROCEDURE','TRIGGER') and status = 'INVALID';
 if (rec_count > 0) then
  for recs in (select owner, object_type, object_name from sys.dba_objects where owner in
('SCHEMA1','SCHEMA2') and object_type in ('FUNCTION','VIEW','PROCEDURE','TRIGGER') and status

SQL> alter view OWNER.VIEW1 compile;

Warning: View altered with compilation errors.

SQL> select text from dba_errors where name='VIEW1';
ORA-28545: error diagnosed by Net8 when connecting to an agent
Unable to retrieve text of NETWORK/NCR message 65535

ORA-02063: preceding 2 lines from TUBPR01
ORA-28545: error diagnosed by Net8 when connecting to an agent
Unable to retrieve text of NETWORK/NCR message 65535
```

```
alter session set current_schema=&SCHEMA;

DECLARE
BEGIN
  FOR r1 IN ( SELECT 'DROP ' || object_type || ' ' || owner || '."'|| object_name || '"' || DECODE (
object_type, 'TABLE', ' CASCADE CONSTRAINTS PURGE' ) AS v_sql  FROM dba_objects
               WHERE owner in ('&SCHEMA') and object_type IN ( 'TABLE', 'VIEW', 'PACKAGE', 'TYPE',
'PROCEDURE', 'FUNCTION', 'TRIGGER', 'SEQUENCE', 'MATERIALIZED VIEW', 'JAVA CLASS' )
               ORDER BY object_type,
                        object_name ) LOOP
   EXECUTE IMMEDIATE r1.v_sql;
  END LOOP;
END;
/
purge dba_recyclebin;

ALTER PACKAGE pkg1 COMPILE REUSE SETTINGS;


break on owner skip1 on object_type
compute  sum of cnt on owner
col owner for a30
select distinct owner,object_type, count(*) cnt
from dba_objects
group by  object_type,owner order by owner;

clear break
clear computes
select
      table_name,
      to_number(
        extractvalue(
          xmltype(dbms_xmlgen.getxml(‘select count(*) c from ‘||table_name))
          ,’/ROWSET/ROW/C’)
          )
          count
    from user_tables order by table_name;

col index_name for a30
col table_name for a30
col column_name for a30
select
owner,b.uniqueness, a.index_name, a.table_name, a.column_name
from dba_ind_columns a, dba_indexes b
```

```
 col owner for a30
col index_name for a30
col table_name for a30
col column_name for a30
select owner,b.uniqueness, a.index_name, a.table_name, a.column_name
from dba_ind_columns a, dba_indexes b
where a.index_name=b.index_name
and b.owner='&OWNER'
and b.index_name = upper('&INDEX_NAME')
order by a.table_name, a.index_name, a.column_position;
select 'ALTER TRIGGER '||OWNER||'.'||TRIGGER_NAME||' DISABLE '||';' from dba_triggers where table_n

select 'ALTER TABLE '||OWNER||'.'||TABLE_NAME||' DISABLE CONSTRAINT '||CONSTRAINT_NAME ||';'
  SELECT
      'ALTER TABLE '||OWNER||'.'||
      TABLE_NAME||' '||var_action||' CONSTRAINT '||CONSTRAINT_NAME AS sql_string,
      CONSTRAINT_NAME
   FROM
      ALL_CONSTRAINTS
   WHERE
      CONSTRAINT_TYPE='R' AND OWNER=Target_Schema_Name;


select name, type, owner
    from all_dependencies
    where referenced_owner = 'USER_NAME'
    and referenced_name = 'OBJECT_NAME';
col owner for a30
col table_name for a50
 select t.owner, t.table_name, t.num_rows, count(*)
from dba_tables t left join dba_tab_columns c on t.table_name = c.table_name
where  t.owner='&OWNER_NAME' and num_rows is not null
SELECT TYPE
  || ' '
  || OWNER
  || '.'
  || NAME
SELECT owner, constraint_name, constraint_type, table_name, r_owner, r_constraint_name
    FROM all_constraints
    WHERE owner='&OWNER'
set lines 750 pages 999
col owner for a30
col table_name for a50
select column_name, count(column_name)
from table
group by column_name


SELECT DBMS_METADATA.GET_DDL(db.OBJECT_TYPE,db.OBJECT_NAME,db.OWNER)|| '/' from dba
```

```
col owner for a30
col db_link for a50
col username for a30
col host for a100
Select * from dba_db_links;


set serveroutput on
DECLARE
  l_sql CLOB :=
   'CREATE PROCEDURE <OWNER>.drop_db_links_prc
    IS
```

```
select a.*, round(index_leaf_estimate_if_rebuilt/current_leaf_blocks*100) percent, case when
index_leaf_estimate_if_rebuilt/current_leaf_blocks < 0.5 then 'candidate for rebuild' end status
from
(
select table_name, index_name, current_leaf_blocks, round (100 / 90 * (ind_num_rows * (rowid_length +
uniq_ind + 4) + sum((avg_col_len) * (tab_num_rows) ) ) / (8192 - 192) ) as index_leaf_estimate_if_rebuilt
from (
select tab.table_name, tab.num_rows tab_num_rows , decode(tab.partitioned,'YES',10,6) rowid_length ,
ind.index_name, ind.index_type, ind.num_rows ind_num_rows, ind.leaf_blocks as current_leaf_blocks,
decode(uniqueness,'UNIQUE',0,1) uniq_ind,ic.column_name as ind_column_name, tc.column_name ,
tc.avg_col_len
from dba_tables tab
join dba_indexes ind on ind.owner=tab.owner and ind.table_name=tab.table_name
join dba_ind_columns ic on ic.table_owner=tab.owner and ic.table_name=tab.table_name and
ic.index_owner=tab.owner and ic.index_name=ind.index_name
join dba_tab_columns tc on tc.owner=tab.owner and tc.table_name=tab.table_name and
tc.column_name=ic.column_name
where tab.owner='&OWNER' and ind.leaf_blocks is not null and ind.leaf_blocks > 1000
) group by table_name, index_name, current_leaf_blocks, ind_num_rows, uniq_ind, rowid_length
) a where index_leaf_estimate_if_rebuilt/current_leaf_blocks < 0.5
order by index_leaf_estimate_if_rebuilt/current_leaf_blocks;

SELECT ' alter index ' || owner || '.' || index_name || ' rebuild ; ' FROM dba_indexes where owner not in ('SY

define owner='&OWNER'         -- table owner
define table='&TABLE_NAME'    -- table name
define index='&INDEX_NAME'    -- index name
define buckets=10           -- number of buckets
define sample=100           -- 100% scans all the index

column "free" format A5

variable c refcursor;
```

```
ANALYZE INDEX  emp_id_idx  VALIDATE STRUCTURE;
SELECT
    name
  , del_lf_rows
  , lf_rows - del_lf_rows lf_rows_used
  , TO_CHAR( del_lf_rows /(DECODE(lf_rows,0,0.01,lf_rows))*100,'999.99999') ibadness
FROM   index_stats;
SELECT table_name, index_name
from user_indexes i
WhERE uniqueness <> 'UNIQUE'
AND index_name NOT IN
(sELECT DISTINCT object_name
from v$sql_plan
where operation LIKE '%INDEX%'
AND object_owner='&OWNER');

select owner,table_name,round((blocks*8),2)||' kb' "TABLE SIZE",round((num_rows*avg_row_len/1024),2)|
select
   table_name,round((blocks*8),2) "size (kb)" ,
   round((num_rows*avg_row_len/1024),2) "actual_data (kb)",
   (round((blocks*8),2) - round((num_rows*avg_row_len/1024),2)) "wasted_space (kb)"
from
   dba_tables
where  owner='&OWNER' and table_name='&TABLE_NAME' and
select
‘exec analyzedb.reorg_a_table4(’||””||rtrim(t.table_owner)||””||’,’||””||
rtrim(t.table_name)||””||’);’,
t.table_owner||’.'||t.table_name name,
a.num_rows,
set lines 750 pages 9999
col owner for a30
col object_name for a50
col object_type for a30
select 'select owner,object_name,object_type from dba_objects where object_name='''||
object_name ||''' and owner='''|| owner ||''';' from dba_objects where status='INVALID';
```

Set this below parameter if u want to run the compilation script in PDB

```
 alter session set "_ORACLE_SCRIPT"=true;

Session altered.
```

```
 set lines 750 pages 9999
 col text for a150
 col owner for a15
 col name for a50
 col position for a10
SELECT e.owner, e.name, TO_CHAR(e.line) || '/' || TO_CHAR(e.position) "POSITION",
e.text
   FROM dba_errors e
   ORDER BY e.owner, e.name, e.sequence;
```

```
select 'drop '||object_type||' '|| object_name||  DECODE(OBJECT_TYPE,'TABLE','
CASCADE CONSTRAINTS;',';')
from user_objects;

select 'drop java class "'||object_name||'";' from dba_objects WHERE owner in
('&SCHEMA');
```

```
ame in  ('&TABLE_NAME1') and owner='&OWNER';
```

```
 from dba_constraints where table_name in  ('&TABLE1') and owner='&OWNER';
```

```
set serveroutput on size 1000000
DECLARE
CURSOR get_ts IS SELECT * FROM dba_tablespaces
                    WHERE tablespace_name != 'SYSTEM';
```

https://rodgersnotes.wordpress.com/2011/12/27/scripts-to-find-object-dependencies/

https://bommaritollc.com/2012/01/22/debugging-ora-02292-integrity-constraint-owner-cons

```
a_objects db where status='INVALID';
```

```
S','SYSTEM');
```

https://blog.dbi-services.com/how-i-measure-oracle-index-fragmentation/

```
|' kb' "ACTUAL DATA" from dba_tables where table_name='&Table_name';
```

Table size (with fragmentation)

```
SQL> select table_name,round((blocks*8),2)||'kb' "size"
 from user_tables
 where table_name = 'BIG1';

TABLE_NAME size
```

```
exec dbms_pdb.exec_as_oracle_script('alter package SYS.DBMS_METADATA compile body;');
```

```
SET ECHO ON  SERVEROUTPUT ON SIZE 1000000
BEGIN
  FOR cur_rec IN (SELECT object_type, owner,
object_name FROM dba_objects WHERE
owner='&OWNER' and object_type IN ('TABLE',
'VIEW', 'PACKAGE', 'PROCEDURE', 'FUNCTION',
'SEQUENCE','SYNONYM','TYPE')) LOOP
    BEGIN
      IF cur_rec.object_type = 'TABLE' THEN
        EXECUTE IMMEDIATE 'DROP
'||cur_rec.object_type||'
'||cur_rec.owner||'.'||cur_rec.object_name||' CASCADE
CONSTRAINTS';
      ELSE
        EXECUTE IMMEDIATE 'DROP
'||cur_rec.object_type||'
'||cur_rec.owner||'.'||cur_rec.object_name||' ';
      END IF;
   EXCEPTION
      WHEN OTHERS THEN
        DBMS_OUTPUT.put_line('FAILED: DROP
'||cur_rec.object_type||'
```

```
select
owner,table_name,blocks,num_rows,avg_row_len,roun
d(((blocks*8/1024)),2) "TOTAL_SIZE( MB)",
round((num_rows*avg_row_len
 /1024/1024),2) "ACTUAL_SIZE(MB)",
round(((blocks*8/1024)-
(num_rows*avg_row_len/1024/1024)),2)
```

check this
v$parameter for undo

% UNDO tablespacee Free As per expired segments

does undo have expired segments ?

User Generated UNDO

UNDO tablespace usage by USER

Running Sql's

recreating the undo tblspce or resolving undo issue

ORA-30019: Illegal rollback Segment operation in Automatic Undo mode

active,expired, unexpired undo

active,expired, unexpired undo by Size in MB

Undo Tablespace Advice

undo advisor

Monitoring undo segements

how long will that session take to finish rolling back?

Undo Not releasing the unexpired segments

TEMP

sort space usage by sesison

History of Temp tablespace usage

shrink tempfile

http://vsbabu.org/oracle/sect07.html

```
select inst_id,name,value from gv$parameter where name like '%undo%';

SELECT d.tablespace_name, round(((NVL(f.bytes,0) + (a.maxbytes -
a.bytes))/1048576+ u.exp_space),2)
as max_free_mb, round(((a.bytes - (NVL(f.bytes,0)+
(1024*1024*u.exp_space)))*100/a.maxbytes),2)
used_pct FROM   sys.dba_tablespaces d, (select tablespace_name, sum(bytes) bytes,
sum(greatest(maxbytes,bytes)) maxbytes from sys.dba_data_files group by
tablespace_name) a,
(select tablespace_name, sum(bytes) bytes from sys.dba_free_space group by
tablespace_name) f ,
(select tablespace_name , sum(blocks)*8/(1024)  exp_space from
dba_undo_extents where status NOT IN ('ACTIVE','UNEXPIRED')  group by
tablespace_name) u
WHERE d.tablespace_name = a.tablespace_name(+) AND d.tablespace_name =
f.tablespace_name(+)
AND d.tablespace_name=u.tablespace_name  AND d.contents = 'UNDO' AND
u.tablespace_name = (select UPPER(value)
from v$parameter where name = 'undo_tablespace');
select status, count(1)
   from dba_undo_extents
   group by status;

col sql_text format a40
set lines 130
select sq.sql_text sql_text, t.USED_UREC Records, t.USED_UBLK Blocks,
(t.USED_UBLK*8192/1024) KBytes from v$transaction t,
v$session s,
v$sql sq
where t.addr = s.taddr
and s.sql_id = sq.sql_id
and s.username = '<user>'
select
  TO_CHAR(s.sid)||','||TO_CHAR(s.serial#) sid_serial,
  NVL(s.username, 'None') orauser,
  s.program,
  r.name undoseg,
  t.used_ublk * TO_NUMBER(x.value)/1024/1024||'M' "Undo"
col o format a10
col u format a10
select osuser o, username u, sid,
segment_name s, substr(sa.sql_text,1,200) txt
from v$session s,
v$transaction t,
select a.name,b.status , d.username , d.sid , d.serial#
   from v$rollname a,v$rollstat b, v$transaction c , v$session d
   where a.name IN ( select segment_name
   from dba_segments where tablespace_name = 'UNDOTBS1')
   and a.usn = b.usn
   and a.usn = c.xidusn
```

```
ALTER SYSTEM SET UNDO_SUPPRESS_ERRORS=TRUE scope=both;

select status,
  round(sum_bytes / (1024*1024), 0) as MB,
  round((sum_bytes / undo_size) * 100, 0) as PERC
from
(
  select status, sum(bytes) sum_bytes
  from dba_undo_extents
  group by status
 select status,TABLESPACE_NAME,sum(bytes) / 1024 / 1024 || 'MB' from
DBA_UNDO_EXTENTS group by status,TABLESPACE_NAME;
prompt << Estimate the required UNDO_Size for the above retention
>>
prompt << by this query: select
dbms_undo_adv.required_undo_size(<new_undo_retention>) new_undo_MB from dual
>>
prompt
======================================================================
======================================
prompt
SELECT d.undo_size/(1024*1024) "ACTUAL UNDO SIZE [MByte]",
       SUBSTR(e.value,1,25) "UNDO RETENTION [Sec]",
       ROUND((d.undo_size / (to_number(f.value) *
       g.undo_block_per_sec))) "OPTIMAL UNDO RETENTION [Sec]"
  FROM (
       SELECT SUM(a.bytes) undo_size
          FROM v$datafile a,
               v$tablespace b,
               dba_tablespaces c
         WHERE c.contents = 'UNDO'
           AND c.status = 'ONLINE'
           AND b.name = c.tablespace_name
           AND a.ts# = b.ts#
       ) d,
       v$parameter e,
       v$parameter f,
       (
       SELECT MAX(undoblks/((end_time-begin_time)*3600*24))
              undo_block_per_sec
         FROM v$undostat
       ) g
WHERE e.name = 'undo_retention'
  AND f.name = 'db_block_size'
/
```

```
SQL> select max(maxquerylen),max(tuned_undoretention) from v$undostat;
     SQL> select max(maxquerylen),max(tuned_undoretention) from dba_hist_undostat;

col undo_records    new_value undo_records_start
col time_now        new_value time_start
col session_id      new_value session_id
col session_serial  new_value session_serial

col 1 new_value 1
select 1 from dual where 1=2;
def session_specifier = '&1'

select dbms_utility.get_time as time_now
     , se.sid         as session_id
     , se.serial#     as session_serial
     , se.username
     , se.status      as session_status
     , tr.status      as transaction_status
     , tr.start_time  as transaction_start
     --, tr.name        as transaction_name
     --, tr.used_ublk   as undo_blocks
     , tr.used_urec   as undo_records
```

```
SELECT a.ksppinm "Parameter",
b.ksppstvl "Session Value",
c.ksppstvl "Instance Value"
FROM sys.x$ksppi a, sys.x$ksppcv b, sys.x$ksppsv c
WHERE a.indx = b.indx
AND a.indx = c.indx
AND a.ksppinm in ( '_undo_autotune' , '_smu_debug_mode' ,
'_highthreshold_undoretention' ,
'undo_tablespace' , 'undo_retention' , 'undo_management' )
 order by 2
 /
```

```
column sid format 9999
column username format a15
column SQL_EXEC_START for a21
column sql_text format a50
column module format a35
column sql_text format a50
break on report
compute SUM of MB_USED on report
 SELECT a.inst_id,a.username, a.sid, a.serial#, a.osuser,
(b.blocks*d.block_size)/1048576 MB_USED,a.sql_id,a.sql_child_number
child,c.plan_hash_value,to_char (a.sql_exec_start, 'dd-Mon-yyyy hh24:mi:ss')
sql_exec_start,c.rows_processed,a.status,
-- c.sql_text
select
sql_id,SQL_PLAN_HASH_VALUE,max(TEMP_SPACE_ALLOCATED)/(1024*1024*102
4) gig
from DBA_HIST_ACTIVE_SESS_HISTORY
where
sample_time > sysdate-10 and
TEMP_SPACE_ALLOCATED > (10*1024*1024*1024)
group by sql_id,SQL_PLAN_HASH_VALUE order by sql_id;
ALTER TABLESPACE temp SHRINK TEMPFILE
'+DATA_KRONIA/p1kronia/tempfile/temp.313.868020865' KEEP 500M;
```

```
  SELECT v$session.SID,
v$session.SERIAL#,r.NAME "Undo Segment
Name", dba_seg.size_mb,
DECODE(TRUNC(SYSDATE - LOGON_TIME), 0,
NULL, TRUNC(SYSDATE - LOGON_TIME) || '
Days' || ' + ') ||
```

https://community.oracle.com/thread/1098943?tstart=0
http://www.oracleflash.com/32/Change-or-switch-undo-tablespace-in-Oracle-database.html

If TRUE means it will not show any undo errors , So

```
select count(status) from dba_undo_extents where
status = 'EXPIRED';
```

```
select count(status) from dba_undo_extents where
status = 'UNEXPIRED';
```

```
select status, sum(bytes) / 1024 / 1024 || 'MB' from D
```

```
SELECT
dbms_undo_adv.required_undo_size(1800,
SYSDATE-30, SYSDATE)
FROM dual;
```

```
select maxquerysqlid, maxquerylen from
dba_hist_undostat order by maxquerylen desc;
                select maxqueryid, maxquerylen from
v$undostat order by maxquerylen desc;
```

http://www.williamrobertson.net/documents/undo_tra

```
SELECT   S.sid || ',' || S.serial# sid_serial,
S.username, S.osuser, P.spid, S.module,
         S.program, SUM (T.blocks) * TBS.block_size
/ 1024 / 1024 mb_used, T.tablespace,
         COUNT(*) sort_ops
FROM     v$sort_usage T, v$session S,
dba_tablespaces TBS, v$process P
WHERE    T.session_addr = S.saddr
AND      S.paddr = P.addr
AND      T.tablespace = TBS.tablespace_name
GROUP BY S.sid, S.serial#, S.username, S.osuser,
P.spid, S.module,
         S.program, TBS.block_size, T.tablespace
```

```
SELECT distinct rpad(s.sid,3) "SID",S.USERNAME,
E.SEGMENT_NAME,
T.START_TIME "Start",
rpad(T.STATUS,9) "Status",
round((t.used_ublk*8)/1024) "Size(MB)"
--T.USED_UBLK||' Blocks and '||T.USED_UREC||' Records' "Rollback
```

refer once for CFS1

some times job may complete sucessfully.

DBA_UNDO_EXTENTS group by status;

https://community.oracle.com/community/support/support-blogs/database-support-blog/blog/2015/12/10/ora-1555-do-you-know-how-to-resolve-this-issue

cker.shtml

```
column sid format 9999
column username format a15
column SQL_EXEC_START for a21
column sql_text format a50
column module format a35
column sql_text format a50
break on report
compute SUM of MB_USED on report
 SELECT a.username, a.sid, a.serial#, a.osuser,
(b.blocks*d.block_size)/1048576 MB_USED,a.sql_id,a.sql_child_number
child,c.plan_hash_value,to_char (a.sql_exec_start, 'dd-Mon-yyyy
hh24:mi:ss') sql_exec_start,c.rows_processed,a.status,
-- c.sql_text
```

```
select
ROUND(
SUM(tem
pseg_size
)/1048576
)
temp_mb
from
gv$sql_w
orkarea_a
ctive
WHERE
sid=&sid;
```

Archivelog

archive log size

Finding Busy archive process

Archive generated per month

archive log creation time

scn to archivelog file

REDO

Redo logfile info
Redo log files
standby redelog files

Redo Generated PERDAY

How full is the current redo log file?

Dropping a redolog file error

Dropping a redolog file of other instance on RAC

Adding a New Redo Logfile Group

Adding Members to an existing group

Dropping Members from a group

Dropping Logfile Group

Renaming or Relocating Logfiles

Clearing REDO LOGFILES

```
set lines 1000 pages 9999
col name for a100
select thread#, sequence#, name,
round(blocks*block_size/1024/1024) MBytes
from v$archived_log order by thread#, sequence#;


select * from v$archive_processes;
set lines 750 pages 9999
select to_char(first_time,'YYYY-MON-DD') day,
to_char(sum(decode(to_char(first_time,'HH24'),'00',1,0)),'99
') "00",
to_char(sum(decode(to_char(first_time,'HH24'),'01',1,0)),'99

 select sequence#, substr(name,1,96),creator,
to_char(first_time,'DD-MON HH24:MI'),
to_char(completion_time,'DD-MON HH24:MI')
    from v$archived_log
    where first_time > sysdate-5
   order by 1

select name,thread#, sequence#, first_time, next_time,
first_change#, next_change# from v$archived_log where
9113281811148 between first_change# and next_change#;


col group# format 999
col thread# format 999
col member format a70 wrap
col status format a10
col archived format a10
col fsize format 999 heading "Size (MB)"

select l.group#, l.thread#,
f.member,
l.archived,
l.status,
(bytes/1024/1024) fsize
from
v$log l, v$logfile f
where f.group# = l.group#
order by 1,2
/
select group#,thread#,bytes/1024/1024 from v$log;
select group#,dbid,sequence#,status from v$standby_log;
select trunc(completion_time) rundate
,count(*) logswitch
,round((sum(blocks*block_size)/1024/1024)) "REDO PER
DAY (MB)"
from v$archived_log
group by trunc(completion_time)
order by 1;
```

```
SELECT le.leseq                  "Current log sequence No",
          100*cp.cpodr_bno/le.lesiz "Percent Full",
          cp.cpodr_bno              "Current Block No",
          le.lesiz                  "Size of Log in Blocks"
     FROM x$kcccp cp, x$kccle le
    WHERE le.leseq =CP.cpodr_seq
      AND bitand(le.leflg,24) = 8
   /
```

```
SQL> ALTER DATABASE DROP LOGFILE GROUP 1;
ALTER DATABASE DROP LOGFILE GROUP 1
*
ERROR at line 1:
ORA-01624: log 1 needed for crash recovery of instance
ORA920 (thread 1)
ORA-00312: online log 1 thread 1: '<file_name>'
SQL> ALTER DATABASE DROP LOGFILE GROUP 3;
ALTER DATABASE DROP LOGFILE GROUP 3
*
ERROR at line 1:
ORA-01623: log 3 is current log for instance
UNNAMED_INSTANCE_2 (thread 2) -
cannot drop
ORA-00312: online log 3 thread 2:
'+DATA_OAM/q1oam/onlinelog/group_3.277.790574217'
```

```
alter database add logfile group 3
‘/u01/oracle/ica/log3.ora’ size 10M;
```

```
     alter database add logfile member
     ‘/u01/oracle/ica/log11.ora’ to group 1;
```

```
alter database drop logfile member
‘/u01/oracle/ica/log11.ora’;
```

```
alter database drop logfile group 3;

alter database rename file ‘/u01/oracle/ica/log1.ora’ to
‘/u02/oracle/ica/log2.ora’;
ALTER DATABASE CLEAR LOGFILE GROUP 3;
ALTER DATABASE CLEAR UNARCHIVED LOGFILE
GROUP 3;
ALTER DATABASE CLEAR UNARCHIVED LOGFILE
'/ora10gsoft/10.2.0/oradata/dosa/redo1.log';
ALTER DATABASE CLEAR UNARCHIVED LOGFILE
GROUP 1 UNRECOVERABLE DATAFILE;
```

Then kill that session.nothing will happen to DB

```
 select sequence#, name,creator, to_char(first_time,'DD-MON HH24:MI'),
to_char(completion_time,'DD-MON HH24:MI')
    from v$archived_log
    where first_time > sysdate-1
    and name is not null
   order by 1
```

```
 select * from v$logfile;
```

Easy problem to resolve. Simply perform a checkpoint on the database:

```
SQL> ALTER SYSTEM CHECKPOINT GLOBAL;

System altered.

SQL> ALTER DATABASE DROP LOGFILE GROUP 1;

Database altered.
```

```
select * from gv$thread; ( Check thread is PUBLIC or PRIVATE)
ALTER DATABASE DISABLE THREAD 2 ;
ALTER DATABASE DROP LOGFILE GROUP 3;
ALTER DATABASE ENABLE PUBLIC THREAD 2 ;
select * from gv$thread;
ALTER DATABASE
ADD LOGFILE GROUP 10
('c:\oracle\oradata\whs\redo\redo_05_01.log',
'd:\oracle\oradata\whs\redo\redo_05_02.log')
SIZE 100M;

ALTER DATABASE ADD LOGFILE MEMBER
 /DISK2/log1b.rdo' TO GROUP 1,
'/DISK2/log2b.rdo' TO GROUP 2;
```

In Disaster scenarios (Redo log file deleted From os side)clear that unarchived logfile then delete it.
select * from v$log;
Use this command to check the Redolog file status, If it is UNUSED (or) YES,CLEARING state then, we can able to drop that logfile.

In Disaster scenarios (Redo log file deleted From os side)clear that unarchived logfile then delete it.
select * from v$log;
Use this command to check the Redolog file status, If it is UNUSED (or) YES,CLEARING state then, we can able to drop that logfile.

```
  ALTER DATABASE
ADD LOGFILE MEMBER '+DATAC1'
TO GROUP 1;                                    <-- FOR RAC
```

OEM repository details

Clear OEM alerts

useful link

List Hostnames

target names

standalone db's not part of rac database

Changing DBSNMP password for targets

extracting port/sid/dbname & creating tnsnames.ora file

OEM sqls

emctl commands

list targets

start blackout
All blackout commands

clearing unknown status of targets

Find CLUSTERNAME from hostname

OEM All metric repository tables

```
ps -ef | grep oms
./emctl config oms -list_repos_details
```

```
select t.target_name
, t.target_type
, collection_timestamp
, message
, 'exec sysman.em_severity.delete_current_severity(''' ||
t.target_guid || ''',''' ||
metric_guid || ''',''' ||
key_value || ''')' em_severity
from sysman.mgmt_targets t
inner join
```

http://www.oracledbasupport.co.uk/querying-grid-repository-tables/

```
  FROM mgmt$target
 , mgmt$target_properties
 WHERE ( mgmt$target.target_name =
mgmt$target_properties.target_name )
 AND ( mgmt$target.target_type =
mgmt$target_properties.target_type )
 AND ( mgmt$target_properties.property_name in (
'CPUCount','DBVersion' ) )
 GROUP BY mgmt$target.host_name
```

```
select * from sysman.mgmt$db_dbninstanceinfo
```

```
select * from sysman.mgmt$db_dbninstanceinfo
where target_type = 'oracle_database' and database_name in
(select database_name from sysman.mgmt$db_dbninstanceinfo
where target_type = 'oracle_database'
having database_name not in (select database_name from
sysman.mgmt$db_dbninstanceinfo where target_type =
'rac_database')
group by database_name)
order by database_name;
```

Get the target names using

for rac -->

for non rac -->

use this for both RAC & NON RAC

http://askdba.org/weblog/2011/01/retrieving-database-sidport-information-from-grid-control-repository/

http://www.oracledbasupport.co.uk/querying-grid-repository-tables/

http://satya-dba.blogspot.com/2010/01/emctl-commands.html

```
emctl config agent listtargets
```

```
emctl start blackout test DB1:oracle_database -d 07:00
```

http://dbaclass.com/article/emcli-command-oem-12c/

```
 oracle@testhost [oemagent] cd /u01/app/oracle/agent/agent12c/bin
 ./emctl stop agent
 ./emctl clearstate agent
 ./emctl start agent
 ./emctl upload agent
```

```
col target_name for a30
col host_name for a60
col property_value for a30
SELECT a.target_name,a.host_name ,b.property_value
 FROM sysman.mgmt$target a
 , sysman.mgmt$target_properties b
 WHERE  a.target_name = b.target_name
 and a.target_type = 'rac_database'
 and b.property_name in (  'ClusterName')
 and a.host_name  like lower('%&HOST_NAME%') order by a.host_name;
```

https://blog.dbi-services.com/querying-the-oracle-management-repo

```
select database_name from
sysman.mgmt$db_dbninstanceinfo
having database_name not in (select database_name from
sysman.mgmt$db_dbninstanceinfo where target_type =
'rac_database')
group by database_name
minus
select database_name from
sysman.mgmt$db_dbninstanceinfo where
lower(database_name) = lower(instance_name)
;
```

2nd query will not return properly. May be or

```
emcli argfile Chng_DB_mon_passwd1
```

```
select target_name from
sysman.mgmt$db_dbninstanceinfo where target_type =
```

```
update_password -target_type=rac_database -target_name=d1dbname -credential_type=DBCreds -key_
```

```
target_name=db_name -credential_type=DBCreds -
key_column=DBUserName:dbsnmp -
non_key_column=DBPassword:p@ssw0rdold:m

emcli set_monitoring_credential -
target_type=rac_database -target_name=db_name -
set_name="DBCredsMonitoring" -cred_type="DBCreds" -
attributes="DBUserName:dbsnmp;DBPassword:m0nit0r;D
BRole:NORMAL"
emcli sync
```

ository/

```
_column=DBUserName:dbsnmp -non_key_column=DBPassword:p@ssw0rdold:m0nit0r
```

Troubleshootin GG ABENDED process
GG running or not

agctl location
agctl start

agctl status

agctl relocate

gg status

Get lag report

view report

to check the detail

error
no data found

start extract

check manager details
params of extract process

others

Identify the Lag from PDB (execute on target side)

View errorlog from ggsci

shell commands in ggsci

View report (Rutime statistics)

history command

execute the previous command

http://www.oracle-scn.com/approach-to-troubleshoot-an-abended-ogg-process/

```
 ps -ef | grep ./mgr

locate agctl
 /opt/oracle/product/emagent/gg/install/xag/bin/agctl start goldengate ggate_prod --node server2
cd /opt/oracle/product/emagent/gg/install/xag/bin
 ./agctl status goldengate ggate_<uat/prod/bcp>
cd /opt/oracle/product/emagent/gg/install/xag/bin/
./agctl relocate goldengate ggate_prod --node secondnode


send EWDBNAMEP1, status
 GGSCI (server1) 3> send RW12345PB1, getlag

Sending GETLAG request to REPLICAT RWDBNAME1 ...
Last record lag 1,218 seconds.

view report EXT02

info REPNAME detail
```

check the table name , add "HANDLECOLLISIONS"   & "NOHANDLECOLLISIONS"   before and after the MAP statement in parameterfile and restart the EXTRACT/REPLICAT

Once the LAG is zero , stop the replicat & remove the "HANDLECOLLISIONS"   & "NOHANDLECOLLISIONS" from parameter file. Then start the replicat

```
START EXTRACT <extract name>, DETAIL

view params mgr
view params EXTPROCESS

$GRID_HOME/bin/crsctl stat res -w "TYPE = xag.goldengate.type" -p
crsctl status res xag.ggate_uat.goldengate -f
agctl config goldengate  ggate_uat

select incoming_extract incoming_extract, round(
       (extract (second from a)
     + extract (minute from a) * 60
     + extract (hour   from a) * 60 * 60
     + extract (day    from a) * 60 * 60 * 24)/60,0) as heartbeat_lag_secs
from (
  select remote_database||'_'||incoming_extract incoming_extract
  , (systimestamp - EXTRACT(TIMEZONE_HOUR FROM SYSTIMESTAMP)/24 + EXTRACT(TIMEZONE_MINUTE FROM SYSTIMESTAMP)/60/24
  - heartbeat_timestamp) +
    (systimestamp - EXTRACT(TIMEZONE_HOUR FROM SYSTIMESTAMP)/24 + EXTRACT(TIMEZONE_MINUTE FROM SYSTIMESTAMP)/60/24
  - incoming_replicat_ts) as a
from C##GGADMIN.GG_HEARTBEAT);


VIEW GGSEVT
```

```
ggsci > sh date

Send <Process_name>, Report
View <process_name>, report

h
h 10
GGSCI (oelr5u7) 30> !
 h 29

GGSCI Command History

    2: start mgr
    3: start er *
    4: status all
    . . .

   29:  h 29
   30:  h 29

GGSCI (oelr5u7) 31> !4
status all

Program     Status      Group       Lag at Chkpt  Time Since Chkpt

MANAGER     RUNNING
EXTRACT     RUNNING     EBATCH      00:00:00      00:00:02
EXTRACT     RUNNING     ETEST       00:00:00      00:00:03
REPLICAT    RUNNING     RBATCH      00:00:00      00:00:04
REPLICAT    RUNNING     RTEST       00:00:00      00:00:05
REPLICAT    RUNNING     RTESTA      00:00:00      00:00:01
REPLICAT    RUNNING     RTESTB      00:00:00      00:00:00

GGSCI (oelr5u7) 32>
```

more debuggin in memory

SGA Dynamic allocation

Free/Used   SGA

memory utilization in SGA by shared pool

Determine PGA and Process Memory in use by Process

Total PGA used (MB)

History of pga memeory used

PGA Memory used by Each Process

Total Memory used

PGA memory usage by sql id

PGA Memery used by a particular spid

PGA Memery used by a Background processes

MAX Memory used by a person or process

Memory used by each user (pga,uga)

session memory used in the PGA

I /O Process

amount of session memory used in the shared pool for all sessions

Free memory in Shared pool

Finding statement/s which use lots of shared pool memory:

Memory issue (MOSC Note:146599.1, Diagnosing and Resolving Error ORA-04031)

From when are you getting ORA-4031 ERROR- HISTORY

troubleshoot Tool

https://coskan.wordpress.com/2007/09/14/what-i-learned-about-shared-pool-management/
http://yong321.freeshell.org/computer/SharedPoolPurging.html

```
select component,current_size/1024/1024 || 'M' from v$sga_dynamic_components;
column TOTAL_SGA for a20
column USEED for a20
column FREE for a20
select  round(sum(bytes)/1024/1024,2)||' MB' total_sga,
      round(round(sum(bytes)/1024/1024,2) - round(sum(decode(name,'free
memory',bytes,0))/1024/1024,2))||' MB' used,
      round(sum(decode(name,'free memory',bytes,0))/1024/1024,2)||' MB' free
    from v$sgastat;
select * from ( select POOL, NAME, BYTES, BYTES/1048576 as MBytes from v$sgastat
where pool='shared pool' order by BYTES desc ) where rownum <= 25;
```

```
set lines 110
col unm format a30 hea "USERNAME (SID,SERIAL#)"
col pus format 999,990.9 hea "PROC KB|USED"
col pal format 999,990.9 hea "PROC KB|MAX ALLOC"
col pgu format 99,999,990.9 hea "PGA KB|USED"
col pga format 99,999,990.9 hea "PGA KB|ALLOC"
col pgm format 99,999,990.9 hea "PGA KB|MAX MEM"

select s.username||' ('||s.sid||','||s.serial#||')' unm, round((sum(m.used)/1024),1) pus,
round((sum(m.max_allocated)/1024),1) pal, round((sum(p.pga_used_mem)/1024),1) pgu,
round((sum(p.pga_alloc_mem)/1024),1) pga, round((sum(p.pga_max_mem)/1024),1) pgm
from v$process_memory m, v$session s, v$process p
where m.serial# = p.serial# and p.pid = m.pid and p.addr=s.paddr and

select sum(pga_used_mem)/1024/1024 pga from v$process;
select min(a.begin_time),max(a.end_time),b.USED_TOTAL,b.ALLOCATED_TOTAL from
dba_hist_sysmetric_summary a,DBA_HIST_PROCESS_MEM_SUMMARY b where
begin_time between SYSDATE-3 and SYSDATE and a.snap_id=b.snap_id group by
b.USED_TOTAL,b.ALLOCATED_TOTAL order by 1 asc;
```

```
SELECT to_char(ssn.sid, '9999') || ' - ' || nvl(ssn.username, nvl(bgp.name, 'background')) ||
nvl(lower(ssn.machine), ins.host_name) "SESSION",
to_char(prc.spid, '999999999') "PID/THREAD",
to_char((se1.value/1024)/1024, '999G999G990D00') || ' MB' " CURRENT SIZE",
to_char((se2.value/1024)/1024, '999G999G990D00') || ' MB' " MAXIMUM SIZE"
FROM v$sesstat se1, v$sesstat se2, v$session ssn, v$bgprocess bgp, v$process prc,
v$instance ins, v$statname stat1, v$statname stat2
WHERE se1.statistic# = stat1.statistic# and stat1.name = 'session pga memory'
AND se2.statistic# = stat2.statistic# and stat2.name = 'session pga memory max'
AND se1.sid = ssn.sid
AND se2.sid = ssn.sid
AND se1.sid=114
AND ssn.paddr = bgp.paddr (+)
AND ssn.paddr = prc.addr (+);
```

```
select (sga+pga)/1024/1024 as "sga_pga"
     from
     (select sum(value) sga from v$sga),
     (select sum(pga_used_mem) pga from v$process);
col percent head '%' for 99990.99
col star for A10 head ''

accept seconds prompt "Last Seconds [60] : " default 60;
accept top prompt "Top  Rows    [10] : " default 10;

select SQL_ID,round(PGA_MB,1) PGA_MB,percent,rpad('*',percent*10/100,'*') star
from
(
select SQL_ID,sum(DELTA_PGA_MB) PGA_MB ,(ratio_to_report(sum(DELTA_PGA_MB))
over ())*100 percent,rank() over(order by sum(DELTA_PGA_MB) desc) rank
from
(
SELECT spid, program,
          pga_max_mem      max,
          pga_alloc_mem    alloc,
          pga_used_mem     used,
          pga_freeable_mem free
      FROM V$PROCESS
     WHERE spid = 7735;

select
PID,SPID,serial#,USERNAME,PROGRAM,PGA_USED_MEM,PGA_ALLOC_MEM,PGA_F
REEABLE_MEM,PGA_MAX_MEM from v$process order by PGA_USED_MEM desc;

set linesize 750
set pagesize 250
column box format a20
column username format a7
column program format a20
column os_user format a20

select s.sid, p.pid, p.spid,substr(s.machine,1,20) box,s.logon_time logon_date,to_char
(s.logon_time, 'hh24:mi:ss') logon_time,
substr(s.username,1,7) username,
substr(s.osuser,1,20) os_user,
substr(s.program,1,35) program from v$session s, v$process p where s.paddr = p.addr
and
s.sid=(select s.sid from v$session s where paddr=(select addr from v$process where
pga_used_mem=(select max(pga_used_mem) from v$process)));
```

```
select s.osuser osuser,s.username,s.program,s.serial# serial,se.sid,n.name,
       max(se.value) maxmem
from v$sesstat se,
     v$statname n
,v$session s
where n.statistic# = se.statistic#
and   n.name in ('session pga memory','session pga memory max',
                'session uga memory','session uga memory max')
and s.sid=se.sid
group by n.name,se.sid,s.osuser,s.username,s.program,s.serial#
order by 2
;
```

```
set pages 999
column pga_size format 999,999,999
select
    1048576+a.value+b.value   pga_size
from
   v$parameter a,
   v$parameter b
where
   a.name = 'sort_area_size'
and
   b.name = 'hash_area_size';

COL ts FORMAT a10 HEADING "Tablespace";
COL reads FORMAT 9999900;
COL writes FORMAT 9999900;
COL br FORMAT 9999900 HEADING "BlksRead";
COL bw FORMAT 9999900 HEADING "BlksWrite";
COL rtime FORMAT 9999900;
COL wtime FORMAT 9999900;
SELECT ts.name AS ts, fs.phyrds "Reads", fs.phywrts "Writes"
, Fs.phyblkrd AS br, fs.phyblkwrt AS bw
, Fs.readtim "RTime", fs.writetim "WTime"
FROM v$tablespace ts, v$datafile df, v$filestat fs
WHERE ts.ts# = df.ts# AND df.file# = fs.file#
UNION
SELECT ts.name AS ts, ts.phyrds "Reads", ts.phywrts "Writes"
, Ts.phyblkrd AS br, ts.phyblkwrt AS bw
, Ts.readtim "RTime", ts.writetim "WTime"
FROM v$tablespace ts, v$tempfile tf, v$tempstat ts
WHERE ts.ts# = tf.ts# AND tf.file# = ts.file# ORDER BY 1;

select SUM(value)/1024/1024||' Mb' "Total session memory"
     from v$sesstat, v$statname
     where name = 'session uga memory'
     and v$sesstat.statistic# = v$statname.statistic#;
```

```
SELECT * FROM v$sgastat
WHERE name = 'free memory';

set lines 1000 pages 9999
col stmt for a150
SELECT substr(sql_text,1,150) "Stmt", count(*),
                sum(sharable_mem)    "Mem",
                sum(users_opening)   "Open",
                sum(executions)      "Exec"
          FROM v$sql
         GROUP BY substr(sql_text,1,150)
        HAVING sum(sharable_mem) > 20000
        order by sum(sharable_mem);
```

http://www.dbas-oracle.com/2013/05/5-Easy-Step-to-Solve-ORA-04031-with-Oracle-Support-Provided-Tool.html

For 10g:

```
set pages 999
set lines 130
col component for a25 head "Component"
col status format a10 head "Status"
col initial_size for 999,999,999,999 head "Initial"
col parameter for a25 heading "Parameter"
col final_size for 999,999,999,999 head "Final"
col changed head "Changed At"

select component, parameter, initial_size, final_size, status,
to_char(end_time ,'mm/dd/yyyy hh24:mi:ss') changed
from gv$sga_resize_ops
order by 6
/
```

Metalink ID 559339.1

```
SELECT to_char(ssn.sid, '9999') || ' - ' || nvl(ssn.username,
nvl(bgp.name, 'background')) ||
nvl(lower(ssn.machine), ins.host_name) "SESSION",
to_char(prc.spid, '999999999') "PID/THREAD",
to_char((se1.value/1024)/1024, '999G999G990D00') || ' MB' "
CURRENT SIZE",
to_char((se2.value/1024)/1024, '999G999G990D00') || ' MB' "
MAXIMUM SIZE"
FROM v$sesstat se1, v$sesstat se2, v$session ssn,
v$bgprocess bgp, v$process prc,
v$instance ins, v$statname stat1, v$statname stat2
WHERE se1.statistic# = stat1.statistic# and stat1.name =
'session pga memory'
AND se2.statistic# = stat2.statistic# and stat2.name = 'session
pga memory max'
AND se1.sid = ssn.sid
```

<--For a Single sid

https://bdrouvot.wordpress.com/2013/03/19/link-huge-pga-temp/

The output from this data dictionary query shows that every connected Oracle session will about 2.5 megabytes of RAM memory for the Oracle PGA

See all the data file i / o, and if too many files, can be rewritten for the top 10 select * (order by xx desc) where rownum <= 10 http://www.dba-oracle.com/t_x$ksmlru_x$ksmsp_shared_pool_monitoring.htm

For 11g:

```
set pages 999
set lines 130
col component for a25 head "Component"
col status format a10 head "Status"
col initial_size for 999,999,999,999 head "Initial"
col parameter for a25 heading "Parameter"
col final_size for 999,999,999,999 head "Final"
col changed head "Changed At"

select component, parameter, initial_size, final_size, status,
to_char(end_time ,'mm/dd/yyyy hh24:mi:ss') changed
from gv$memory_resize_ops
order by 6
/
```

patching post activity

pdb violations

applied patch versions

patch validation
Patch check

componenets status

invalid components error messages

invalid components after all procedure

Auto Patching Code

```
/opt/oracle/product/12.1.0.2.PSUAPR2017/OPatch
./datapatch -verbose
```

```
col CAUSE for a15
col MESSAGE for a30
select name,cause,message, STATUS from pdb_plug_in_violations;
```

```
select PATCH_ID, ACTION, ACTION_TIME, DESCRIPTION, STATUS, LOGFILE from registry$sqlpatch;
```

```
datapatch -db db_name -prereq
```

```
 select PATCH_ID,PATCH_UID,VERSION,ACTION,ACTION_TIME,STATUS,DESCRIPTION,BUNDLE_SERIES,BUNDLE_ID,BUNDLE_DATA from dba_registry_sqlpatch;
```

Database PSUs
CRS□PSUs
GI (Grid Infrastructure) PSUs
Enterprise Manager Agent PSUs

```
col COMP_ID for a25
col COMP_NAME for a60
set lines 700
select COMP_ID,COMP_NAME,VERSION,STATUS from dba_registry;
```

```
set lines 750 pages 9999
col text for a80
SELECT e.owner, e.name, TO_CHAR(e.line) || '/' || TO_CHAR(e.position) "POSITION", e.text
   FROM dba_errors e
   ORDER BY e.owner, e.name, e.sequence;
```

```
delete from obj$ where name in (select OBJECT_NAME from dba_objects where status ='INVALID' and owner ='XDB');
commit;
set serveroutput on
execute sys.dbms_regxdb.validatexdb;
select comp_name, version, status from dba_registry where comp_id='XDB';
```

```
for i in `crsstat.sh |grep .db| grep -v svc |grep -v .vip|grep -v .lsnr |grep -v .mgmt | grep -v OFFLINE | grep
"Open,STABLE"|cut -d"." -f2`

do
GRID_ENV=`ps -ef|grep pmon|awk '{print $NF}'|grep asm|cut -d_ -f3`
export GRID_ENV
. oraenv $GRID_ENV
NEW_HOME=`srvctl config database -v|grep $i|awk '{print $2}'`
export NEW_HOME
. /opt/oracle/local/bin/oraenv $i
for j in `srvctl status database -d $i | grep "running" | grep -v "not running" |awk '{print $2}' | awk 'NR==1'`
do
node=`srvctl status database -d $i |grep $j| awk '{print $NF}'`
connection="nohup `ssh -q $node "$(typeset -f Datapatch_apply); Datapatch_apply $j "` &"
done
done
```

<-- status should be SUCCESS

```
opatch lsinventory -bugs_fixed | egrep -i 'PSU|DATABASE PATCH SET UPDATE'
opatch lsinventory -bugs_fixed | grep -i 'TRACKING BUG' | grep -i 'PSU'
opatch lsinventory -bugs_fixed | grep -i 'GRID INFRASTRUCTURE PATCH SET UPDATE'
opatch lsinventory -bugs_fixed | grep -i 'ENTERPRISE MANAGER' | grep -i 'AGENT'
```

scheduled job run details

scheduled job logs

DBA JOBS

SCHEDULED JOBS

```
select
LOG_ID,LOG_DATE,OWNER,JOB_NAME,STATUS,ERROR#,REQ_START_DATE,ACTU
AL_START_DATE,RUN_DURATION,SESSION_ID,CPU_USED,ADDITIONAL_INFO from
DBA_SCHEDULER_JOB_RUN_DETAILS;

select LOG_ID,LOG_DATE,OWNER,
JOB_NAME,STATUS,USER_NAME,CLIENT_ID,GLOBAL_UID,ADDITIONAL_INFO from
DBA_SCHEDULER_JOB_LOG;

SELECT LOG_USER,SCHEMA_USER,LAST_DATE, LAST_SEC,THIS_DATE,THIS_SEC,
NEXT_DATE,
NEXT_SEC,TOTAL_TIME,BROKEN,INTERVAL,FAILURES,WHAT,INSTANCE FROM D
BA_JOBS;

SELECT OWNER,JOB_NAME,CLIENT_ID, PROGRAM_OWNER,
PROGRAM_NAME,SCHEDULE_OWNER,START_DATE
END_DATE,ENABLED,STATE,RUN_COUNT,LAST_START_DATE,LAST_RUN_DURATIO
N,
NEXT_RUN_DATE, MAX_RUN_DURATION,COMMENTS FROM
DBA_SCHEDULER_JOBS;
```

X$ Views

List the hidden parameters in the oracle database
undocumented init parameters:
find all internal tables like user$
how to find all x$views list

Hidden parameter default value

unset hidden parameter
Reading Alert log

Row Limiting Clause

v$ views

Info.sql

sp file parameter difference between two rac instances

Database started time
Database current time
DNS lookup in oracle
Service Name

Database Characterset
Finding DBNAME as a normal user
How to see oracle Enterprise edition or not

Showing Current user

My current session sid & spid
generate random password
Tablespace

General database comparision

others

Case Insensitive search in sql where condition

Active Services running

spool with pdb name

Creating Oracle Alert log table

```
ORA-00600: internal error code, arguments: [kdourp_inorder2],
[44], [0], [48], [44], [], [], []
ORA-08007: Further changes to this block by this transaction
not allowed
Doing block recovery for file 21 block 456408
SMON: Parallel transaction recovery slave got internal error
SMON: Downgrading transaction recovery to serial
```

Blocking or terminating Development tools

Flash Recover area full .Unable to archive…
Change the archive log location

Getting oracle_home from sql*plus

Database 32bit or 64bit

DBV

Service Statics/Service previously running or not

External table in oracle for excel sheet

Create and drop db link as another user

ORA-28547: connection to server failed, probable Oracle Net admin error

```
$ sqlplus
Error 6 initializing SQL*Plus
Message file sp1<lang>.msb not found
SP2-0750: You may need to set ORACLE_HOME to your Oracle software directory
Exit 1
```

Creating Big table

```
ERROR at line 1:
ORA-01565: error in identifying file
'/ora10gsoft/10.2.0/oradata/sambaar/users01.dbf'
ORA-27037: unable to obtain file status
Linux Error: 2: No such file or directory
```

```
ORA-01139: RESETLOGS option only valid after an incomplete database recovery
```

one redo log file deleted on os side.(redolog group contains only one member)

Oracle internal

Writinh own message in Alertlog

License Usage report

Profile for OS
dissassembling_the_data_block

Freezing the oracle database(If we want to suspend all i/o operations.)

Restrict the database logins (To put our database in a single user mode)

Quiescing Oracle Database

create snapshot

Tricks

Hiding user id's
Executing some script without logging as that user

Get your name & database name in sqlprompt

Connect to other user without password

column separator

ipv6 connectivity

Magical Faster secret query

case insensitive search

DBMS-insufficient privileges

put lines inside oracle datbase

time zone of the database

Database home options check

Check the option

Product/Component
Automated Storage Management
Oracle Data Mining
Database Vault
Oracle OLAP
Oracle Label Security
Oracle Partitioning
Real Application Cluster
Real Application Testing

Enabling & disabling the options

Product/Component
Oracle Data Mining
Database Vault
Oracle OLAP
Oracle Label Security
Oracle Partitioning
Real Application Testing

```
set lines 750 pages 9999
column KSPPINM format a50
column KSPPSTVL format a50
select a.ksppinm, b.ksppstvl  FROM x$ksppi a, x$ksppcv b  WHERE a.indx=b.indx;
SELECT * FROM   SYS.X$KSPPI WHERE  SUBSTR(KSPPINM,1,1) = '_';
select object_name from dba_objects where object_name like '%$' and object_name not like 'SYS%'
select distinct table_name from V$INDEXED_FIXED_COLUMN where table_name like 'X$%';

select ksppstvl from x$ksppi join x$ksppcv using (indx) where ksppinm='_high_priority_processes';

ALTER SYSTEM RESET "_some_hidden_parameter" scope = spfile;
select message_text from X$DBGALERTEXT where rownum <= 20;
SELECT val
FROM   rownum_order_test
ORDER BY val DESC
FETCH FIRST 5 ROWS ONLY;

set long 20000 longchunksize 20000 pagesize 9999 linesize 1000
col PDB_NAME for a15
col HOST_NAME for a35
select db_unique_name CDB_NAME,(select name from v$pdbs where name not like '%SEED')
PDB_NAME,open_mode,database_role,(SELECT to_char(sysdate,'DD-MON-YYYY HH24:MI:SS') FROM dual)
"Current_time_db" ,(select INSTANCE_NAME from v$instance) INSTANCE_NAME,(select HOST_NAME from
v$instance ) HOST_NAME from v$database;
select INST_ID,INSTANCE_NAME,HOST_NAME,status,logins,VERSION from gv$instance order by 1;
select con_id,NAME,OPEN_MODE,RESTRICTED,OPEN_TIME from gv$pdbs order by 1;
 col WRL_PARAMETER for a30
select INST_ID,STATUS,WALLET_TYPE,WRL_TYPE,WRL_PARAMETER,WALLET_ORDER  from
gv$encryption_wallet;

SELECT p1.name, p1.value, p2.value FROM gv$parameter p1
  JOIN gv$parameter p2 ON p1.name = p2.name
  WHERE p1.inst_id = 1
    AND p2.inst_id = 2
    AND p1.value != p2.value
    AND p1.name NOT IN ('instance_number', 'instance_name', 'local_listener');
SELECT inst_id,to_char(startup_time,'DD-MON-YYYY HH24:MI:SS') "DB Startup Time" ,host_name FROM
sys.gv_$instance order by 1;
SELECT to_char(sysdate,'DD-MON-YYYY HH24:MI:SS') "sysdate" FROM   dual;
SELECT utl_inaddr.get_host_name('68.180.206.184') from dual;
select value from v$parameter where name='service_names';

SELECT value$ FROM sys.props$ WHERE name = 'NLS_CHARACTERSET' ;
select ora_database_name from dual
select * from product_component_version;

select sys_context( 'userenv', 'current_schema' ) from dual;
```

```
select
        s.sid, p.spid, substr(s.username,1,20) username, s.terminal, p.Program
  from
       v$session s, v$process p
 where
       s.paddr = p.addr
   and
       s.sid = (select sid from v$mystat where rownum=1)
;

 select DBMS_RANDOM.string('x',10) PASSWD from dual
select name from ts$;
SELECT t1.* FROM v$session t1 WHERE NOT EXISTS
(SELECT 1 FROM V$session@dblink_prod t2 WHERE t1.sid = t2.sid and t1.serial#=t2.serial# and
t1.AUTH_TYPE_ID=t2.AUTH_TYPE_ID
and t1.APP_VERSION=t2.APP_VERSION and t1.CREATE_TIME=t2.CREATE_TIME ) )



alter session set nls_comp=linguistic;
alter session set nls_sort=BINARY_CI;
select distinct metric_name from DBA_HIST_SYSMETRIC_SUMMARY where metric_name like '%memory%';
```

else following format required for searching

```
 select distinct metric_name from DBA_HIST_SYSMETRIC_SUMMARY where metric_name like '%Memory%';
col NAME for a20
col SERVICENAME_AVAIL for a20
select
                --b.inst_id ,
                b.name , a.inst_id Inst_id_avail , a.name servicename_avail
from
                gv$active_services a , dba_services b --, gv$instance c
where
                --b.inst_id = a.inst_id(+) and
                b.name = a.name(+) and
                --(a.inst_id=c.inst_id and a.instance_name=c.instance_name) and
                (a.name not like '%XDB%' AND  a.NAME NOT LIKE '%SYS$%' and a.name not like '%DGB%') and
                (b.name not like '%XDB%' AND  b.NAME NOT LIKE '%SYS$%' and b.name not like '%DGB%')
order by 1,2
/
```

this gives proper output during spool

```
-----------------------------------------------------------------
column dbname new_value dbname print
select name dbname from v$pdbs;
column timendate new_value spooltime print
select SYS_CONTEXT('USERENV', 'DB_UNIQUE_NAME')||'_'||SYS_CONTEXT('USERENV',
'SESSION_USER')||'_'||to_char(sysdate,'dd-mon-yyyy-hh24-mi-ss') timeNdate from dual;
spool &dbname-&spooltime..log
```

```
create directory BDUMP as '/u01/app/oracle/admin/mysid/bdump';

create table
   alert_log ( msg varchar2(80) )
organization external (
select
   segment_name,
   status
from
   dba_rollback_segs
where
   tablespace_name='undotbs_corrupt'
and
   status = 'NEEDS RECOVERY';
CREATE OR REPLACE TRIGGER block_tools_from_prod
AFTER LOGON ON DATABASE
DECLARE
v_program sys.v_$session.program%TYPE;
BEGIN
SELECT program INTO v_program
FROM sys.v_$session
WHERE audsid = USERENV('SESSIONID')


SQL> SELECT * FROM V$RECOVERY_FILE_DEST;


CREATE OR REPLACE FUNCTION get_java_system_property (prop IN VARCHAR2) RETURN VARCHAR2 IS
LANGUAGE JAVA
name 'java.lang.System.getProperty(java.lang.String) return java.lang.String';
/
   length(addr)*4 || '-bits' word_length
from
   v$process
where
   ROWNUM =1;



dbv file=/usr/acct/dba/dbs/dbf/data1/smp_data.dbf blocksize=4096 feedback=100
BREAK ON SNAP_ID
select snap_id,instance_number,SERVICE_NAME,value/100000 as DBTIME from DBA_HIST_SERVICE_STAT
where STAT_NAME='DB time' order by snap_id desc,instance_number FETCH FIRST 50 ROWS ONLY;
```

```
CREATE OR REPLACE DIRECTORY costtoserve_dir AS '/work/oracle/costtoserve';
grant read,write on directory costtoserve_dir to costtoserve;
CREATE TABLE costtoserve.costtoserve
 (
    parent      VARCHAR2(10),
    host_to    VARCHAR2(30),
    child    VARCHAR2(30)
)
ORGANIZATION EXTERNAL
  (
       TYPE oracle_loader
       DEFAULT DIRECTORY costtoserve_dir
       ACCESS PARAMETERS
       (
            RECORDS DELIMITED BY NEWLINE
            badfile costtoserve_dir:'upload_costtoserv_file.bad'
            logfile costtoserve_dir:'upload_costtoserv_file.log'
            FIELDS TERMINATED BY ','
            MISSING FIELD VALUES ARE NULL
            (
                 parent,
                 host_to,
                 child
     ))
          LOCATION ('hosttoserver.csv')
)REJECT LIMIT UNLIMITED;
```

```
SQL> CREATE or replace PROCEDURE scott.create_db_link AS
BEGIN
EXECUTE IMMEDIATE 'create database link LINK1 connect to scott identified by tiger using ''testdb''';
END create_db_link;
2 3 4 5
6 /

Procedure created.

SQL> show user
USER is "SYS"
SQL> exec scott.create_db_link

PL/SQL procedure successfully completed.

SQL> select * from dba_db_links where OWNER='SCOTT';

OWNER DB_LINK USERNAME HOST CREATED
------------------------------ -------------------- ------------------------------ -------------------- ---------
SCOTT LINK1 SCOTT testdb 04-NOV-11

SQL> drop database link scott.LINK1;
drop database link scott.LINK1
*
ERROR at line 1:
ORA-02024: database link not found


SQL> CREATE PROCEDURE scott.drop_db_link AS
BEGIN
EXECUTE IMMEDIATE 'drop database link LINK1';
END drop_db_link; 2 3 4
5 /
```

In my case, I comment out one line starting PROGRAM from listener.ora, and after restarting listener. I can connect successfully.

```
SID_LIST_KOREA =
(SID_LIST =
(SID_DESC =
(GLOBAL_DBNAME=korea)
(SID_NAME = KOREA)
(ORACLE_HOME = D:\Applications\oracle\product\10.2.0)
# (PROGRAM = extproc)
)
)
```

Run this

```
$ cd $ORACLE_HOME/install
$ ./changePerm.sh
```

```
create table bigtab
as
select rownum id, a.*
  from all_objects a
 where 1=0;
```

The datafile is lost and data is not accessible.

However, the datafile should still have an open file descriptor by an oracle background process

```
RECOVER DATABASE UNTIL CANCEL
alter database open resetlogs
```

redo log file deleted

```
alter database drop logfile group 1;SQL>
alter database drop logfile group 1
*
```

Template location
----
c:\oracle\product\10.2.0\db_1\assistants\dbca\templates

OEM Scripts

\\networkdrive\k$\oracle\product\10.2.0\db_1\sysman\admin\emdrep\sql

Database Creation

```
C:\WINDOWS\system32\cmd /c call K:\oracle\product\10.2.0\db_1/bin/dbca.bat  -progress_only  -createDatabase -templateName General_Purpose.dbc -gdbName test -sid test  -sysPassword
```

```
begin
  sys.dbms_system.ksdwrt(2, 'My own message');
end;
/
```

```
select
   samp.dbid,
   fu.name,
   samp.version,
   detected_usages,
   total_samples,
    decode(to_char(last_usage_date, 'MM/DD/YYYY, HH:MI:SS'),
         NULL, 'FALSE',
         to_char(last_sample_date, 'MM/DD/YYYY, HH:MI:SS'), 'TRUE',
```

```
PATH=/usr/bin:/etc:/usr/sbin:/usr/ucb:$HOME/bin:/usr/bin/X11:/sbin:.:/usr/local/bin:${ORACLE_HOME}:${ORACLE_HOME}/bin:${ORACLE_HOME}/OPatch:/oracle/dba/bin

export PATH
```

http://www.orafaq.com/papers/dissassembling_the_data_block.pdf

```
Sql> alter system suspend;
Sql> select database_status from v$instance;

Database_status
-----------------
Sql>startup restrict;
Sql>alter system disable restricted session;
Sql> alter system enable restricted session;



ACTIVE_ST
---------
NORMAL

SQL> ALTER SYSTEM QUIESCE RESTRICTED;

System altered.

SQL> select active_state from v$instance;
EXEC DBMS_WORKLOAD_REPOSITORY.create_snapshot;



update sys.user$ set name='NEW' where user#=N and name='OLD';
```

http://www.gokhanatil.com/2011/02/syssystem-users-and-ora-01031-prior-to-oracle-9-2.html

```
set termout off
define gname=idle
column global_name new_value gname
select lower(user)||'@' ||substr(global_name,1,decode(dot,0,length(global_name),dot-1)) global_name from
(select global_name, instr(global_name,'.') dot from global_name);
set sqlprompt '&gname>'
set termout on

(connect as SYSTEM)
SQL> alter user APEX_030200 grant connect through system;
User altered.
SQL> alter user apex_030200 account unlock;
User altered.
SQL> connect system[apex_030200]/systempassword
Connected.
SQL> select user from dual;
APEX_030200

SET COLSEP '|'

connect arup/arup@[fe80::219:21ff:febb:9aa5]/D112D1

jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)
  (HOST=[fe80::219:21ff:febb:9aa5]) (PORT=1526))
  (CONNECT_DATA=(SERVICE_NAME=D112D1)))
```

```
select /*+ richs_secret_hint */ ename, job
```

```
SELECT * FROM TABLE WHERE REGEXP_LIKE (TABLE.NAME,'IgNoReCaSe','i');
```

```
GRANT CREATE JOB, MANAGE SCHEDULER, MANAGE ANY QUEUE TO USER1;
```

```
set serveroutput on
set heading off
set feedback off
select sysdate from dual;
exec dbms_output.put_line('------------------------------');
select sysdate from dual;
```

```
SELECT EXTRACT(TIMEZONE_HOUR FROM SYSTIMESTAMP)||':'||
       EXTRACT(TIMEZONE_MINUTE FROM SYSTIMESTAMP)
FROM dual;
```

http://m.blog.itpub.net/17252115/viewspace-1160554/

```
cd $ORACLE_HOME/rdbms/lib
ar -tv libknlopt.a | grep -c kkxwtp.o
ar -tv libknlopt.a | grep -c kfoff.o
ar -tv libknlopt.a | grep -c ktd.o
ar -tv libknlopt.a | grep -c kxmwsd.o
ar -tv libknlopt.a | grep -c kciwcx.o
ar -tv libknlopt.a | grep -c sllfls.o
ar -tv libknlopt.a | grep -c kprnts.o
ar -tv libknlopt.a | grep -c xsnoolap.o
ar -tv libknlopt.a | grep -c kdzof.o
ar -tv libknlopt.a | grep -c kecnr.o
ar -tv libknlopt.a | grep -c dmndm.o
ar -tv libknlopt.a | grep -c kkpoban.o
ar -tv libknlopt.a | grep -c kcsm.o
ar -tv libknlopt.a | grep -c jox.o
ar -tv libknlopt.a | grep -c kzlilbac.o
ar -tv libknlopt.a | grep -c kzvidv.o
```

Short Name
ASM
DM
DV
OLAP
OLS
PART
RAC
RAT

```
chopt enable partitioning
chopt disable partitioning
```

Option
dm
dv
olap
lbac
partitioning
rat

```
select name, value from sys.V$PARAMETER where name like '\_%' escape '\'  and ISDEFAULT='FALSE';
```

```
SELECT
a.ksppinm Param ,
```

```
create or replace view all_parameters
as
```

```
select * from database_properties where property_name like
'%CHARACTERSET';
select global_name from global_name;

SELECT sys_context('USERENV', 'HOST') FROM dual;
SELECT sys_context('USERENV', 'INSTANCE') FROM dual;
```

```
SELECT * FROM NLS_DATABASE_PAR
```

https://www.morganslibrary.org/reference this gives lot of space output during spool -WASTE

```
--------------------------------------------------------------
column dbname new_value dbname noprint
select name dbname from v$pdbs;
column timendate new_value spooltime noprint
select SYS_CONTEXT('USERENV',
'DB_UNIQUE_NAME')||'_'||SYS_CONTEXT('USERENV',
'SESSION_USER')||'_'||to_char(sysdate,'dd-mon-yyyy-hh24-mi-ss')
timeNdate from dual;
spool &dbname-&spooltime..log
```

```
column dbname new_value dbname
noprint
select name dbname from v$pdbs;
SET MARKUP HTML ON
spool C:\Oracle\login-
db\folder\&dbname..xls
select * from all_db_links;
spool off
SET MARKUP HTML OFF
```

```
SELECT utl_inaddr.get_host_address('yahoo.com') FROM dual;
```

I created the tablespace with this SQL

```
create undo tablespace undotbs2 datafile
'/u02/oracle/oradata/test/undotbs2.dbf' size 500m;
```

Alter the database to use the new UNDO tablespace.

```
alter system set undo_tablespace=undotbs2 scope=both;
```

```
alter system set log_archive_dest_1='location=/ora10gsoft/archsam
reopen';
```

The last step in resolving ORA-00257 is to change the logs for verification using:

```
SQL> alter system switch logfile;
```

```
col ORIGINATING_TIMESTAMP for a40
col MESSAGE_TEXT for a80
set linesize 500
SELECT
originating_timestamp,
```

```
SELECT
originating_timestamp,
message_text
FROM
sys.x$dbgalertext
```

```
Solaris   --> /usr/bin/isainfo -kv
Linux --> uname -m
```

```
cd $ORACLE_HOME/bin
file oracl*
```

```
   from v$tablespace t, dba_segments s
   where s.owner = 'SCOTT'
   and s.segment_name='DEPARTMENT'
   and t.name = s.tablespace_name;
```

http://www.oracle.com/technetwork/articles/saternos-tables-090560.html http://dbaoracletips.blogspot.com/2011/11/how-to-dropcreate-database-link-from.html http://www.orafaq.com/wiki/ChangePerm_sh

This is used when when we loss our all redologs.

U can connect as user called OLD. But if u query it will show only user as NEW. U can't connect as NEW also.

If you want to display your SID, your connected USER name instead of “SQL>” you’ve to change “glogin.sql” in your $ORACLE_HOME/sqlplus/admin directory as below

Revoking that privilege

ALTER USER apex_030200 REVOKE CONNECT THROUGH system;

```
 select current_timestamp from dual;
select dbtimezone from dual;
```

948061.1

General All

```
ar -tv libknlopt.a
```

kfon.o
dmwdm.o
kzvidv.o
xsyeolap.o
kzlilbac.o
kkpoban.o
kcsm.o
kecwr.o

<== 11g

==>

10g

```
make -f ins_rdbms.mk option_switch ioracle

cd $ORACLE_HOME/rdbms/lib
make -f ins_rdbms.mk part_off ioracle
```

Database Option	ON
Data Mining	dm_on
Data Mining Scoring Engine	dmse_on
Database Vault	dv_on
Label Security	lbac_on
Partitioning	part_on
Real Application Clusters	rac_on
Spatial	sdo_on
Real Application Testing	rat_on
OLAP	olap_on
Automatic Storage Management	asm_on
Context Management Text	ctx_on

RAMETERS;

/sys_context.html

OFF
dm_off
dmse_off
dv_off
lbac_off
part_off
rac_off
sdo_off
rat_off
olap_off
asm_off
ctx_off

Pwd verify function extract password

Backdoor Entry to oracle database

oracle password hack

extract dblink password

spoof OSUSER in v$session

Wallet

--ORA-28365: wallet is not open

Auditing

SP parameter modification in last 7 days

who locked my account

user audit info

audited privileges

Mask password in shell script

user creation time

CREATE statement

Profile creation time

when was my table last changed

When was the table last accessed

when was the password changed for a user

Oracle database vault

checking it is enabled or not

disable

enable

Wallet status

http://orafaq.com/node/58

```
# sqlplus -prelim / as sysdba
(or)
# sqlplus /nolog
SQL> set _prelim on
SQL> conn / as sysdba
Prelim connection established
```

http://www.soonerorlater.hu/index.khtml?article_id=513
http://www.petefinnigan.com/weblog/archives/00001103.htm

works under 11.2.0.2

```
select passwordx from sys.link$ where name='mydblink';

set serveroutput on
declare
 db_link_password varchar2(500);
begin
 db_link_password :=
'06D05F5E36F13A08FD3C5FE489EB89B094701C114FF156A92D84A5724EF5FC2BA4F25BF
BE99146C22075BEF3012D0F9DC6231FBD1A5EFBFA97DCD8FD13737243992EA16AD5A23
B7DC823346DEB4CD69FE6F20B3F15821FEFF9F44430EE40C78CAEE37DF25F25C2BEDED
1DD2A61C72351E462BF1B844B2599E5125AE0135EAF7';

 dbms_output.put_line ('Plain password: ' || utl_raw.cast_to_varchar2 ( dbms_crypto.decrypt (
substr (db_link_password, 19) , dbms_crypto.DES_CBC_PKCS5 , substr (db_link_password, 3,
16) ) ) );
end;
/
```

https://oraganism.wordpress.com/2009/10/06/spoofing-vsession-osuser/

Connect to container and check whether WALLET is OPEN or not ? If it is in OPEN state then connect to PDB and check the same

```
--ORA-28365: wallet is not open
select status from v$encryption_wallet;
```

If the wallet is closed at Contaner DB then OPEN it using below command (run in container):

```
ADMINISTER KEY MANAGEMENT SET KEYSTORE open IDENTIFIED BY abc123
container=all;
```

If the WALLET IS OPENED in container but CLOSED in PDB level, then run below at container level

```
ADMINISTER KEY MANAGEMENT SET KEYSTORE close IDENTIFIED BY  abc123
container=all;
ADMINISTER KEY MANAGEMENT SET KEYSTORE open IDENTIFIED BY abc123
container=all;
```

Now check the wallet status at Container and PBD level:

```
select status from v$encryption_wallet;
```

```
[illegible]
col time for a15
col parameter_name format a50
col old_value format a30
col new_value format a30
break on instance skip 3
select instance_number instance, snap_id, time, parameter_name, old_value, new_value from (
select a.snap_id,to_char(end_interval_time,'DD-MON-YY HH24:MI') TIME,  a.instance_number,
parameter_name, value new_value,
lag(parameter_name,1) over (partition by parameter_name, a.instance_number order by
a.snap_id) old_pname,
lag(value,1) over (partition by parameter_name, a.instance_number  order by a.snap_id)
old_value ,
decode(substr(parameter_name,1,2),'__',2,1) calc_flag
set lines 750 pages 9999
column USERNAME format a20
column OS_USERNAME format a20
column USERHOST format a40
column EXTENDED_TIMESTAMP format a60

SELECT USERNAME, OS_USERNAME, USERHOST, EXTENDED_TIMESTAMP,returncode
FROM SYS.DBA_AUDIT_SESSION WHERE returncode != 0 and username =
'&Account_Locked'
and EXTENDED_TIMESTAMP > (systimestamp-1) order by 4;
```

```
select 'standard audit', sessionid,
    proxy_sessionid, statementid, entryid, extended_timestamp, global_uid,
    username, client_id, null, os_username, userhost, os_process, terminal,
    instance_number, owner, obj_name, null, new_owner,
    new_name, action, action_name, audit_option, transactionid, returncode,
    scn, comment_text, sql_bind, sql_text,
    obj_privilege, sys_privilege, admin_option, grantee, priv_used,
    ses_actions, logoff_time, logoff_lread, logoff_pread, logoff_lwrite,
    logoff_dlock, session_cpu
  from
  dba_audit_trail;

SELECT * FROM dba_stmt_audit_opts union SELECT * FROM dba_priv_audit_opts;
```

http://www.tek-tips.com/viewthread.cfm?qid=1605767
http://www.idevelopment.info/data/Unix/Linux/LINUX_CryptCommand.shtml

```
select a.* from sys.aud$ a, dba_users b
where a.action# = 51  --corresponds to CREATE USER
 and a.OBJ$NAME = b.username;

select * From dba_audit_trail
 where action_name like '%ALTER USER%'
  or action_name like '%CREATE USER%';

select * From dba_audit_trail
 where action_name like '%ALTER PROFILE%'
  or action_name like '%CREATE PROFILE%';
```

http://blog.tanelpoder.com/2009/02/07/when-was-a-table-last-changed/

```
select INSERTS,UPDATES,DELETES,TABLE_NAME,to_char(TIMESTAMP,'DD-MM-YY
HH24:MI') from dba_tab_modifications where table_owner not like '%SYS%' order by
TIMESTAMP desc;
```

but for the above statement the table level monitoring should be enabled
http://blog.tanelpoder.com/2009/02/07/when-was-a-table-last-changed/

```
set lines 750 pages 9999
select USER#,NAME, TO_CHAR(ptime, 'DD-MON-YYYY HH24:MI')  from user$ order by 1;
```

http://oradb-srv.wlv.ac.uk/E16655_01/server.121/e17608/dvdisabl.htm

```
SELECT PARAMETER, VALUE FROM V$OPTION WHERE PARAMETER = 'Oracle Database V

EXEC DVSYS.DBMS_MACADM.DISABLE_DV;
EXEC DVSYS.DBMS_MACADM.ENABLE_DV;
SELECT VALUE FROM V$OPTION WHERE PARAMETER = 'Oracle Label Security';
EXEC LBACSYS.CONFIGURE_OLS;
EXEC LBACSYS.OLS_ENFORCEMENT.ENABLE_OLS;
```

```
 set lines 750 pages 9999
 col  WRL_PARAMETER for a50
 select INST_ID,WRL_TYPE,WRL_PARAMETER,STATUS,WALLET_TYPE from
gv$encryption_wallet;
```

https://taliphakanozturken.wordpress.com/2012/06/12/the-backdoor-entry-to-oracle-database/

http://murty4all.blogspot.ch/2013/11/decrypting-oracle-database-db-link_29.html

```
else
administer key management set keystore close;
```

https://blog.dbi-services.com/encryption-in-oracle-public-cloud/

```
ORA-01017: invalid username/password; logon denied
ORA-28000: the account is locked
```

http://blog.tanelpoder.com/2009/02/07/when-was-a-table-last-changed/#more-198

ault';

Oracle Label security must be enabled before you can use Database Vault. If it is not enabled,
restart required

then this query returns FALSE.

tools

oswatcher gui

Notepad++ to add "'," at the end of each table in notepad++

How to find NON-ASCII characters in notepad ++
asqlmon for visual explain plan in putty

putty autologin script

set sql prompt

simulate ORA-600 error

getting input
reset the input value

Burning the CPU 100% usage from db level

Diag collection

TFA collection for database

http://www.oraclerealworld.com/best-oracle-performance-tools/
http://ba6.us/node/177
http://windows.podnova.com/trends/oracle_performance_trace.html

```
server123:$java -jar -Xmx1024m oswg.jar -is /work/oracle/server123_OSW/osw/archive
```

ctrl + H, select "regular expression " in search mode ,
find with   [\r]+
replace with   ',
then replace all
incase if you need all the tables in one line use [\r\n]+
[^\x00-\x7F]+
Tick off 'Search Mode = Regular expression', and click --> Find Next
http://blog.tanelpoder.com/2013/03/17/asqlmon-sql-sql-monitoring-like-execution-plan-line-level-drilldow

```
Dim UserName
Dim Passwrd
ServerName = InputBox("Please Enter Your Servername:")
Passwrd  = InputBox("Please Enter Your RSA TOKEN:")
If ServerName ="" Then
  Wscript.Quit
Else
 Set shell = WScript.CreateObject("WScript.Shell")
 pcmd = "C:\Program Files (x86)\Putty\putty.exe "&Servername & " -pw firstpwd"&Passwrd
Set exec = shell.Exec(pcmd)
```

```
set sqlprompt "_user '@' _connect_identifier >"
```

```
execute dbms_system.ksdwrt(2,'ORA-600: test');
```

```
accept sid    default '' -
  prompt 'Please provide the sid: '
undefine sid
```

```
DECLARE
L_n NUMBER;
BEGIN
WHILE (TRUE)
LOOP
L_ n:= dbms_random.random();
END LOOP;
END;
/
```

Oracle

Root

run as root

```
./tfactl diagcollect -database mydb -from "Nov/16/2017 21:00:00" -to "Nov/17/2017 02:00:00"
```

start of line
----
ctrl + H, select "regular expression " in search mode ,
find with ^
replace with '
END OF LINE
----
ctrl + H, select "regular expression " in search mode ,
find with $
replace with ',

place ur mouse at the start of the first line in ur list.
Press Alt + "C"
A window gets opened. Enter the text that u want to add under "Text to Insert" and press OK.
Apply the same procedure for adding text at the end of all the lines.

https://stackoverflow.com/questions/13806355/add-text-after-every-line-multiple-entries

wn-into-sql-response-time/

https://blog.dbi-services.com/oracle-11g-instance-caging-limit-database-cpu-consumption/

```
$GI_HOME/bin/diagcollection.pl --collect --chmos --incidenttime 11/04/201702:00:00 --incidentduration 02:00
```

This will trim and collect all key diagnostics for the past 12 hours.

```
    tfactl diagcollect
```

To collect for a specific date use:

```
      tfactl diagcollect -for "yyyy-mm-dd"
```

```
tfactl diagcollect -from "yyyy-mm-dd hh:mm:ss" -to "yyyy-mm-dd hh:mm:ss"
```

```
(Or)   $TFA_HOME/bin/tfactl diagcollect -srdc dbperf
```

TFA Collector - TFA with Database Support T

ools Bundle (Doc ID 1513912.1)

Oracle Hacking

Tracing User session

Monitoring scripts

Oracle University Books

RAC Performance issue

RAC

DB Parameter to Oracle Compile-Time Constant Mappin

secret functions of oracle

sql query writing Best practices

check this url before tuning database

forcing parallel in sql without code change

All important scripts (Performance Tuning)

optimizer Magic

Optimizer bug fix control

A tour of AWR Tables

Tanel pader's explain plan scripts

Best Url for sql_id performance scripts

Sql id Perf.sql one script

10046 trace analysis

http://theinterw3bs.com/wiki/index.php?title=Hacking_Oracle

http://www.scribd.com/doc/2966777/oracle-cheat-SQLPlus-Commands

http://www.oracle-base.com/articles/10g/SQLTrace10046TrcsessAndTkprof10g.php

http://www.blacksheepnetworks.com/security/resources/www.think-forward.com/sqltips.htm

http://www.colestock.com/blogs/labels/Monitoring.html

http://allr1.blackapplehost.com/
http://goodies.dale-emmons.com/Documentation/books/Oracle/
http://search.4shared.com/q/10/oracle%2011g

node-rac

http://www.morganslibrary.com/reference/rac.html

http://yong321.freeshell.org/oranotes/DbParam_CompTimeConst_Map.txt

http://www.perfvision.com/papers.php

http://beginner-sql-tutorial.com/sql-query-tuning.htm
http://sql-tuning.com/oracle-sql-performance-tuning/
https://gavinsoorma.com/2012/11/ash-and-awr-performance-tuning-scripts/

http://dbaclass.com/article/change-the-execution-plan-without-changing-the-sql-query/

http://guyharrison.squarespace.com/opsgsamples/

http://optimizermagic.blogspot.com/

https://coskan.wordpress.com/2011/02/14/plan-stability-through-upgrade-why-is-my-plan-changed-bugfixes-1/

http://www.nocoug.org/download/2008-08/a-tour-of-the-awr-tables.nocoug-Aug-21-2008.abercrombie.html#script-a

https://blog.tanelpoder.com/2009/05/26/scripts-for-showing-execution-plans-via-plain-sql-and-also-in-oracle-9i/

http://www.williamrobertson.net/documents/index-code.shtml

https://mjsoracleblog.wordpress.com/2012/12/31/sql_sql_id-sql-1-7/

10046_events.pl

Super

ıas-exact

Scenarios		
TYPE	LEVELS	OPTIONS
RMAN Clone	Same server	Same diskgroup
		Different diskgroup
	Different server	Same diskgroup
		Different diskgroup
RMAN Restore	Same server	Same diskgroup
		Different diskgroup
	Different server	Same diskgroup
		Different diskgroup
EXP	Table Level	Single Table
		Multiple Table
	Schema Level	Single Schema
		Multiple Schema
	Database	FULL DB
IMP	Table Level	Single Table
		Multiple Table
	Schema Level	Single Schema
		Multiple Schema
	Database	FULL DB
EXPDP	Table Level	Single Table
		Multiple Table

	Schema Level	Single Schema
		Multiple Schema
	Database	FULL DB
IMPDP	Table Level	Single Table
		Multiple Table
	Schema Level	Single Schema
		Multiple Schema
	Database	FULL DB
	Network Direct IMPORT	

Features

scp algorithm

exclude Table

Exclude Common

Remap Schema

Remap Tablespace

Remap Table

TABLE_EXISTS_ACTION

12c

expdp in Background

check free space on importing tablespace

Script	

cat > expdp_table_refresh.par <<EOF
userid='/ as sysdba'
directory=DATA_PUMP_DIR
dumpfile=TABLE1_%U.dmp
logfile=TABLE1.log
tables=SCHEMA.TABLE1
parallel=8
EOF
	nohup expdp parfile=testfile.par &

cat > expdp_table_refresh.par <<EOF
userid='/ as sysdba'
directory=DATA_PUMP_DIR
dumpfile=TABLE1_%U.dmp
logfile=TABLE1.log
tables=SCHEMA1.TABLE1,SCHEMA2.TABLE2
parallel=8
EOF	

```
cat > expdp_schema_refresh.par <<EOF
userid='/ as sysdba'
directory=DATA_PUMP_DIR
dumpfile=SCHEMA_NOBIGTABLES_%U.dmp
logfile=SCHEMA_NOBIGTABLES.log
schemas=SAMPLE_SCHEMA
exclude=table:"IN('TABLE1','TABLE2','TABLE3')"
parallel=8
EOF
```

```
cat > expdp_schema_refresh.par <<EOF
userid='/ as sysdba'
directory=DATA_PUMP_DIR
dumpfile=SCHEMA_NOBIGTABLES_%U.dmp
logfile=SCHEMA_NOBIGTABLES.log
schemas=SAMPLE_SCHEMA1,SAMPLE_SCHEMA2
exclude=table:"IN('TABLE1','TABLE2','TABLE3')"
parallel=8
EOF
```

```
cat > expdp_schema_refresh.par <<EOF
userid='/ as sysdba'
directory=DATA_PUMP_DIR
dumpfile=SCHEMA_NOBIGTABLES_%U.dmp
logfile=SCHEMA_NOBIGTABLES.log
FULL=Y
parallel=8
EOF
```

```
scp -c arcfour  -r
myuserid@sourceserver:/explocation/dumpfile.dmp  .
```

```
exclude=table:"IN('TABLE1','TABLE2','TABLE3')"
exclude=TABLE:"LIKE 'EXAM%'"
exclude=TABLES:">'F'"
```

http://www.

```
exclude=SEQUENCE,PROCEDURES,INDEXES,TABLES:"IN ('EMP1','DEPT')"
```

```
REMAP_SCHEMA=SCHEMA1:DUPSCHEMA2
TABLE_EXISTS_ACTION=SKIP
```

```
SELECT table_name, tablespace_name FROM dba_tables
WHERE owner='HR';
 REMAP_TABLESPACE =
USERS:EXP_TBS1,USERS2:EXP_TBS2
```

Same Schema

```
remap_table=emp:emp_bkup
```

different schema

```
REMAP_SCHEMA=SCHEMA1:DUPSCHEMA2
remap_table=emp:emp_bkup
```

APPEND, REPLACE, [SKIP] and TRUNCATE

```
TRANSFORM=DISABLE_ARCHIVE_LOGGING:Y
```

```
nohup expdp parfile=testfile.par &
tail -200f nohup.out
```

```
SELECT tablespace_name,
SUM (bytes) / (1024 * 1024) "FREE(MB)"
FROM dba_free_space where
tablespace_name=UPPER('&TABLESPACE_NAME')
GROUP BY tablespace_name;
```

.acehints.com/2012/11/data-pump-impdp-remaptable-option-to.html

Issue
ADR Different Methods to Create IPS Package (Doc ID)
Quick Steps for ADRCI
ORA-04031: troubleshoot Tool
XDB Component has INVALID
SRDC - How to Collect Standard Information for a SQL Performance Problem
TFA Collector - TFA with Database Support Tools Bundle

ID
411.1
443529.1
559339.1
1552438.1
1594386.1
1513912.1