Oracle DBA Daily Routine Checklist and Daily Check Scripts

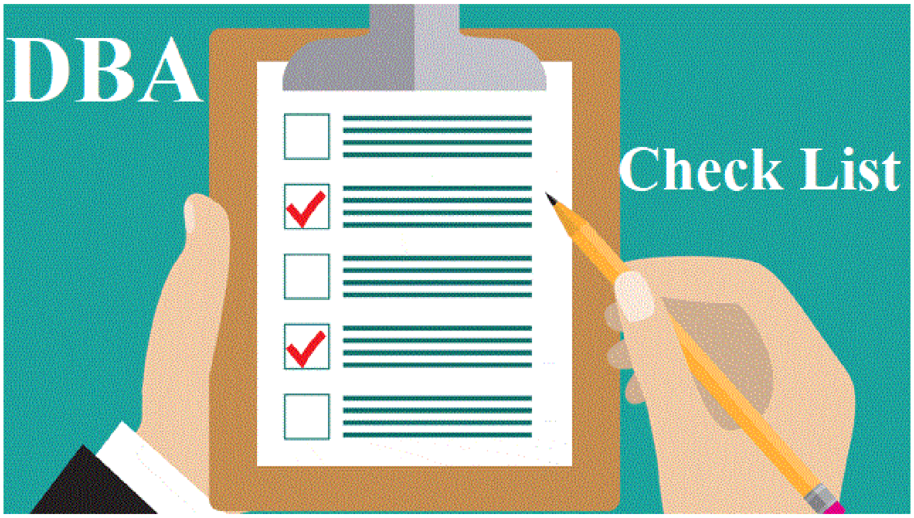
Mehmet Salih Deveci March 10, 2020

Hi,

I will explain what are the Oracle DBA Daily/Weekly/Monthly or Quarterly Routine Checklist and Tasks in this article.

Oracle DBA Daily Checklist

There are several routine checklist and tasks to do in Oracle database by DBA (Database Administrator). This checklist and tasks are as follows.



All scripts are valid for Single and RAC Database and Exadata.

If you don't use RAC, then you can ignore scripts for other Nodes and SCAN Listener etc..

Oracle DBA Daily Scripts

1- Oracle instance(s) are running or not.

Use the following script to check Oracle Instance Processes

If RAC is used, then check all Instances of database.

Linux:

Check SMON or PMON process

```
[oracle@msddbadm01 ~]$ ps -ef | grep smon

or

[oracle@msddbadm01 ~]$ ps -ef | grep pmon



Check all Instance of any database as follows.
[oracle@msddbadm01 ~]$ srvctl status database -d <DB_NAME>




[oracle@msddbadm01 ~]$ ps -ef | grep smon
oracle 140550 1 0 2019 ? 00:01:49 asm_smon_+ASM1
root 147726 1 1 2019 ? 1-04:06:08 /u01/app/12.1.0.2/grid/bin/osysmond.bin
oracle 156342 1 0 2019 ? 00:13:57 ora_smon_msd1
oracle 248609 246524 0 12:02 pts/0 00:00:00 grep smon
[oracle@msddbadm01 ~]$

[oracle@msddbadm01 ~]$ srvctl status database -d msd
Instance msd1 is running on node msddbadm01
Instance msd2 is running on node msddbadm02
[oracle@msddbadm01 ~]$
```

2- Local (and SCAN) listeners are running or not.

Use the following script to check Listeners

```
[oracle@msddbadm01 ~]$ lsnrctl status

[oracle@msddbadm01 ~]$ srvctl status scan_listener
```

```
[oracle@msddbadm01 ~]$ lsnrctl status

LSNRCTL for Linux: Version 18.0.0.0.0 - Production on 10-MAR-2020 14:42:31

Copyright (c) 1991, 2019, Oracle.  All rights reserved.
```

```
Connecting to (ADDRESS=(PROTOCOL=tcp)(HOST=)(PORT=1521))
STATUS of the LISTENER
------------------------
Alias                     LISTENER
Version                   TNSLSNR for Linux: Version 18.0.0.0.0 - Production
Start Date                04-JAN-2020 17:36:35
Uptime                    65 days 21 hr. 5 min. 55 sec
Trace Level               off
Security                  ON: Local OS Authentication
SNMP                      OFF
Listener Parameter File   /u01/grid/network/admin/listener.ora
Listener Log File         /u01/app/grid/diag/tnslsnr/msdidb01/listener/alert/log.xml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=LISTENER)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=192.168.63.34)(PORT=1521)))
Services Summary...
Service "+APX" has 1 instance(s).
  Instance "+APX1", status READY, has 1 handler(s) for this service...
Service "+ASM" has 1 instance(s).
  Instance "+ASM1", status READY, has 1 handler(s) for this service...
Service "+ASM_DATA" has 1 instance(s).
  Instance "+ASM1", status READY, has 1 handler(s) for this service...
Service "+ASM_RECO" has 1 instance(s).
  Instance "+ASM1", status READY, has 1 handler(s) for this service...
Service "MSDB" has 1 instance(s).
  Instance "MSDB1", status READY, has 1 handler(s) for this service...
Service "MSDBXDB" has 1 instance(s).
  Instance "MSDB1", status READY, has 1 handler(s) for this service...
Service "SYS$GGS_ADMIN.OGGQ$RCCB01.MSDB" has 1 instance(s).
  Instance "MSDB1", status READY, has 1 handler(s) for this service...
Service "SYS$GGS_ADMIN.OGGQ$RCCB02.MSDB" has 1 instance(s).
  Instance "MSDB1", status READY, has 1 handler(s) for this service...
The command completed successfully
[oracle@msddbadm01 ~]$
[oracle@msddbadm01 ~]$

[oracle@msddbadm01 ~]$
[oracle@msddbadm01 ~]$ srvctl status scan_listener
SCAN Listener LISTENER_SCAN1 is enabled
SCAN listener LISTENER_SCAN1 is running on node msdidb03
SCAN Listener LISTENER_SCAN2 is enabled
SCAN listener LISTENER_SCAN2 is running on node msdidb01
SCAN Listener LISTENER_SCAN3 is enabled
SCAN listener LISTENER_SCAN3 is running on node msdidb06
[oracle@msddbadm01 ~]$
```

Oracle DBA Daily Routine Checklist

3- Check the Server Storage or Disk of Oracle database .

Check if ASM and File system disks size are enough or not.

Use the following script to check Filesystem disks.

Check Filesystem disks.

```
[oracle@msddbadm01 ~]$ df -h

Unix

[oracle@msddbadm01 ~]$ df -g



[oracle@msddbadm01 ~]$ df -h
Filesystem            Size  Used Avail Use% Mounted on
/dev/mapper/VGExaDb-LVDbSys1
                       30G   25G  3.8G  87% /
tmpfs                 755G  6.5G  749G   1% /dev/shm
/dev/sda1             488M   28M  425M   7% /boot
/dev/mapper/VGExaDb-LVDbOra1
                      197G  109G   79G  59% /u01
/dev/sda2             254M   24M  231M  10% /boot/efi
/dev/mapper/VGExaDb-openvlv
                       15G   38M   14G   1% /usr/openv
/dev/asm/acfs_vol1-371
                      9.8T  4.8T  5.0T  49% /goldengate
[oracle@msddbadm01 ~]$
```

Check Oracle ASM diskgroups.

Use the following script to check Diskgroup size.

```
[grid@msdidb01 ~]$ asmcmd lsdg
```

```
[grid@msdidb01 ~]$ asmcmd
ASMCMD> lsdg
State    Type  Rebal  Sector  Logical_Sector  Block       AU    Total_MB    Free_MB  Req_mir_free_MB  Usable_file_MB  Offline_disks
MOUNTED  HIGH  Y         512             512   4096  4194304  1056964608  214081832         14680064        66467256              2
MOUNTED  HIGH  N         512             512   4096  4194304   134507520  101014692          1868160        33048844              0
ASMCMD>
```

4- Check the Tablespaces for objects to extend if required.

Use the following script to check Tablespace usage

```
set pagesize 1000 linesize 180
tti 'Tablespace Usage Status'
col "TOTAL(MB)" for 99,999,999.999
col "USAGE(MB)" for 99,999,999.999
col "FREE(MB)" for 99,999,999.999
col "EXTENSIBLE(MB)" for 99,999,999.999
col "FREE PCT %" for 999.99
col "USED PCT OF MAX %" for 999.99
col "NOTO" for 9999
col "OTO" for 999
select d.tablespace_name "NAME",
d.contents "TYPE",
nvl(a.bytes /1024/1024,0) "TOTAL(MB)",
nvl(a.bytes - nvl(f.bytes,0),0)/1024/1024 "USAGE(MB)",
nvl(f.bytes,0)/1024/1024 "FREE(MB)",
nvl((a.bytes - nvl(f.bytes,0))/a.bytes * 100,0) "FREE PCT %",
nvl(a.ARTACAK,0)/1024/1024 "EXTENSIBLE(MB)",
nvl((a.bytes - nvl(f.bytes,0))/ (a.bytes + nvl(a.ARTACAK,0)) * 100,0) "USED PCT OF MAX %",
a.NOTO, a.OTO
from sys.dba_tablespaces d,
(select tablespace_name, sum(bytes) bytes,
sum(decode(autoextensible,'YES',MAXbytes - bytes,0 )) ARTACAK,
count(decode(autoextensible,'NO',0)) NOTO,
count(decode(autoextensible,'YES',0)) OTO
from dba_data_files
group by tablespace_name) a,
(select tablespace_name, sum(bytes) bytes
from dba_free_space
group by tablespace_name) f
where d.tablespace_name = a.tablespace_name(+)
and d.tablespace_name = f.tablespace_name(+)
and NOT (d.extent_management like 'LOCAL'and d.contents like 'TEMPORARY')
UNION ALL
select d.tablespace_name "NAME",
d.contents "TYPE",
nvl(a.bytes /1024/1024,0) "TOTAL(MB)",
nvl(t.bytes,0)/1024/1024 "USAGE(MB)",
nvl(a.bytes - nvl(t.bytes,0),0)/1024/1024 "FREE(MB)",
nvl(t.bytes/a.bytes * 100,0) "FREE PCT %",
nvl(a.ARTACAK,0)/1024/1024 "EXTENSIBLE(MB)",
nvl(t.bytes/(a.bytes + nvl(a.ARTACAK,0)) * 100,0) "USED PCT OF MAX %", a.NOTO, a.OTO
from sys.dba_tablespaces d,
(select tablespace_name, sum(bytes) bytes,
sum(decode(autoextensible,'YES',MAXbytes - bytes,0 )) ARTACAK,
count(decode(autoextensible,'NO',0)) NOTO,
count(decode(autoextensible,'YES',0)) OTO
from dba_temp_files
group by tablespace_name) a,
(select tablespace_name, sum(bytes_used) bytes
from v$temp_extent_pool
group by tablespace_name) t
where d.tablespace_name = a.tablespace_name(+)
and d.tablespace_name = t.tablespace_name(+)
and d.extent_management like 'LOCAL'
and d.contents like 'TEMPORARY%'
order by 3 desc;
exit;
```

Result of script is as follows.

```
SQL> set pagesize 1000 linesize 180
SQL> tti 'Tablespace Usage Status'
SQL> col "TOTAL(MB)" for 99,999,999.999
SQL> col "USAGE(MB)" for 99,999,999.999
SQL> col "FREE(MB)" for 99,999,999.999
SQL> col "EXTENSIBLE(MB)" for 99,999,999.999
SQL> col "FREE PCT %" for 999.99
SQL> col "USED PCT OF MAX %" for 999.99
SQL> col "NOTO" for 9999
SQL> col "OTO" for 999
SQL> select d.tablespace_name "NAME",
2 d.contents "TYPE",
3 nvl(a.bytes /1024/1024,0) "TOTAL(MB)",
4 nvl(a.bytes - nvl(f.bytes,0),0)/1024/1024 "USAGE(MB)",
5 nvl(f.bytes,0)/1024/1024 "FREE(MB)",
6 nvl((a.bytes - nvl(f.bytes,0))/a.bytes * 100,0) "FREE PCT %",
7 nvl(a.ARTACAK,0)/1024/1024 "EXTENSIBLE(MB)",
8 nvl((a.bytes - nvl(f.bytes,0))/ (a.bytes + nvl(a.ARTACAK,0)) * 100,0) "USED PCT OF MAX %",
9 a.NOTO, a.OTO
10 from sys.dba_tablespaces d,
11 (select tablespace_name, sum(bytes) bytes,
12 sum(decode(autoextensible,'YES',MAXbytes - bytes,0 )) ARTACAK,
13 count(decode(autoextensible,'NO',0)) NOTO,
14 count(decode(autoextensible,'YES',0)) OTO
15 from dba_data_files
16 group by tablespace_name) a,
17 (select tablespace_name, sum(bytes) bytes
18 from dba_free_space
19 group by tablespace_name) f
20 where d.tablespace_name = a.tablespace_name(+)
21 and d.tablespace_name = f.tablespace_name(+)
22 and NOT (d.extent_management like 'LOCAL'and d.contents like 'TEMPORARY')
23 UNION ALL
24 select d.tablespace_name "NAME",
25 d.contents "TYPE",
26 nvl(a.bytes /1024/1024,0) "TOTAL(MB)",
27 nvl(t.bytes,0)/1024/1024 "USAGE(MB)",
28 nvl(a.bytes - nvl(t.bytes,0),0)/1024/1024 "FREE(MB)",
29 nvl(t.bytes/a.bytes * 100,0) "FREE PCT %",
30 nvl(a.ARTACAK,0)/1024/1024 "EXTENSIBLE(MB)",
31 nvl(t.bytes/(a.bytes + nvl(a.ARTACAK,0)) * 100,0) "USED PCT OF MAX %", a.NOTO, a.OTO
32 from sys.dba_tablespaces d,
33 (select tablespace_name, sum(bytes) bytes,
34 sum(decode(autoextensible,'YES',MAXbytes - bytes,0 )) ARTACAK,
35 count(decode(autoextensible,'NO',0)) NOTO,
36 count(decode(autoextensible,'YES',0)) OTO
37 from dba_temp_files
38 group by tablespace_name) a,
39 (select tablespace_name, sum(bytes_used) bytes
40 from v$temp_extent_pool
41 group by tablespace_name) t
42 where d.tablespace_name = a.tablespace_name(+)
43 and d.tablespace_name = t.tablespace_name(+)
44 and d.extent_management like 'LOCAL'
45 and d.contents like 'TEMPORARY%'
46 order by 3 desc;
exit;


Tue Mar 10 page 1
```

```
                                                                    Tablespace Usage Status

NAME                           TYPE                        TOTAL(MB)       USAGE(MB)        FREE(MB) FREE PCT %  EXTENSIBLE(MB) USED
------------------------------ --------------------- --------------- --------------- --------------- ---------- --------------- ----
USERS                          PERMANENT                 625,384.391     621,953.953       3,430.438      99.45      83,223.281
MSD_TBS                        PERMANENT                 550,324.313     277,077.188     273,247.125      50.35  33,004,107.664
DEFAULT_TBS1                   PERMANENT                 410,624.000       4,623.438     406,000.563       1.13  31,046,656.000
```

```
TEMP                           TEMPORARY               393,214.828          7.000     393,207.828        .00            .984
DEFAULTTBS                     PERMANENT               278,171.938     98,217.375     179,954.563      35.31  33,276,260.039
UNDOTBS1                       UNDO                     22,180.000        314.063      21,865.938       1.42     141,659.922
SYSAUX                         PERMANENT                13,670.000     12,562.313       1,107.688      91.90      19,097.984
SYSTEM                         PERMANENT                 7,880.000      7,113.375         766.625      90.27      57,655.969

8 rows selected.

SQL>
```

5- Check the Recovery Size Area

Use the following script to check Recovery Size Area

```
set pagesize 1000 line 200
col "db_recovery_file_dest" for a32;
col size_m for 999,999,999;
col used_m for 999,999,999;
col pct_used for 999;
select name "db_recovery_file_dest",ceil(space_limit/1024/1024) TOTAL_MB, ceil( space_used /1024/1024) USED_MB,
decode( nvl(space_used, 0),0,0,ceil(( space_used /space_limit) * 100)) PERCENTAGE(%)
from v$recovery_file_dest
order by 1;
exit;
```

```
SQL> set pagesize 1000 line 200
SQL> col "db_recovery_file_dest" for a32;
SQL> col size_m   for 999,999,999;
SQL> col used_m   for 999,999,999;
SQL> col pct_used for 999;
SQL> select name "db_recovery_file_dest",ceil(space_limit/1024/1024) TOTAL_MB, ceil( space_used /1024/1024) USED_MB,
  2  decode( nvl(space_used, 0),0,0,ceil(( space_used /space_limit) * 100)) PERCENTAGE
  3  from v$recovery_file_dest
  4  order by 1;
exit;
db_recovery_file_dest              TOTAL_MB     USED_MB      PERCENTAGE(%)
-------------------------------- ---------- ---------- ----------
+DATAC1                            2048000         79          1

SQL>
```

```
SQL>
```

6-Check the alert log if a vital error exists or not (Corruption)

Use the following script to check alertlog

```
set linesize 150
set pagesize 150


SELECT SUBSTR (MESSAGE_TEXT, 1, 300) MESSAGE_TEXT, COUNT (*) cnt
FROM X$DBGALERTEXT
WHERE (MESSAGE_TEXT LIKE '%ORA-%' OR UPPER (MESSAGE_TEXT) LIKE '%ERROR%')
AND CAST (ORIGINATING_TIMESTAMP AS DATE) > SYSDATE - 1
GROUP BY SUBSTR (MESSAGE_TEXT, 1, 300);

exit
```

```
SQL> SELECT SUBSTR (MESSAGE_TEXT, 1, 300) MESSAGE_TEXT, COUNT (*) cnt
2 FROM X$DBGALERTEXT
3 WHERE (MESSAGE_TEXT LIKE '%ORA-%' OR UPPER (MESSAGE_TEXT) LIKE '%ERROR%')
4 AND CAST (ORIGINATING_TIMESTAMP AS DATE) > SYSDATE - 1
5 GROUP BY SUBSTR (MESSAGE_TEXT, 1, 300);

MESSAGE_TEXT
--------------------------------------------------------------------------------
CNT
----------

Fatal NI connect error 12170.
63

Errors in file /u01/app/oracle/diag/rdbms/MSDB/MSDB/trace/MSDB_j000_33284.trc
(incident=663845):
ORA-01578: ORACLE data block corrupted (file # 7, block # 2241925)
ORA-01110: data file 7: '+DATATEST/MSDB/DATAFILE/default_tbs.298.1031185857'
ORA-26040: Data block was loaded using the NOLOGGING o

MESSAGE_TEXT
--------------------------------------------------------------------------------
CNT
----------
1
```

7- Check the latest Archivelog and Full Backup are done or not

Use the following script to check Backups.

```
SELECT TO_CHAR (start_time, 'DD-MM-YYYY HH24:MI:SS') start_time, input_type, status, ROUND (elapsed_seconds / 3600, 1) time_hr,INPUT
v$rman_backup_job_details WHERE START_TIME > SYSDATE - 3 ORDER BY start_time DESC;
```

```
SQL> set pagesize 1000 line 200
SQL>
SQL> SELECT TO_CHAR (start_time, 'DD-MM-YYYY  HH24:MI:SS') start_time,         input_type,         status,         ROUND (elapsed_se
  2  v$rman_backup_job_details   WHERE START_TIME > SYSDATE - 3 ORDER BY start_time DESC;

START_TIME           INPUT_TYPE    STATUS                     TIME_HR      IN_GB     OUT_GB OUTPUT_DEVICE_TYP
-------------------- ------------- ----------------------- ---------- ---------- ---------- -----------------
10-03-2020  14:38:40 ARCHIVELOG    COMPLETED                        0 .874987602 .876464844 SBT_TAPE
09-03-2020  18:39:21 DB INCR       COMPLETED                      5.9 1281.08734 1202.46387 SBT_TAPE
08-03-2020  18:00:30 DB INCR       COMPLETED                      5.7 1279.00639  1200.3584 SBT_TAPE
08-03-2020  17:27:04 ARCHIVELOG    COMPLETED                        0  .60765028 .608642578 SBT_TAPE
07-03-2020  18:00:23 DB INCR       COMPLETED                      7.3 1279.00921 1200.35254 SBT_TAPE

07-03-2020  15:22:58 ARCHIVELOG    COMPLETED                        0 .625060558 .626220703 SBT_TAPE

73 rows selected.

SQL>
```

8- Check any session blocking the other session (blocking session and Lock control)

Use the following script to check Blocking session state.

```
select s1.username || '@' || s1.machine
|| ' ( SID=' || s1.sid || ' ) is blocking '
|| s2.username || '@' || s2.machine || ' ( SID=' || s2.sid || ' ) ' AS blocking_status
from gv$lock l1, gv$session s1, gv$lock l2, gv$session s2
where s1.sid=l1.sid and s2.sid=l2.sid
and l1.BLOCK=1 and l2.request > 0
and l1.id1 = l2.id1
and l2.id2 = l2.id2 ;
```

9- Check the DBMS jobs running or not and check the status of the Jobs

Use the following script to check Scheduler jobs state.

```
-- Failed Scheduled Jobs
SELECT owner, job_name,status,LOG_DATE, ERROR#, ( EXTRACT (SECOND FROM run_duration) /60 + EXTRACT (MINUTE FROM run_duration) + EXTR
FROM dba_scheduler_job_run_details
WHERE LOG_DATE > SYSDATE - 1 AND status != 'SUCCEEDED' ORDER BY 1 ASC, 4 DESC;



-- Running and Succeeded Scheduled Jobs
SELECT OWNER, JOB_NAME, LAST_START_DATE, STATE
FROM DBA_SCHEDULER_JOBS
WHERE LAST_START_DATE > SYSDATE - 1 AND STATE <> 'SCHEDULED';
```

10- Check the Dataguard is synchronized or not.

Use the following script to check Dataguard status

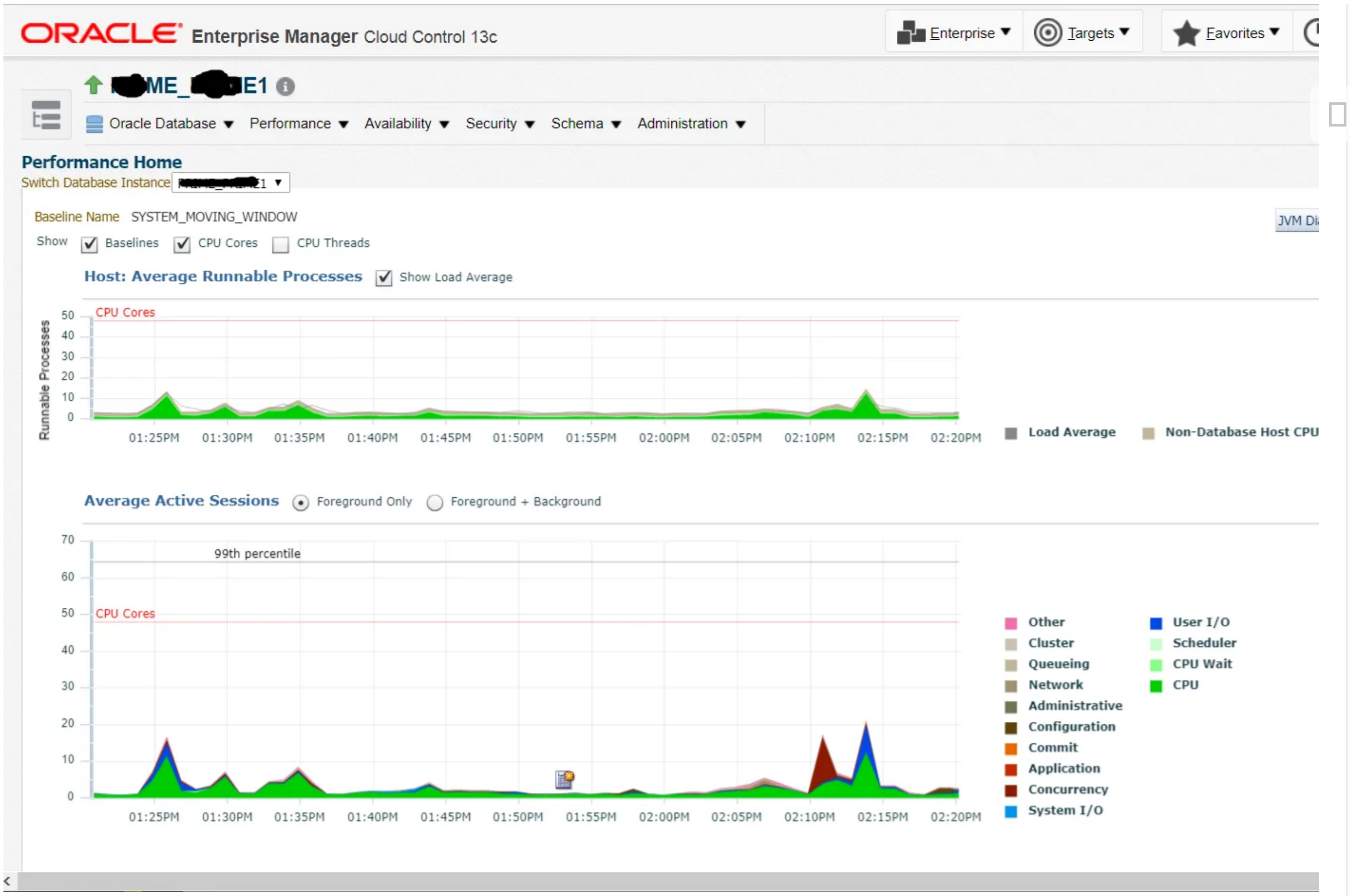
```
select process, client_process,thread#,sequence#,status from v$managed_standby where process like '%MRP%';

select name,value from v$dataguard_stats;
```

```
set lines 1000
select name,value from v$dataguard_stats;
```

```
 NAME                   VALUE
-------------------------------- -------------------------
transport lag          +00 00:03:52
apply lag              +00 00:03:54
apply finish time      +00 00:00:00.001
estimated startup time 16 second
```

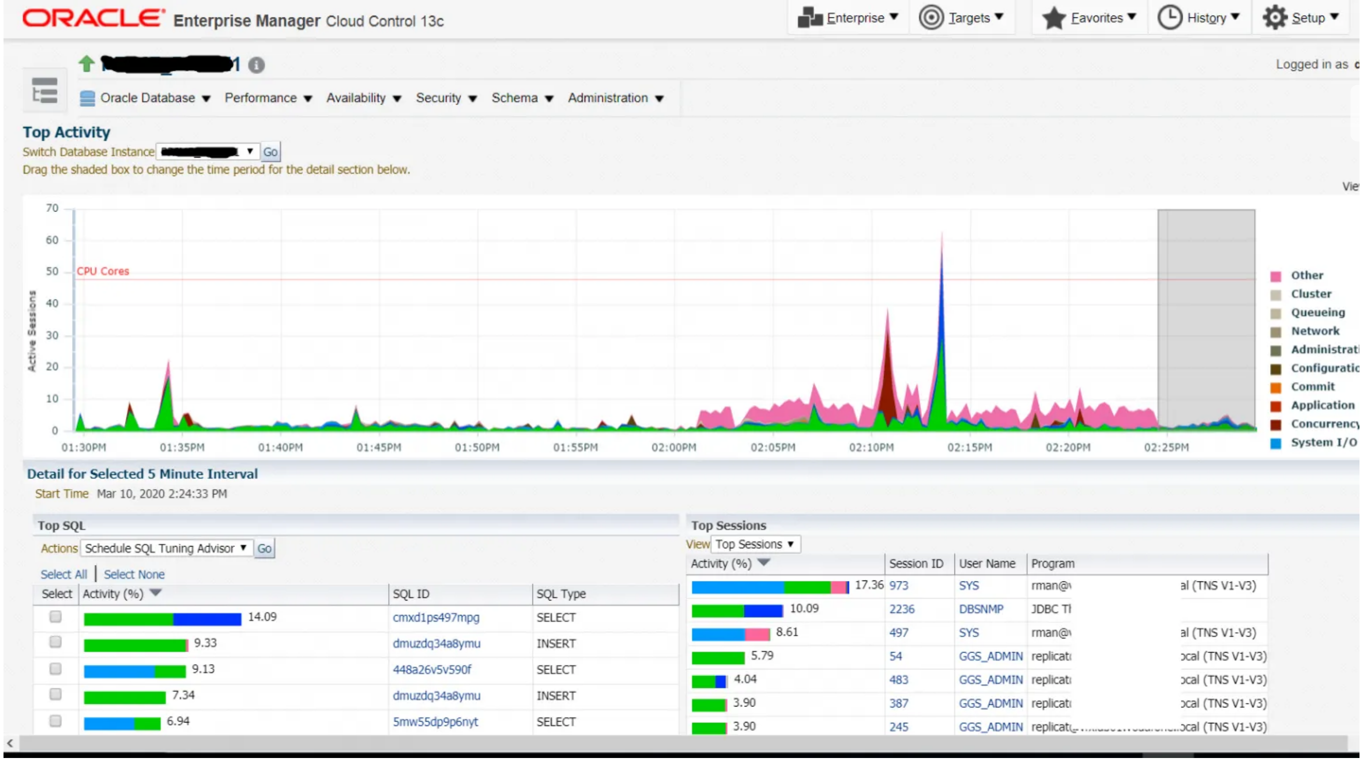
11- Check the Performance Page of Enterprise Manager or Enterprise Manager Cloud Control

Open Performance Page of Enterprise manager Cloud Control as follows to check Performance.



12- Check the TOP session and TOP activity of database.

Open TOP Activity Page of Enterprise manager Cloud Control as follows to check TOP Activity.



Top Activity

Switch Database Instance [redacted] ▾ Go

Drag the shaded box to change the time period for the detail section below.

Detail for Selected 5 Minute Interval

Start Time Mar 10, 2020 2:24:33 PM

Top SQL

Actions Schedule SQL Tuning Advisor ▾ Go

Select All | Select None

Select	Activity (%)	SQL ID	SQL Type
□	14.09	cmxd1ps497mpg	SELECT
□	9.33	dmuzdq34a8ymu	INSERT
□	9.13	448a26v5v590f	SELECT
□	7.34	dmuzdq34a8ymu	INSERT
□	6.94	5mw55dp9p6nyt	SELECT

Top Sessions

View Top Sessions ▾

Activity (%)	Session ID	User Name	Program
17.36	973	SYS	rman@\ al (TNS V1-V3)
10.09	2236	DBSNMP	JDBC Tl
8.61	497	SYS	rman@\ al (TNS V1-V3)
5.79	54	GGS_ADMIN	replicat ɔcal (TNS V1-V3)
4.04	483	GGS_ADMIN	replicat ɔcal (TNS V1-V3)
3.90	387	GGS_ADMIN	replicat ɔcal (TNS V1-V3)
3.90	245	GGS_ADMIN	replicat ɔcal (TNS V1-V3)

13- Detect lock objects

Use the following script to check Lock objects and tables.

```
SELECT B.Owner, B.Object_Name, A.Oracle_Username, A.OS_User_Name
FROM gv$Locked_Object A, All_Objects B
WHERE A.Object_ID = B.Object_ID;
```

14- Check the SQL query consuming lot of resources (CPU and Disk Resources)

Use the following script to check TOP CPU and Disk SQL Statements.

```
select * from (
select ss.sql_text,
a.SQL_ID,
sum(CPU_TIME_DELTA),
sum(DISK_READS_DELTA),
count(*)
from
DBA_HIST_SQLSTAT a, dba_hist_snapshot s,v$sql ss
where
s.snap_id = a.snap_id and a.sql_id=ss.sql_id
and s.begin_interval_time > sysdate -1
group by
ss.sql_text,a.SQL_ID
order by
sum(CPU_TIME_DELTA) desc)
where rownum<20;
```

15- Check the usage of physical RAM and SGA – Paging or Swapping exist or not.

```
[oracle@msddbadm01 ~]$ free -m
```

```
[oracle@msddbadm01 ~]$ free -m
             total       used       free     shared    buffers     cached
Mem:        772257     739605      32652       6174        491     397834
-/+ buffers/cache:     341279     430978
Swap:        24575       8035      16540
```

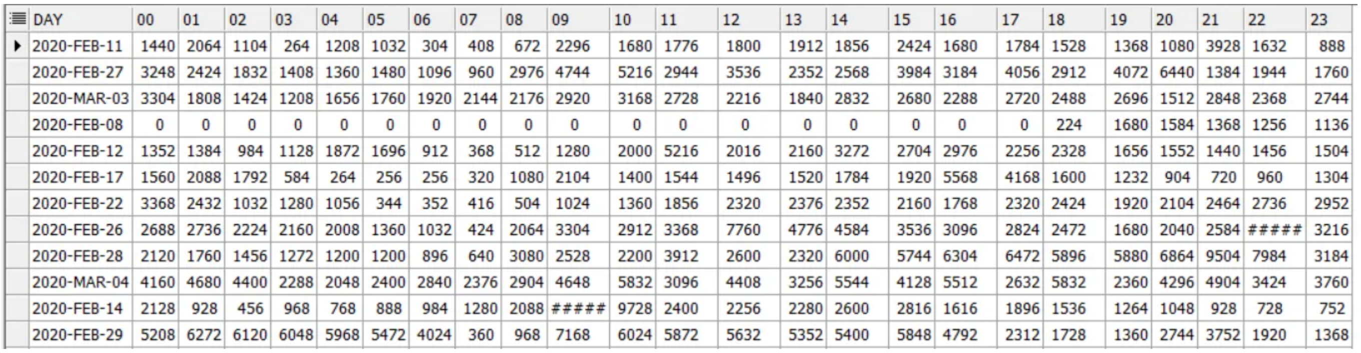
```
SQL> set linesize 150
SQL> set pagesize 150
SQL> select * from v$sgainfo;

NAME                                  BYTES RES     CON_ID
-------------------------------- ---------- --- ----------
Fixed SGA Size                     29906520 No           0
Redo Buffers                      207720448 No           0
Buffer Cache Size                1.1543E+11 Yes          0
In-Memory Area Size              2.1475E+10 No           0
Shared Pool Size                 4.5634E+10 Yes          0
Large Pool Size                  3758096384 Yes          0
Java Pool Size                    536870912 Yes          0
Streams Pool Size                2.6844E+10 Yes          0
Shared IO Pool Size               536870912 Yes          0
Data Transfer Cache Size                  0 Yes          0
Granule Size                      536870912 No           0
Maximum SGA Size                 2.1445E+11 No           0
Startup overhead in Shared Pool  1180135528 No           0
Startup NUMA Shared Pool memory  1.0737E+10 No           0
Free SGA Memory Available         536870912              0

15 rows selected.

SQL> select * from v$sgastat;

POOL           NAME                            BYTES     CON_ID
-------------- -------------------------- ---------- ----------
               fixed_sga                    29906520          0
               buffer_cache               1.1489E+11          0
               log_buffer                  207720448          0
               shared_io_pool              536870912          0
shared pool    free memory                6701646384          0
shared pool    v_inc_meter_info_problem         1728          0
shared pool    kghpinfo freelist                 192          0
shared pool    enqueue_hash                   222864          0
shared pool    kmgsb circular statistics      162560          0
shared pool    SGA Obsolete Param Source          24          0
```

16- Check Log Switch and Archivelog generation frequency.

You can query and list the Log Switch (Archivelog) Frequency map according to per hour and daily as follows.

```
select to_char(first_time,'YYYY-MON-DD') day,
to_char(sum(decode(to_char(first_time,'HH24'),'00',1,0)),'9999') "00",
to_char(sum(decode(to_char(first_time,'HH24'),'01',1,0)),'9999') "01",
to_char(sum(decode(to_char(first_time,'HH24'),'02',1,0)),'9999') "02",
to_char(sum(decode(to_char(first_time,'HH24'),'03',1,0)),'9999') "03",
to_char(sum(decode(to_char(first_time,'HH24'),'04',1,0)),'9999') "04",
to_char(sum(decode(to_char(first_time,'HH24'),'05',1,0)),'9999') "05",
to_char(sum(decode(to_char(first_time,'HH24'),'06',1,0)),'9999') "06",
to_char(sum(decode(to_char(first_time,'HH24'),'07',1,0)),'9999') "07",
to_char(sum(decode(to_char(first_time,'HH24'),'08',1,0)),'9999') "08",
to_char(sum(decode(to_char(first_time,'HH24'),'09',1,0)),'9999') "09",
to_char(sum(decode(to_char(first_time,'HH24'),'10',1,0)),'9999') "10",
to_char(sum(decode(to_char(first_time,'HH24'),'11',1,0)),'9999') "11",
to_char(sum(decode(to_char(first_time,'HH24'),'12',1,0)),'9999') "12",
to_char(sum(decode(to_char(first_time,'HH24'),'13',1,0)),'9999') "13",
to_char(sum(decode(to_char(first_time,'HH24'),'14',1,0)),'9999') "14",
to_char(sum(decode(to_char(first_time,'HH24'),'15',1,0)),'9999') "15",
to_char(sum(decode(to_char(first_time,'HH24'),'16',1,0)),'9999') "16",
to_char(sum(decode(to_char(first_time,'HH24'),'17',1,0)),'9999') "17",
to_char(sum(decode(to_char(first_time,'HH24'),'18',1,0)),'9999') "18",
to_char(sum(decode(to_char(first_time,'HH24'),'19',1,0)),'9999') "19",
to_char(sum(decode(to_char(first_time,'HH24'),'20',1,0)),'9999') "20",
to_char(sum(decode(to_char(first_time,'HH24'),'21',1,0)),'9999') "21",
to_char(sum(decode(to_char(first_time,'HH24'),'22',1,0)),'9999') "22",
to_char(sum(decode(to_char(first_time,'HH24'),'23',1,0)),'9999') "23"
from v$log_history group by to_char(first_time,'YYYY-MON-DD');
```

Query result is as follows.



	DAY	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23
▶	2020-FEB-11	1440	2064	1104	264	1208	1032	304	408	672	2296	1680	1776	1800	1912	1856	2424	1680	1784	1528	1368	1080	3928	1632	888
	2020-FEB-27	3248	2424	1832	1408	1360	1480	1096	960	2976	4744	5216	2944	3536	2352	2568	3984	3184	4056	2912	4072	6440	1384	1944	1760
	2020-MAR-03	3304	1808	1424	1208	1656	1760	1920	2144	2176	2920	3168	2728	2216	1840	2832	2680	2288	2720	2488	2696	1512	2848	2368	2744
	2020-FEB-08	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	224	1680	1584	1368	1256	1136
	2020-FEB-12	1352	1384	984	1128	1872	1696	912	368	512	1280	2000	5216	2016	2160	3272	2704	2976	2256	2328	1656	1552	1440	1456	1504
	2020-FEB-17	1560	2088	1792	584	264	256	256	320	1080	2104	1400	1544	1496	1520	1784	1920	5568	4168	1600	1232	904	720	960	1304
	2020-FEB-22	3368	2432	1032	1280	1056	344	352	416	504	1024	1360	1856	2320	2376	2352	2160	1768	2320	2424	1920	2104	2464	2736	2952
	2020-FEB-26	2688	2736	2224	2160	2008	1360	1032	424	2064	3304	2912	3368	7760	4776	4584	3536	3096	2824	2472	1680	2040	2584	#####	3216
	2020-FEB-28	2120	1760	1456	1272	1200	1200	896	640	3080	2528	2200	3912	2600	2320	6000	5744	6304	6472	5896	5880	6864	9504	7984	3184
	2020-MAR-04	4160	4680	4400	2288	2048	2400	2840	2376	2904	4648	5832	3096	4408	3256	5544	4128	5512	2632	5832	2360	4296	4904	3424	3760
	2020-FEB-14	2128	928	456	968	768	888	984	1280	2088	#####	9728	2400	2256	2280	2600	2816	1616	1896	1536	1264	1048	928	728	752
	2020-FEB-29	5208	6272	6120	6048	5968	5472	4024	360	968	7168	6024	5872	5632	5352	5400	5848	4792	2312	1728	1360	2744	3752	1920	1368

Do you want to learn Oracle Database for Beginners, then read the following articles.

> *Oracle Tutorial | Oracle Database Tutorials for Beginners (Junior Oracle DBA)*

Do you want to use Oracle DBA Scripts All in One For Oracle RAC, then read the following articles.

> *Useful Oracle RAC (Cluster Command) Scripts | Oracle DBA Scripts All in One -1*

📊 2,193 views last month, 1 views today