

FERRAMENTAS SCRIPTs DBA

Este documento foi originado no dia a dia de trabalho é fruto de diversas situações, em diversos projetos, clientes e versões diferentes de bancos de dados e sistemas.

Teve colaboração de diversos profissionais brilhantes*, algumas coisas da internet e foi elaborado e utilizado em diversos clientes e seu início foi em 2009.

Não me parei para formatá-lo, acertar parágrafos, letras, corrigir erros de português. Isso não tem necessidade no dia a dia. Não é para ser uma obra de literatura e sim um ferramenta útil (e muito) para o Administrador de Banco de Dados, de Sistemas, etc.

Para terem uma idéia, eu utilizo ele em formato txt que é o mais simples para copiar/colar quando preciso utilizá-lo.

A sua idéia inicial é ter tudo a mão quando precisar, de forma rápida, simples e eficiente.

Espero que seja muito útil no seu dia a dia.

Caso queira se aprofundar, conhecer mais da ferramenta Banco de dados. Estarei à disposição em: www.rauldba.com.br/cursos

**Agradecimentos pela brilhante participação aos amigos:*

Alessandro Gerondo

Roberto Faucz

Felipe Romeu Gregolewich

José Laurindo Chiappa

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

Scripts Oracle ferramentas de apoio para Dia a dia do DBA*Versão - 02/06/2016 –10:40***Índice:****LOCKS**

- 1- Verifica Lock ORACLE 10g, 11g e 12c
- 2- Verifica Lock ORACLE 8i e 9i
- 3- Verifica lock de dicionário de dados Oracle 10g
- 4- Verificar quem está bloqueando quem
- 5- Verifica qual sessão está bloqueando no caso de lock (pode ser usado com o script 1)
- 6- Verifica usuários locando tabelas
- 7- Verifica quais sessões estão utilizando um determinado objeto no RAC

OWI

- 7- OWI - Verificar eventos de espera tempos de resposta do BD
- 8- OWI - Verificando enfileiramentos
- 9- Mostra desde quando a sessão está parada
- 10- Mostra qual horário a sessão fez seu ultimo trabalho
- 11- Identifica as transações Pendentes
- 12- Mostra o tempo estimado de execução de uma tarefa inclusive backup

VERIFICAR PARA KILL

- 13- Verificacao e SQL TEXT
- 14- Verificando instrucao SQL pelo SID
- 15- Identificar usuário por SID
- 16- Identificar usuário por SPID
- 17- Gera spid de todas as sessões que utilizam um determinado objeto
- 18- Gerar Trace (manual da geração de trace)

ALOCÇÃO

- 19- verifica o tamanho do BD (físico)
- 20- verifica valor alocado para bd por tablespace
- 20.2- Verificando total do banco incluindo controlfiles, redolog, temp e datafiles
- 21- Verifica realmente utilizado pelo BD por tablespace
- 21.1- Script TOAD detalha tablespace
- 22- Saber valor que um owner ocupa no banco de dados efetivamente (nao espaço alocado) separado por tablespace
- 23- Verifica a alocação de determinada tabela separada por datafiles ou seja consumo de espaço da tabela por DF
- 24- Saber valor que uma tabela ocupa no banco de dados efetivamente (nao espaço alocado) separado por tablespace
- 25- verifica entre duas tabelas quantos byte ainda falta inserir no caso de estar sendo feito insert ele mostra dinâmico

DIVERSOS

- 26- Lista execuções e loads por SQL_ID(Considerar ofensivos quando o número de loads for maior que executions)
- 27- Área ocupada pelos componentes da SGA

Proibida distribuição sem citar a fonte**Raul Francisco C F Andrade – RAULdba****Consultor Oracle nas versões 9i, 10 e 11g e 12c.****Instrutor Oracle**

- 28- Geração de senha randômica
- 29- Verificando limites de banco de dados Sessions, process, etc
- 30- Agrupar usuários por tablespaces
- 31- verifica a quantidade de REDO gerado por Sessão
- 32- 32- Resolver problema com transação distribuída pendente
- 33- mostra as conexões de entrada e saída para conexões pendentes
- 34- Mostra as alternâncias de log (switch log)
- 35- Recompilar todos objetos inválidos
- 36- Verifica a utilização da área de recuperação Flash DB_RECOVERY_FILE_DEST
- 37- Trabalhando com segmento de rollback (ver tamanho dos segmentos, verificar quem está utilizando, promover o shrink)
- 38- startar queue Oracle (AQ)
- 39- Alterar Profile
- 40- Caso o log do Listener cresça muito e precise zerá-lo
- 41- criando dblink com RAC
- 42- Criando sinônimo
- 43- Verificando auditoria de login

TRABALHANDO COM ASM

- 44- Verifica espaço livre em discos ASM
- 44.1-Verificando diferença de alocado alocado para tablespace e livre
- 45- Mostra o crescimento do BD listado por mês (para 6 meses) totalizando no final
- 46- Verificando asm disks file system, nome físico, etc
- 47- Alocação de discos no diskgroup e adicionar datafiles ASM
- 47.2-ver operação no asm

VERIFICAÇÃO DE TEMPOS E CONSUMOS NO BANCO

- 48- Movimentacao de I/O
- 49- verificação de tempo de banco pelo SID (tempo em microsegundos)
- 51- Mostra quais sessões estão consumindo mais CPU
- 52- Encontra quem está utilizando a tablespace temporária (TEMP)
- 53- encontrar as 10 tabelas que mais fazem I/O no BD
- 54- Encontra sessões ativas executando queries com o maior tempo
- 55- Encontra sessões com o maior tempo de banco usado
- 56- Verifica o tempo de backup de cada bd cadastrado no catálogo do RMAN
- 57- Verificações básicas RAC
- 58- Verificação do plano de execução
- 59- Liberando espaço com Shrink
- 60- Verifica sessões que estão utilizando determinado objeto
- 61- Criando índice exemplo
- 62- Verificar onde usuários estão conectando
- 63- corrigir erro de flashback
- 64- Criação de Tablespace
- 65- Movendo tabela entre tablespaces
- 66- verifica Laudo do ADDM
- 66.2- Criando report ADDM com DBMS_ADVISOR
- 67- Ver se alguém ficou preso no backup (status active)
- 68- Apagar backups obsoletos e fazer crosscheck
- 69- Erro na v\$recovery_file_dest

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

- 70- Fazer insert as select com hint de append e parallel
- 71- Verificar quantidade de conexões por instância RAC
- 72- Trace do Listener
- 73- LINUX/AIX/HPUX
- 74- Copiar de uma máquina para a outra com o Linux
- 75- Estimando espaço para criação de um índice
- 76- Trabalhando com tabelas e índices fragmentados (descobrimo e desfragmentando)
- 76.2 Desfragmentando com MOVE
- 77- Gerando exp e imp sem gerar arquivo físico
- 78- DBMS coleta automática de estatísticas
- 79- Descobrir quais tabelas usam determinada PK como FK
- 80- Conectar via sqlplus sem TNSNAMES registrado
- 81- Verifica sessões consumidoras de UNDO
- 82- Extrair DDL de um objeto
- 83- Verificando traces rodando
- 84- Criar script de grant por atacado
- 85- Descobrir o SCN de uma banco de dados
- 86- Transportar tablespace de uma base para outra
- 87- Saber estatisticamente informações de um SQL que esteja rodando
- 88- Duplicando banco com RMAN usando catálogo
- 89- Verificando tamanho máximo do datafile
- 90- Descobrir o valor consumido em bytes pos determinada quantidade de linhas
- 91- Verificando estatísticas de uma determinada sessão
- 92- Fazendo um import em background para não perder caso a sessão caia
- 93- Verificar quais views utilizam determinado DBLINK
- 94- RECUPERANDO DADOS MODIFICADOS ACIDENTALMENTE através da RECYCLEBIN E FLASHBACK (table e database)
- 95- Verificando latch free
- 96- Aplicando patch do Oracle
- 97- Configurando manualmente o Oracle Enterprise Manager
- 98- Contar quantidade de Commits em pl com LOOP
- 99- Voltando Flashback database
- 100- Determina o período de Flashback atual (até onde pode-se voltar o banco)
- 101- Exibe a cota de disco da área de recuperação flash
- 102- Mostra as alterações de taxa na geração de dados de flashback
- 103- Gerando AWR pelo SO (Sem EM) e gerando relatório comparativo
- 104- LOCK DE ENQUEUE
- 105- gc current block 2-way
- 106- Verificar a seletividade de uma coluna
- 107- Verificar o plano de execução
- 108- Verificação de qual objeto de uma sessão vem um wait event
- 109- Verificar a quantidade de archive gerado por thread e hora
- 109.1- Demonstra o Redolog size recomendado
- 110- Conectar como outro usuário
- 111- Armazenar tabela no Buffer para resolver problemas de I/O
- 112- Verificar registros duplicados
- 113- Verificar usuário que está tentando logar com senha incorreta.
- 114- Atualizar Materialized View (SNAPSHOT)
- 115- Verificar JOB's que estão rodando

- 116- Resolver problema quando sessão cai enquanto está criando o índice e ele aparece na dba_indexes e não na dba_segments e demais (índice é criado pela metade)
- 117- Comando para manutenção de banco bloqueando acessos sem parar o banco.
- 118- Habilitando Block change tracking para otimizar backups incrementais
- 119- verifica a porcentagem de processos em relação à quantidade máxima de processos permitidos pelo parâmetro de inicialização PROCESSES.
- 120- Descobrir os SPID e gerar kill -9 para usuários que estejam paradas a mais de 60 minutos (3600 segundos)
- 121- Mostra os objetos que estão sendo lidos fisicamente e causando o db sequential read
- 122- Encontrar qual o valor atribuído para determinada bind
- 123- Trigger para monitorar registro modificado via rowid
- 124- Melhora de performance com objetos colocados keep na shared_pool
- 125- Verificar principais recursos da máquina
- 126- Verificar início e final de processo para tomada de tempo
- 127- Trigger para detectar tentativas de logon frustradas
- 128- Criar script com todos os grants da base por usuários e base toda
- 129- Trabalhando com sequence
- 130- Procedimento para compilar Objeto em uso
- 131- Verificar quantidade média (não exata) de registros inseridos durante um insert.
- 132- Fazer carga de arquivo texto em tabela do banco.
- 133- DEBUG Oracle
- 134- Segment advisor (verifica a necessidade de shrink)
- 135- Descobrir os High Water Mark do banco
- 136- Exemplo de PL para criar commit a cada quantidade específica de registros
- 137- instalando Auditoria de banco de dados (by Burleson com adaptações)
- 138- Verificações de Response-time
- 139- Aplicando patch com Opatch Linux e Unix
- 140- Relink do Oracle Home
- 141- Instalando Oracle APEX 3.2 no Oracle 10g R2
- 142- Sessões com DB links - resolvendo problemas de excesso de sessões
- 143- Aplicando archives em dataguard quando ocorre perda de sequência
- 144- Executar comando shell de um servidor para outro
- 145- Criando tabelas externas
- 146- Rodando o hanganalyze
- 147- Habilitando o Resource Manager para evitar o paralelismo
- 148- Verificando a utilização da PGA estatisticamente
- 149- Verificando como setar o OPTIMIZER_INDEX_COST_ADJ
- 150- Alterações em REDOLOG
- 151- Verificar a quantidade de commits das sessões;
- 152- Verificando quais tabelas sofreram modificações e não tem estatísticas
- 153- Query para detectar excessos de archivelog
- 154- Mostrar SQL master and slaves e pra fazer monitoração de Diferença de waits entre sessões
- 155- Criando Script para startup automático do Oracle no Linux
- 156- Verificação dos privilégios de um determinado usuário inclusive com os herdados.
- 157- Trabalhando com DATAPUMP
- 158- Verificação das configurações NLS da banco de dados
- 159- Habilitar/desabilitar paralelos para a sessão.
- 160- Habilitando e desabilitando o job de coleta automática de estatísticas.
- 161- verificar JOB's rodando.

- 162- Criando JOB para coleta de estatísticas
- 163- Verificando a geração das estatísticas
- 164- Haduken
- 165- Criando procedure para usuário executar kill session
- 166- Movendo LOB
- 167- Encontrando HOTBLOCK
- 168- Debug Completo ORACLE
- 169- Criando partição por data
- 170- Monitora status do RMAN enquanto o backup está rodando
- 171- Testando a eficiência da Library Cache
- 172- Setando coleta de estatística automática
- 173- Criando Job para coleta de estatísticas de schema
- 174- Script para contagem de linha de todas as tabelas de um owner (faz em massa)
- 175- Mostrar hidden parameters (parâmetros Ocultos Oracle)
- 176- Verificar objetos em cache.
- 177- Descobrir objetos com erros de compilação e os erros
- 178- Trigger que registra CREATE ou DROP
- 179- Cria JOB para executar procedure de limpa PRIS
- 180- Alterar nome ou local de datafiles em um tablespace;
- 181- Alterando o datafile de um tablespace para outro filesystem
- 182- Criando index buffer em uma área de 32K de buffer
- 183- Descobrir quais queries estão gerando mais parse
- 184- Verifica constraint FK que estão sem índices.
- 185- Inclui índice para foreign key sem índice
- 186- Criando scheduler para executar determinada procedure
- 187- Histórico das conexões
- 188- Fazendo select na tabela para saber como ela era em determinada data
- 189- Inserir as select colocando log em outra tabela e não fazendo rollback após erro.
- 190- Verificando propriedades do Banco de dados.
- 191- Eliminar job
- 192- Criando Mview para ser atualizada a cada dia 02 de cada mês
- 193- Usando Oracle Replay (RAT – Real Application Testing)
- 194- Automatic SQL Tuning in Oracle Database 10g
- 195- Trabalhando com Flashback Query, database e version.
- 196- Se apagar datafile com banco aberto.
- 197- Criando JOB para DML
- 198- Verificando datas da coleta de estatísticas
- 199- Trabalhando com ADVISOR TUNING
- 200- Extraíndo DDL de criação de objetos diversas formas
- 201- Setando o log do listener
- 202- Criando tablespace e trabalhando com ele
- 203- Tornar usuário como usuário do Enterprise Manager
- 204- Comandos úteis de cluster
- 205- Verificar versão do SO Linux
- 206- Acertar backspace no sqlplus
- 207- Estimando o tamanho de uma Materialized View

1- Verifica Lock ORACLE 10, 11g e 12c

1.a - Versão 1

```
select
B.OSUSER OSUSER_BLOQUADOR
,A.OSUSER OSUSER_BLOQUADO
,D.OBJECT_NAME
,B.USERNAME OWNER_BLOQUEADOR
,A.USERNAME OWNER_BLOQUEADO
,A.blocking_session sessao_bloqueadora
,A.sid sessao_sendo_bloqueada
,A.INST_ID INSANCIA_BLOQUEADORA
,B.INST_ID INSANCIA_SENDO_BLOQUEADA
,A.serial#
,A.wait_class
,A.machine
,A.seconds_in_wait
from gv$session A,gv$session B,gv$locked_object C, dba_objects D where
A.blocking_session=B.SID
AND A.blocking_session is not null
and A.INST_ID = C.INST_ID
and A.sid = C.session_id
and C.object_id = D.object_id
order by A.blocking_session;
```

1.b - Versão 2

```
SELECT os_user_name "OS User",
process "OS Pid",
oracle_username "Oracle User",
l.sid "SID",
DECODE(type,
'MR', 'Media Recovery',
'RT', 'Redo Thread',
'UN', 'User Name',
'TX', 'Transaction',
'TM', 'DML',
'UL', 'PL/SQL User Lock',
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
'DX', 'Distributed Xaction',
'CF', 'Control File',
'IS', 'Instance State',
'FS', 'File Set',
'IR', 'Instance Recovery',
'ST', 'Disk Space Transaction',
'TS', 'Temp Segment',
'IV', 'Library Cache Invalidation',
'LS', 'Log Start or Switch',
'RW', 'Row Wait',
'SQ', 'Sequence Number',
'TE', 'Extend Table',
'TT', 'Temp Table', type) "Lock Type",
DECODE(lmode,
0, 'None',
1, 'Null',
2, 'Row-S (SS)',
3, 'Row-X (SX)',
4, 'Share',
5, 'S/Row-X (SSX)',
6, 'Exclusive', lmode) "Lock Held",
DECODE(request,
0, 'None',
1, 'Null',
2, 'Row-S (SS)',
3, 'Row-X (SX)',
4, 'Share',
5, 'S/Row-X (SSX)',
6, 'Exclusive', request) "Lock Requested",
DECODE(block,
0, 'Not Blocking',
1, 'Blocking',
2, 'Global', block) "Status",
owner "Owner",
object_name "Object name"
FROM gv$locked_object lo, dba_objects do, gv$lock l --- ambiente RAC
--FROM v$locked_object lo, dba_objects do, v$lock l --- standalone
WHERE lo.object_id = do.object_id
AND l.sid = lo.session_id
--and os_user_name = 'cintia.morandi' --- PODE FILTRAR O USUARIO
and block=1 ---- esse mostra as sessões que estão bloqueando (LOCK)
--and request = 6 ---- esse mostra as sessões que estão esperando (BLOCK)
--and (block=1 or request = 6)
```

Visualização rápida do DEADLOCK:

```
select sid, serial#, username, command, lockwait, osuser from v$session where lockwait is not
null
```


Qual a query causadora:

```
select sql_text from v$sqltext where (address,hash_value) in (select
sql_address,sql_hash_value from v$session where lockwait is not null) order by address,
hash_value, piece
```

```
SELECT /*+ rule */ l.inst_id,s.event, l.SID, s.serial# serial, p.spid, s.username,
s.status, s.osuser, s.machine, s.program,
      to_char(s.logon_time,'dd/mm/yyyy hh24:mm:ss') LOGON_TIME, l.ctime LOCK_TIME
FROM gv$lock l, gv$session s, gv$process p
WHERE s.inst_id = l.inst_id
      and s.inst_id = p.inst_id
      AND s.SID = l.SID
      and s.PADDR = p.addr
      AND (l.id1, l.id2, l.TYPE) IN (SELECT id1, id2, TYPE
                                   FROM gv$lock
                                   WHERE request > 0)
ORDER BY ctime DESC;
```

-- 1.1 verifica lock e mostra a query

```
SELECT w.SID,
       w.event,
       w.seconds_in_wait,
       SQL.sql_text
FROM v$session_wait w, v$session s, v$process p, v$sqltext SQL
WHERE w.SID = s.SID
      AND s.paddr = p.addr
      AND SQL.address = s.sql_address
      AND SQL.hash_value = s.sql_hash_value
      AND w.wait_class != 'Idle'
ORDER BY w.seconds_in_wait, w.SID, SQL.piece;
```

-- Visualiza todos os correntes objetos lockados:

```
SELECT username U_NAME, owner OBJ_OWNER,
object_name, object_type, s.osuser,
DECODE(l.block,
0, 'Not Blocking',
1, 'Blocking',
2, 'Global') STATUS,
DECODE(v.locked_mode,
0, 'None',
1, 'Null',
2, 'Row-S (SS)',
3, 'Row-X (SX)',
4, 'Share',
5, 'S/Row-X (SSX)',
6, 'Exclusive', TO_CHAR(lmode)
) MODE_HELD
FROM gv$locked_object v, dba_objects d,
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
gv$lock l, gv$session s
WHERE v.object_id = d.object_id
AND (v.object_id = l.id1)
AND v.session_id = s.sid
ORDER BY username, session_id;
```

-- Lista os objetos que tem sido Locado por 60 segundos ou mais:

```
SELECT SUBSTR(TO_CHAR(w.session_id),1,5) WSID, p1.spid WPID,
SUBSTR(s1.username,1,12) "WAITING User",
SUBSTR(s1.osuser,1,8) "OS User",
SUBSTR(s1.program,1,20) "WAITING Program",
s1.client_info "WAITING Client",
SUBSTR(TO_CHAR(h.session_id),1,5) HSID, p2.spid HPID,
SUBSTR(s2.username,1,12) "HOLDING User",
SUBSTR(s2.osuser,1,8) "OS User",
SUBSTR(s2.program,1,20) "HOLDING Program",
s2.client_info "HOLDING Client",
o.object_name "HOLDING Object"
FROM gv$process p1, gv$process p2, gv$session s1,
gv$session s2, dba_locks w, dba_locks h, dba_objects o
WHERE w.last_convert > 60
AND h.mode_held != 'None'
AND h.mode_held != 'Null'
AND w.mode_requested != 'None'
AND s1.row_wait_obj# = o.object_id
AND w.lock_type(+) = h.lock_type
AND w.lock_id1(+) = h.lock_id1
AND w.lock_id2 (+) = h.lock_id2
AND w.session_id = s1.sid (+)
AND h.session_id = s2.sid (+)
AND s1.paddr = p1.addr (+)
AND s2.paddr = p2.addr (+)
ORDER BY w.last_convert DESC;
```

-- Mostra a sessão que está locada e não o causador do lock;

```
select
c.owner,
c.object_name,
c.object_type,
b.sid,
b.serial#,
b.status,
b.osuser,
b.machine,
s.sql_text
from v$locked_object a,
v$session b,
dba_objects c,
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
v$sqlarea s
where
b.sid = a.session_id
and
a.object_id = c.object_id
and
b.sql_id = s.sql_id;
```

-- A Sessão que o SQL_TEXT ficar em branco é a sessão causadora do LOCK:

```
SELECT username U_NAME, owner OBJ_OWNER,
object_name, object_type, s.osuser,s.SID,a.sql_text,
DECODE(l.block,
0, 'Not Blocking',
1, 'Blocking',
2, 'Global') STATUS,
DECODE(v.locked_mode,
0, 'None',
1, 'Null',
2, 'Row-S (SS)',
3, 'Row-X (SX)',
4, 'Share',
5, 'S/Row-X (SSX)',
6, 'Exclusive', TO_CHAR(lmode)
) MODE_HELD
FROM gv$locked_object v, dba_objects d,
gv$lock l, gv$session s, gv$sqlarea a
WHERE v.object_id = d.object_id
AND (v.object_id = l.id1)
AND v.session_id = s.sid
AND s.sql_id = a.sql_id(+)
ORDER BY username, session_id;
```

-- Lista todos os DMLs com LOCK
select * from DBA_DML_LOCKS;

-- Este SQL mostra a Sessão causadora do LOCK:
select * from DBA_BLOCKERS;

-- Mostra dos os locks adquiridos por transações do sistema:
select * from V\$LOCKED_OBJECT;

select * from v\$LOCK;

-- Descobrir qual Sessão está bloqueando tal:

```
select 'SID ' || l1.sid || ' is blocking ' || l2.sid blocking
from v$lock l1, v$lock l2
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
where l1.block =1 and l2.request > 0
and l1.id1=l2.id1
and l1.id2=l2.id2;
```

2- Verifica Lock ORACLE 8i e 9i

```
SELECT /*+ RULE */ l.SID, s.serial# serial, p.spid, s.username, s.status, s.osuser, s.machine,
s.program,
      to_char(s.logon_time,'dd/mm/yyyy hh24:mm:ss') LOGON_TIME, l.ctime LOCK_TIME
FROM gv$lock l, gv$session s, gv$process p
WHERE s.inst_id = l.inst_id
and s.inst_id = p.inst_id
AND s.SID = l.SID
and s.PADDR = p.addr
AND (l.id1, l.id2, l.TYPE) IN (SELECT id1, id2, TYPE
                              FROM gv$lock
                              WHERE request > 0)
ORDER BY ctime DESC;
```

Observação: pode-se ainda utilizar o dba_blockers para verificar as sessões causadoras de bloqueios.

3 - Verifica lock de dicionário de dados Oracle 10g

```
select /*+ ordered */
w1.sid waiting_session, h1.sid holding_session,
w.kglktype lock_or_pin, w.kglkhd address,
decode(h.kglkmod, 0, 'None', 1, 'Null', 2, 'Share', 3, 'Exclusive', 'Unknown') mode_held,
decode(w.kglkreq, 0, 'None', 1, 'Null', 2, 'Share', 3, 'Exclusive', 'Unknown') mode_requested
from
dba_kgllock w,
dba_kgllock h,
v$session w1,
v$session h1
where
(((h.kglkmod != 0)
and (h.kglkmod != 1)
and ((h.kglkreq = 0) or (h.kglkreq = 1)))
and (((w.kglkmod = 0) or (w.kglkmod = 1))
and ((w.kglkreq != 0)
and (w.kglkreq != 1))))
and w.kglktype = h.kglktype
and w.kglkhd = h.kglkhd
and w.kglkuse = w1.saddr
and h.kglkuse = h1.saddr;
```

4 - Verificar quem está bloqueando quem

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
select a.username || '@' || a.machine || ' session id = ' || a.sid || ' esta bloqueando '
|| b.username || '@' || b.machine || ' session id = ' || b.sid
from gv$lock c,
gv$session a,
gv$lock d,
v$session b
where a.sid=c.sid
and b.sid=d.sid
and c.block=1
and d.request > 0
and c.id1 = d.id1
and d.id2 = d.id2;
```

5- Verifica qual sessao está bloqueando no caso de lock (pode ser usado com o script 1)

```
Select * from dba_blockers;
```

6- Verifica usuários locando tabelas

```
select b.owner, b.object_name, b.object_type, a.*,
DECODE ("LOCKED_MODE"
, 0, 'None'
, 1, 'NULL'
, 2, 'Row-Share'
, 3, 'Row-Excl.'
, 4, 'Share'
, 5, 'S/Row-Excl.'
, 6, 'Exclusive'
) as "FORMA DE BLOQUEIO"
from sys.v_$locked_object a, all_objects b
where a.object_id = b.object_id;
```

7- Verifica quais sessões estão utilizando um determinado objeto no RAC

```
select /*+ rule */ s.sid,s.username, a.object, a.type, s.machine, s.terminal, p.spid
from gv$access a, gv$process p, gv$session s
where a.object = 'NOME_OBJETO'
and a.inst_id = s.inst_id
and a.inst_id = p.inst_id
and s.paddr = p.addr
and s.sid = a.sid
```

7- OWI Verificar eventos de espera tempos de resposta do BD

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```

select event, time_waited as time_spent
from v$session_event
where sid = &sid
and event not in (
    'Null event',
    'client message',
    'KXFX: Execution Message Dequeue - Slave',
    'PX Deq: Execution Msg',
    'KXFQ: kxfqdeq - normal dequeue',
    'PX Deq: Table Q Normal',
    'Wait for credit - send blocked',
    'PX Deq Credit: send blkd',
    'Wait for credit - need buffer to send',
    'PX Deq Credit: need buffer',
    'Wait for credit - free buffer',
    'PX Deq Credit: free buffer',
    'parallel query dequeue wait',
    'PX Deque wait',
    'Parallel Query Idle Wait - Slaves',
    'PX Idle Wait',
    'slave wait',
    'dispatcher timer',
    'virtual circuit status',
    'pipe get',
    'rdbms ipc message',
    'rdbms ipc reply',
    'pmon timer',
    'smon timer',
    'PL/SQL lock timer',
    'SQL*Net message from client',
    'WMON goes to sleep')
union all
select b.name, a.value
from v$sesstat a, v$statname b
where a.statistic# = b.statistic#
and b.name = 'CPU used when call started'
and a.sid = &sid;

```

8- OWI - Verificando enfileiramentos

```

SELECT i.instance_name, w.event, COUNT (*), SUM (seconds_in_wait)
FROM gv$instance i,gv$session_wait w
WHERE
i.inst_id = w.inst_id and
event not like 'Streams AQ%' and
event not like 'queue messages%' and
event not like 'SQL*Net%' and
(w.inst_id, SID) IN (SELECT inst_id, SID

```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```

FROM gv$session
WHERE status = 'ACTIVE' AND username IS NOT NULL)
GROUP BY i.instance_name,event
ORDER BY 3 DESC;

```

--Variação

```

SELECT s.username, s.SID, sw.event, p.spid, s.machine, s.terminal, se.total_waits, q.sql_text
FROM v$session_wait sw, v$process p, v$session s, v$session_event se,
v$sqltext_with_newlines q
WHERE sw.SID = s.SID AND p.spid = s.process AND sw.SID = se.SID AND s.sql_hash_value =
q.hash_value
AND sw.event NOT IN('rdbms ipc message') AND s.username IS NOT NULL
ORDER BY se.total_waits DESC, s.SID, q.piece;

```

#####8.2- mostra qual usuário tem determinado evento #####

```

SELECT a.inst_id,a.sid, c.pid, c.spid, a.username, b.event, b.wait_time, b.seconds_in_wait,
b.p1, b.p2, b.p3
FROM gv$session a, gv$session_wait b, gv$process c
WHERE a.sid = b.sid
AND a.paddr = c.addr
and a.inst_id=c.inst_id
AND b.event = '&event'

```

#####8.3- mostra qual eventos teve determinado SID #####

```

SELECT EVENT, AVERAGE_WAIT, TOTAL_TIMEOUTS FROM V$SESSION_EVENT WHERE SID =
&SID ORDER BY AVERAGE_WAIT;

```

#####8.4- Verifica com query OWI qual objeto tem o evento de espera

```

select b.sid,
       nvl(substr(a.object_name,1,30),
           'P1='||b.p1||' P2='||b.p2||' P3='||b.p3) object_name,
       a.subobject_name,
       a.object_type
from   dba_objects a, v$session_wait b, x$bh c
where  c.obj = a.object_id(+)
and    b.p1 = c.file#(+)
and    b.p2 = c.dbablk(+)
and    b.event = 'db file sequential read'
union
select b.sid,
       nvl(substr(a.object_name,1,30),
           'P1='||b.p1||' P2='||b.p2||' P3='||b.p3) object_name,
       a.subobject_name,
       a.object_type
from   dba_objects a, v$session_wait b, x$bh c
where  c.obj = a.data_object_id(+)
and    b.p1 = c.file#(+)

```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
and b.p2 = c.dbablk(+)  
and b.event = 'db file sequential read'  
order by 1;
```

-- Variação:

```
select * from V$LOCK where SID=214
```

```
select object_name, object_type from dba_objects where object_id=30  
OBJECT_ID é o ID1 da primeira query
```

9- Mostra desde quando a sessão está parada

```
select username, logon_time, last_call_et,  
to_char(sysdate-(last_call_et/(60*60*24)), 'hh24:mi:ss') last_work_time  
from gv$session  
where username is not null  
and status = 'INACTIVE';
```

10- Mostra qual horário a sessão fez seu último trabalho

```
select username, logon_time, last_call_et,  
to_char(sysdate-(last_call_et/(60*60*24)), 'hh24:mi:ss') last_work_time  
from gv$session  
where username is not null  
and status = 'ACTIVE'
```

11- Identifica as transações Pendentes

```
Select * from DBA_PENDING_TRANSACTIONS;
```

12 - Mostra o tempo estimado de execução de uma tarefa inclusive backup

```
SELECT SID, DECODE (totalwork, 0, 0, ROUND (100 *sofar / totalwork, 2)) "Percent", MESSAGE  
"Message", start_time, elapsed_seconds, time_remaining  
FROM v$session_longops  
WHERE (SID = &SID AND serial# = &SERIAL)  
ORDER BY SID;
```

-- Tarefa de backup

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

```
SELECT SID, SERIAL#, context, LAST_UPDATE_TIME, substr(message, 1, 30), DECODE  
(totalwork, 0, 0, ROUND (100 *sofar / totalwork, 2)) "concluido %", time_remaining  
FROM v$session_longops where (DECODE (totalwork, 0, 0, ROUND (100 *sofar / totalwork, 2)))  
not in ('100', '0') and MESSAGE like '%RMAN%';
```



```
SELECT SID,SERIAL#,message, DECODE (totalwork,0, 0,ROUND (100 * sofar / totalwork, 2))
"concluido %"
FROM v$session_longops where (DECODE (totalwork,0, 0,ROUND (100 * sofar / totalwork, 2)))
not in ('100','0') and MESSAGE like '%RMAN%';
```

--Monitorando Backup

```
SELECT sid, opname, context, sofar, totalwork, ROUND(sofar/totalwork, 2) "%Complete"
FROM V$SESSION_LONGOPS
WHERE opname LIKE 'RMAN%'
AND opname NOT LIKE '%aggregate%'
AND totalwork != 0
AND sofar <> totalwork;
```

--Completa informação de backups

```
SELECT sid, opname, context, sofar, totalwork, ROUND(sofar/totalwork, 2) "%Complete"
FROM V$SESSION_LONGOPS
WHERE opname LIKE 'RMAN%'
AND opname LIKE '%aggregate%'
AND totalwork != 0
AND sofar <> totalwork;
```

13- Verificacao e SQL TEXT

```
select sql_text
from GV$sqltext_with_newlines where inst_id = &INSTANCE_NUMBER AND
address = (select DECODE(RAWTOHEX(sql_address), '00', prev_sql_addr, sql_address)
from GV$session
where username = '&USERNAME'
and inst_id = &INSTANCE_NUMBER
and sid = &SID)
ORDER BY piece
```

--Observação:

a instance_number se pega com o select do lock (número 01)

14- Verificando instrucao SQL pelo SID

```
select * from v$sql
where address = (select sql_address from v$session where sid =26)
-- where address in(select sql_address from v$session where sid in(50,17))
```

VARIAÇÃO

```
select
u.sid,
substr(u.username,1,12) user_name,
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
s.sql_text
from
v$sqltext s,
v$session u
where
s.hash_value = u.sql_hash_value
and
sql_text not like '%from v$sqltext s, v$session u%'
and u.sid = '&SID'
order by
u.sid;
```

15- Identificar usuário por SID

```
select s.inst_id, s.sid, s.serial#, p.spid, s.status, s.username, s.program
from gv$session s, gv$process p
where s.paddr = p.addr
and s.inst_id = p.inst_id
and s.sid = &sid;
```

--variação cria kill -9

```
SELECT 'kill -9 ' || SPID
FROM V$PROCESS WHERE ADDR IN
(SELECT PADDR FROM V$SESSION WHERE USERNAME LIKE &USERNAME)
```

--variação do KILL

```
ALTER SYSTEM DISCONNECT SESSION 'sid_number,serial_number' IMMEDIATE;
```

16- Identificar usuário por SPID

```
select s.inst_id, s.sid, s.serial#, p.spid, s.status, s.username, s.program
from gv$session s, gv$process p
where s.paddr = p.addr
and s.inst_id = p.inst_id
and p.spid = &spid;
```

opcional

```
select p.PID,p.SPID,s.SID, s.serial#
from v$process p,v$session s
where s.paddr = p.addr
and s.sid = &SESSION_ID
and s.username = &USERNAME;
```

#####17- Gera spid de todas as sessões que utilizam um determinado objeto #####

```
select /*+ rule */ 'kill -9 ' || p.spid, s.inst_id, s.username from gv$process p, gv$session s
where s.inst_id = p.inst_id
and s.paddr = p.addr
AND S.SID IN (select /*+ rule */ a.SID from gv$access a, gv$session s
where object = 'TESTE_SP_LISTA_RESERVAS_ARMARIO'
and s.inst_id = a.inst_id
and s.sid = a.sid)
```

18- Gerar Trace

--Para verificar quais traces estão rodando:

```
select * from dba_enabled_traces
```

```
EXEC sys.dbms_system.set_sql_trace_in_session(SID,serial,TRUE);
```

```
EXEC DBMS_MONITOR.SESSION_TRACE_ENABLE(
session_id IN BINARY_INTEGER DEFAULT NULL,
serial_num IN BINARY_INTEGER DEFAULT NULL,
waits IN BOOLEAN DEFAULT TRUE,
binds IN BOOLEAN DEFAULT FALSE)
```

```
EXEC DBMS_MONITOR.SESSION_TRACE_ENABLE(SID,SERIAL,TRUE,TRUE)
```

```
EXEC DBMS_MONITOR.SESSION_TRACE_DISABLE(SID,SERIAL)
```

Exemplos:

```
EXEC DBMS_MONITOR.SESSION_TRACE_ENABLE(5055,1701,TRUE,TRUE);
```

```
EXEC DBMS_MONITOR.SESSION_TRACE_DISABLE(5055,1701);
```

TRACE DA SESSÃO CORRENTE

=====

```
ALTER SESSION SET sql_trace = true;
```

-- Se quiser coloque identificador no trace

```
ALTER SESSION SET tracefile_identifier = mysqltrace;
```

-- Para parar o trace da sessão corrente faça:

```
ALTER SESSION SET sql_trace = false;
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

TRACE DE OUTROS USUÁRIOS

=====

--Habilite o trace para seus processos selecionados

```
ALTER SYSTEM SET timed_statistics = true;
```

Execute a package: (como usuário sys)

```
SQL> execute dbms_system.set_sql_trace_in_session(8, 13607, true);
```

ATENÇÃO: Ver com usuário o tempo necessário para ficar gerando a trace

-- Desabilitando o trace na sessão

```
SQL> execute dbms_system.set_sql_trace_in_session(8,13607, false);
```

Veja o arquivo do trace em USER_DUMP_DEST:

```
select value from v$parameter where lower(name) = 'user_dump_dest'
```

```
$ cd $ORACLE_HOME/admin/bd_1/udump
```

TRACE MAIS COMPLETO:

=====

```
EXEC DBMS_System.Set_Ev(593, 2137, 10046, 12, '') sid=44
```

serial=2259

para parar o trace é o mesmo comando só tira o 12 por 0

TRACE DO BANCO DE DADOS INTEIRO

=====

Para executar um trace do Database inteiro execute

```
ALTER SYSTEM SET sql_trace = true SCOPE=MEMORY;
```

--Para PARAR o trace do banco inteiro

```
ALTER SYSTEM SET sql_trace = false SCOPE=MEMORY;
```

-- Veja o arquivo do trace está no diretório UDUMP o nome default é:

INSTANCE_PID_ora_TRACEID.trc where:

INSTANCE = is the name of the Oracle instance,

PID = is the operating system process ID (V\$PROCESS.OSPID); and

TRACEID = is a character string of your choosing.

Para tornar o trace inteligível:

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

=====

tkprof bipdmb1_ora_21734.trc bipdmb1_ora_21734.txt sort=fchela

Observação: sort=fchela serve para demonstrar ordenado por fetch em ordem decrescente.

CRIANDO TRACE PARA FAZER GERAR TRACE DA SESSÃO

=====

```
1- CREATE OR REPLACE TRIGGER SYS.set_trace
AFTER LOGON ON DATABASE
WHEN (USER like '&USERNAME')
DECLARE
    lcommand varchar(200);
BEGIN
    EXECUTE IMMEDIATE 'alter session set statistics_level=ALL';
    EXECUTE IMMEDIATE 'alter session set max_dump_file_size=UNLIMITED';
    EXECUTE IMMEDIATE 'alter session set events "10046 trace name context forever, level 12"';
    EXECUTE IMMEDIATE 'alter session set tracefile_identifier = 'arbor' || V_sid';
END set_trace;
/
```

--Modelo simples

```
create or replace trigger sys.set_trace after logon on database begin if user = 'ARBOR' then
execute immediate 'alter session set sql_trace=true';end if;
end;
```

--Observação: troca-se a variável &USERNAME pelo nome do usuário desejado.

TRACE AUTOMATIZADA POR TRIGGER

=====

--Esta trigger filtra por usuário, máquina e traz valores das binds.

--Caso a package dbms_support não esteja instalada faça rodando o script dbmssupp.sql que encontra-se em \$ORACLE_HOME/admin/rdbms

```
CREATE OR REPLACE TRIGGER SYS.set_trace
AFTER LOGON ON DATABASE
DECLARE
    v_sid    NUMBER;
    v_serial NUMBER;
    v_username VARCHAR2(30);
    v_machine VARCHAR2(64);
BEGIN
    SELECT USER INTO v_username FROM DUAL;
    IF v_username = 'ASAP'
    THEN
        SELECT SID, serial#, machine
        INTO v_sid, v_serial, v_machine
        FROM v$session
        WHERE audsid = SYS_CONTEXT ('USERENV', 'SESSIONID');
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
--
-- Captura machine
--
IF UPPER(v_machine) LIKE '%SV2%'
THEN
    SYS.dbms_support.start_trace_in_session (v_sid, v_serial, TRUE, TRUE);
END IF;
END IF;
END;
/
```

19- verifica o tamanho do BD (físico)

--Separado por datafile

```
select substr(file_name,1,200) as "FileSystem", sum(bytes)/1024/1024 as "Tamanho(MB)"
from dba_data_files
group by rollup(substr(file_name,1,200))
order by substr(file_name,1,200)
```

--Tamanho total

```
select sum(bytes)/1024/1024 as "TamanhoFísico(MB)" from dba_data_files;
```

---Verificando total:

```
select ROUND( ( a.data_size + b.temp_size +
                c.redo_size + d.cf_size +
                e.bct_size)
              /1024/1024/1024
              ) "total_GB_size"
from
( select SUM(bytes) data_size      from v$datafile) a,
( select NVL(sum(bytes),0) temp_size    from v$tempfile) b,
( select SUM(bytes) redo_size      from v$log) c,
( select SUM(block_size*file_size_blks) cf_size from v$controlfile) d,
( select NVL(bytes,0) bct_size      from v$block_change_tracking) e;
```

20- verifica valor alocado para bd por tablespace

-- Datafiles

```
select tablespace_name, sum(bytes)/1024/1024/1024 "Valor alocado em GB" from
dba_data_files group by rollup(tablespace_name)
```

--Tablespace temporary

```
select tablespace_name, sum(bytes)/1024/1024 "Valor alocado em MB" from dba_temp_files
group by tablespace_name
```

-- Undo

```
select sum(bytes)/1024/1024/1024 "Valor alocado em GB" from dba_undo_extents
```

#####20.2- Verificando total do banco incluindo controlfiles, redolog, temp e datafiles

```
select ROUND( ( a.data_size + b.temp_size +
               c.redo_size + d.cf_size +
               e.bct_size)
             /1024/1024/1024
            ) "total_GB_size"
from
( select SUM(bytes) data_size      from v$datafile) a,
( select NVL(sum(bytes),0) temp_size    from v$tempfile) b,
( select SUM(bytes) redo_size      from v$log) c,
( select SUM(block_size*file_size_blks) cf_size from v$controlfile) d,
( select NVL(bytes,0) bct_size      from v$block_change_tracking) e;
```

21- Verifica realmente utilizado pelo BD por tablespace

```
select tablespace_name, sum(bytes)/1024/1024/1024 "valor utilizado em GB" from
dba_segments group by rollup (tablespace_name)
```

--dá pra usar where owner = 'owner' and segment_name= 'nome_tabela' para saber quanto espaço uma tabela está utilizando

-----variação para mostrar quanto tem livre, percentualmente-----

```
SELECT d.status "status",
       SUBSTR (d.tablespace_name, 1, 15) "name",
       d.CONTENTS "type",
       d.extent_management "extent",
       TO_CHAR (NVL (a.BYTES / 1024 / 1024, 0), '99,999,990.900') "size (m)",
       TO_CHAR ((NVL (a.BYTES, 0) - NVL (a.BYTES - NVL (f.BYTES, 0), 0)) / 1024 / 1024,
'99,999,990.900') "free",
       TO_CHAR (NVL (a.BYTES - NVL (f.BYTES, 0), 0) / 1024 / 1024, '99,999,990.900') "used",
       TO_CHAR (NVL ((a.BYTES - NVL (f.BYTES, 0)) / a.BYTES * 100, 0), '990.00') "used %"
FROM SYS.dba_tablespaces d,
     (SELECT tablespace_name,
            SUM (BYTES) BYTES
      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(+)
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```

AND d.tablespace_name = f.tablespace_name(+)
AND NOT ( d.extent_management LIKE 'LOCAL'
          AND d.CONTENTS LIKE 'TEMPORARY')
UNION ALL
SELECT d.status,
       SUBSTR (d.tablespace_name, 1, 15),
       d.CONTENTS,
       d.extent_management,
       TO_CHAR (NVL (a.BYTES / 1024 / 1024, 0), '99,999,990.900'),
       TO_CHAR ((NVL (a.BYTES, 0) - NVL (t.BYTES, 0)) / 1024 / 1024, '99,999,990.900'),
       TO_CHAR (NVL (t.BYTES, 0) / 1024 / 1024, '99,999,990.900'),
       TO_CHAR (NVL (t.BYTES / a.BYTES * 100, 0), '990.00')
FROM SYS.dba_tablespaces d,
     (SELECT tablespace_name,
              SUM (BYTES) BYTES
        FROM dba_temp_files
       GROUP BY tablespace_name) a,
     (SELECT tablespace_name,
              SUM (bytes_cached) 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 = 'LOCAL'
AND d.CONTENTS = 'TEMPORARY'

```

#####21.1- Script TOAD detalha tablespace

```

select a.tablespace_name,
       round(a.bytes_alloc / 1024 / 1024, 2) megs_alloc,
       round(nvl(b.bytes_free, 0) / 1024 / 1024, 2) megs_free,
       round((a.bytes_alloc - nvl(b.bytes_free, 0)) / 1024 / 1024, 2) megs_used,
       round((nvl(b.bytes_free, 0) / a.bytes_alloc) * 100,2) Pct_Free,
       100 - round((nvl(b.bytes_free, 0) / a.bytes_alloc) * 100,2) Pct_used,
       round(maxbytes/1048576,2) Max
from ( select f.tablespace_name,
              sum(f.bytes) bytes_alloc,
              sum(decode(f.autoextensible, 'YES',f.maxbytes,'NO', f.bytes)) maxbytes
        from dba_data_files f
       group by tablespace_name) a,
     ( select f.tablespace_name,
              sum(f.bytes) bytes_free
        from dba_free_space f
       group by tablespace_name) b
where a.tablespace_name = b.tablespace_name (+)
union all
select h.tablespace_name,
       round(sum(h.bytes_free + h.bytes_used) / 1048576, 2) megs_alloc,
       round(sum((h.bytes_free + h.bytes_used) - nvl(p.bytes_used, 0)) / 1048576, 2) megs_free,

```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle


```

round(sum(nvl(p.bytes_used, 0)) / 1048576, 2) megs_used,
round((sum((h.bytes_free + h.bytes_used) - nvl(p.bytes_used, 0)) / sum(h.bytes_used +
h.bytes_free)) * 100, 2) Pct_Free,
100 - round((sum((h.bytes_free + h.bytes_used) - nvl(p.bytes_used, 0)) / sum(h.bytes_used
+ h.bytes_free)) * 100, 2) pct_used,
round(sum(f.maxbytes) / 1048576, 2) max
from sys.v_$TEMP_SPACE_HEADER h, sys.v_$Temp_extent_pool p, dba_temp_files f
where p.file_id(+) = h.file_id
and p.tablespace_name(+) = h.tablespace_name
and f.file_id = h.file_id
and f.tablespace_name = h.tablespace_name
group by h.tablespace_name
ORDER BY 1

```

22- Saber valor que um owner ocupa no banco de dados efetivamente (nao espaço alocado) separado por tablespace

```

select tablespace_name, SUM(bytes)/1024/1024 "valores em MB" from dba_segments where
owner = 'ARBOR'
group by rollup (tablespace_name)

```

-- Variação server para verificar separando índice de tabela

```

SELECT segment_type,
SUM (BYTES) / 1024 / 1024 "valores em MB"
FROM dba_segments
WHERE segment_type IN ('TABLE', 'INDEX')
AND owner = '&OWNER'
GROUP BY ROLLUP (segment_type)

```

23- Verifica a alocação de determinada tabela separada por datafiles ou seja consumo de espaço da tabela por DF

```

select substr(D.file_name,1,200) as "FileSystem", sum(E.bytes)/1024/1024/1024 as
"TAMANHO EM GB"
from dba_data_files D,dba_extents E
where E.segment_name = 'NOME_TABELA'
and D.tablespace_name = 'NOME_TABLESPACE'
and D.FILE_ID = E.FILE_ID
group by rollup(substr(D.file_name,1,200))

```

-- variação verificando banco todo separado por file_system

```

select substr(D.file_name,1,200) as "FileSystem", sum(E.bytes)/1024/1024/1024 as
"TAMANHO EM GB"
from dba_data_files D,dba_segments E
where D.FILE_ID = E.FILE_ID
group by rollup(substr(D.file_name,1,200))

```

24- Saber valor que uma tabela ocupa no banco de dados efetivamente (nao espaço alocado) separado por tablespace

```
select tablespace_name, SUM(bytes)/1024/1024 "valores em MB" from dba_segments where  
segment_name = 'CDR_DATA'  
group by rollup (tablespace_name)
```

25- verifica entre duas tabelas quantos byte ainda falta inserir no caso de estar sendo feito insert ele mostra dinâmico

```
select a.segment_name,  
sum(a.bytes)/1024/1024 "Valor da tabela destino em MB",  
sum(b.bytes)/1024/1024 "Valor da tabela origem em MB",  
(sum(b.bytes)/1024/1024) - (sum(a.bytes)/1024/1024) "Falta inserir (em MB)"  
from dba_segments a, dba_segments@lk_migra b  
where a.segment_name = 'BILL_EQUIP_DETAIL'  
and a.owner = 'ARBOR'  
and b.owner = 'ARBOR9'  
and a.segment_name = b.segment_name  
group by a.segment_name, b.segment_name;
```

26- Lista execuções e loads por SQL_ID(Considerar ofensivos quando o número de loads for maior que executions)

```
select SQL_ID, SHARABLE_MEM/1024/1024, PERSISTENT_MEM/1024/1024, EXECUTIONS,  
LOADS, LOADED_VERSIONS, OPTIMIZER_MODE  
from gv$sql  
where executions > 5  
and loads > 5;
```

27- área ocupada pelos componentes da SGA

```
SELECT NAME, TRUNC(SUM(mb),3) total, TRUNC(SUM(inuse),3) in_use  
FROM (SELECT CASE  
        WHEN NAME = 'buffer_cache'  
        THEN 'db_cache_size'  
        WHEN NAME = 'log_buffer'  
        THEN 'log_buffer'  
        ELSE pool  
        END NAME,  
        BYTES/1024/1024 mb,  
        CASE  
        WHEN NAME = 'buffer_cache'  
        THEN (BYTES - (SELECT COUNT(*)  
                        FROM v$bh  
                        WHERE status = 'free')  
        * (SELECT VALUE  
           FROM v$parameter  
           WHERE NAME = 'db_block_size')  
        ) /1024/1024
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
        WHEN NAME <> 'free memory'
        THEN BYTES/1024/1024
    END inuse
    FROM v$sgastat)
GROUP BY NAME;
```

28- Geração de senha randômica

```
select dbms_random.string('U', 2) || trunc(dbms_random.value(1000, 9999))
from dual;
```

29- Verificando limites de banco de dados Sessions, process, etc

```
select * from v$resource_limit;
```

-- Variação

```
select *
from v$resource_limit
where resource_name in ('enqueue_resources', 'enqueue_locks',
'dml_locks', 'processes', 'sessions');
```

30- Agrupar usuários por tablespaces

```
select default_tablespace, temporary_tablespace, profile, count(*)
from dba_users
where username like '%VB'
group by default_tablespace, temporary_tablespace, profile
order by 4 desc;
```

31- verifica a quantidade de REDO gerado por Sessão

```
select ss.sid, ss.value
from v$sesstat ss, v$statname sn
where sn.statistic# = ss.statistic#
and sn.name = 'redo size'
order by 2 desc;
```

32- Resolver problema com transação distribuída pendente

```
declare
cursor cv_pending is
    SELECT * FROM DBA_2PC_PENDING
    WHERE fail_time < (SYSDATE-5/1440);
begin
    FOR x IN cv_pending LOOP
        DBMS_OUTPUT.PUT_LINE("");
        DBMS_OUTPUT.PUT_LINE('rollback force "' || x.local_tran_id || '"');
        DBMS_OUTPUT.PUT_LINE('COMMIT;');
        DBMS_OUTPUT.PUT_LINE('EXECUTE DBMS_TRANSACTION.PURGE_LOST_DB_ENTRY("' ||
x.local_tran_id || '"');
    END LOOP;
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
DBMS_OUTPUT.PUT_LINE('COMMIT;');
END LOOP;
end;
/
```

--Com esta PI verifica-se quais as transações e elimina elas.

33- mostra as conexões de entrada e saída para conexões pendentes

```
select * from DBA_2PC_NEIGHBORS
```

34- Mostra as alternâncias de log (switch log)

```
select thread#,
       to_char(first_time,'DD-MON-YYYY') creation_date,
       to_char(first_time,'HH24:MI:SS') time,
       sequence#,
       first_change# lowest_SCN_in_log,
       next_change# highest_SCN_in_log,
       recid controlfile_record_id,
       stamp controlfile_record_stamp
from v$log_history
order by sequence#;
```

-- Variação:

```
SELECT v.group# Grupo      ,
       l.status             ,
       v.type               ,
       v.member             ,
       l.sequence# Sequencia ,
       ROUND(l.bytes/1024/1024,2) MB,
       l.members            ,
       l.archived            ,
       l.first_change#      ,
       TO_CHAR(l.first_time, 'dd/mm/yyyy hh24:mi:ss') "Primeira Vez"
FROM v$logfile v
INNER JOIN v$log l
ON l.group# = v.group#
ORDER BY 1;
```

#####34.2- Informações dos redolog files #####

--dados de tamanho e qual é o corrente

```
select * from v$log;
```

--Localização física, etc.

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
select * from v$logfile
```

Observação importante:

Redo faz rollforward e undo faz rollback

35- Recompilar todos objetos inválidos

Rodar o script:

\$ORACLE_HOME/rdbms/admin rodar o utlrp que recompila todos os objetos inválidos

```
SQL> @$ORACLE_HOME/rdbms/admin/utlrp.sql;
```

36- Verifica a utilização da área de recuperação Flash DB_RECOVERY_FILE_DEST

```
SELECT file_type, percent_space_used AS utilizado, percent_space_reclaimable AS obsoleto,  
       number_of_files AS "numero de arquivos"  
FROM v$flash_recovery_area_usage;
```

---Variação

```
select r.name, space_limit/1024/1024 as space_limit_mb,  
(r.space_limit - r.space_used + r.space_reclaimable)/1024/1024 as space_available_mb,  
round ((r.space_used - r.space_reclaimable) / r.space_limit * 100,1) as percent_full,  
a.total_mb as total_diskgroup_mb, a.free_mb as free_diskgroup_mb  
from v$recovery_file_dest r, v$asm_diskgroup a  
where trim('+ ' from r.name) = a.name;
```

37- Trabalhando com segmento de rollback

Verifica a utilização:

```
select name, waits, gets  
from v$rollstat, v$rollname  
where v$rollstat.usn = v$rollname.usn;
```

--Verifica o tamanho 8i:

```
SELECT owner, tablespace_name, segment_name,  
       ROUND ((BYTES / 1024 / 1024), 2) mb, extents,  
       initial_extent / 1024 / 1024 initial_extent_mb,  
       next_extent / 1024 / 1024 next_extent_mb, max_extents  
FROM dba_segments  
WHERE segment_type LIKE '%ROL%'  
ORDER BY 4 DESC
```

-- Verifica o tamanho 10g:

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
select owner,tablespace_name,segment_name,round((bytes/1024/1024),2)
mb,extents,initial_extent/1024/1024 initial_extent_mb,
       next_extent/1024/1024 next_extent_mb,max_extents
from dba_segments
where segment_type like '%UND%'
--order by tablespace_name,segment_name
ORDER BY round((bytes/1024/1024),2) DESC
```

--Shrink:

```
ALTER ROLLBACK SEGMENT rbs_name SHRINK
```

-- Forçar uma sessão a usar um determinado rollback segment:

```
SET TRANSACTION USE ROLLBACK SEGMENT SEGMENT_NAME;
```

-- Verificar quem está usando os segmentos de rollback

```
SELECT  r.segment_name,
        r.tablespace_name,
        s.SID,
        s.serial#,
        s.username,
        s.machine,
        t.status,
        t.cr_get, /*consistent gets*/
        t.phy_io, /*physical io*/
        t.used_ublk, /*Undo blocks used*/
        t.noundo, /*Is a noundo transaction*/
        SUBSTR (s.program, 1, 78) "COMMAND",
        s.username "DB User",
        t.start_time,
        s.sql_address "Address",
        s.sql_hash_value "Sql Hash"
FROM SYS.v_$session s, SYS.v_$transaction t, SYS.dba_rollback_segs r
WHERE t.addr = s.taddr
      AND t.xidusn = r.segment_id
ORDER BY t.start_time
```

38- startar queue Oracle (AQ)

```
BEGIN
SYS.DBMS_AQADM.START_QUEUE
(
  QUEUE_NAME => 'nome_queue'
,ENQUEUE => TRUE
,DEQUEUE => TRUE
);
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

END;

39- Alterar Profile

Alter profile profile_name limit resource_name valor

```
ALTER PROFILE SQL_PROF LIMIT SESSION_PER_USER 10;
```

40- Caso o log do Alert ou do Listener cresça muito e precise zerá-lo

1- tail -10000 listener.log > listener.log_old (para copiar para um novo log os últimos 10000 registros);

2- echo > listener.log (para zerar o arquivo de log do Listener)

--também serve para o alert log

```
tail -10000 alert_qa1ct2.log > alert_qa1ct2_bkp.log
```

```
echo > alert_qa1ct2.log
```

41- criando dblink com RAC.

```
CREATE DATABASE LINK nome_dblink CONNECT TO USUARIO IDENTIFIED BY PASS_usuario  
USING 'NOME_NO_tnsnames';
```

Observação Interessante: caso a base que acesse seja RAC deve-se alterar em todos os nós o tnsnames para acessar a base destino.

Segue abaixo exemplo de uma base destino também RAC:

SVCPOIM =

```
(DESCRIPTION =  
  (ADDRESS = (PROTOCOL = TCP)(HOST = svuxcrs1)(PORT = 1521))  
  (ADDRESS = (PROTOCOL = TCP)(HOST = svuxcrs2)(PORT = 1521))  
  (LOAD_BALANCE = yes)  
  (CONNECT_DATA =  
    (SERVER = DEDICATED)  
    (SERVICE_NAME = nome_service)  
    (FAILOVER_MODE =  
      (TYPE = SELECT)  
      (METHOD = BASIC)  
      (RETRIES = 180)  
      (DELAY = 5)  
    )  
  )  
)
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

)

Neste caso o USING do dblink iria chamar o serviço SVCPOIM

42- Criando sinônimo

-- Público

create public synonym nome_synonym for owner.tabela;

-- Privado

create synonym owner.synonym_name for owner.tabela;

44- Verifica espaço livre em discos ASM

```
select name, total_mb, free_mb,
round((free_mb)/total_mb * 100,1) as percent_free
from Gv$asm_diskgroup
where round((free_mb)/total_mb * 100,1) <=10
order by 1,4;
```

```
select name, total_mb, free_mb,
round((free_mb)/total_mb * 100,1) as percent_free
from v$asm_diskgroup
order by 1,4;
```

44.1-Verificando diferença de alocado alocado para tablespace e livre

```
SELECT NAME,
SUM (total_mb) / 1024 AS "Total disk groups (Gb)",
(SUM (total_mb) / 1024 - SUM (free_mb) / 1024) AS "Alocado p/ tablespaces (Gb)",
SUM (free_mb) / 1024 "Valor livre para alocar (Gb)"
FROM v$asm_diskgroup
GROUP BY ROLLUP (NAME);
```

45- Mostra o crescimento do BD listado por mês (para 6 meses) totalizando no final.

```
Select to_char(creation_time, 'RRRR Month') "Month", sum(bytes)/1024/1024 "Growth in
Meg"
from sys.v_$datafile where creation_time > SYSDATE-180
group by rollup (to_char(creation_time, 'RRRR Month'))
```


-- separando/demonstrando por Tablespace

```
Select t.tablespace_name, to_char(creation_time, 'RRRR Month') "Month",  
sum(v.bytes)/1024/1024/1024 "Growth in GB"  
from sys.v_$datafile v, dba_data_files t where creation_time > SYSDATE-180  
and t.FILE_ID=v.FILE#  
group by rollup (t.tablespace_name, to_char(creation_time, 'RRRR Month'))
```

-- considerando o TEMP

```
Select t.tablespace_name, to_char(creation_time, 'RRRR Month') "Month",  
sum(v.bytes)/1024/1024/1024 "Growth in GB"  
from sys.v_$tempfile v, dba_temp_files t where creation_time > SYSDATE-180  
and t.FILE_ID=v.FILE#  
group by rollup (t.tablespace_name, to_char(creation_time, 'RRRR Month'))
```

46- Verificando asm disks file system, nome físico, etc

```
select * from v$asm_disk
```

--variação (obs rodar logado no ASM)

```
SELECT name, header_status, LABEL, CREATE_DATE, path, TOTAL_MB, FREE_MB FROM  
V$ASM_DISK
```

47- Alocação de discos no diskgroup e adicionar datafiles ASM

--conectado na instância do ASM

```
ALTER DISKGROUP DISK_GROUP_NAME ADD DISK '/dev/oracleasm/disks/nome_disco' size  
25G;
```

Ex:

```
ALTER DISKGROUP PSIG_DATA_01 ADD DISK '/dev/oracleasm/disks/EMCCW1_PSIG_DATA03'  
size 25G;
```

```
ALTER DISKGROUP QA1CT2_DATA_01 ADD DISK '/dev/asm_qa1ct2_power1';
```

-- Adicionando datafiles ASM

```
alter tablespace ARBOR_DATA add datafile '+QA1CT1_DATA_01(datafile)' size 2G;
```

--Rebalance

```
ALTER DISKGROUP DISK_GROUP_NAME REBALANCE
```

--Fazendo resize em datafile:

a) pegar o nome do data file a fazer o resize:

```
select FILE_NAME, (bytes)/1024/1024/1024 "Tamanho do DF em GB" from dba_data_files
where tablespace_name='SAVVION_DATA'
```

b) Proceder o resize:

```
ALTER DATABASE DATAFILE '+PSAV_DATA_01/psav/datafile/savvion_data.309.746118973'
RESIZE 20480M;
```

c) dismantlar disco:

logado como grid

sqlplus / as sysasm

```
alter diskgroup DG_CONTROL dismount;
```

#####47.2- ver operação no asm (rebalance) (obs: rodar logado no ASM) #####

```
select group_number, operation, state, power, actual, sofar, est_work, est_rate, est_minutes
from V$ASM_OPERATION;
```

48- Movimentacao de I/O

```
select sess_io.sid,
       sesion.username,
       sesion.osuser,
       sess_io.block_gets,
       sess_io.consistent_gets,
       sess_io.physical_reads,
       sess_io.block_changes,
       sess_io.consistent_changes
from v$sess_io sess_io, v$session sesion
where sesion.sid = sess_io.sid
-- and sesion.username = '&&USERNAME'
```

49- verificação de tempo de banco pelo SID (tempo em microssegundos)

```
select * from V$SESS_TIME_MODEL where SID = '&&SID'
and value <> '0'
```

51- Mostra quais sessões estão consumindo mais CPU

```
SELECT s.sid, s.serial#, p.spid as "OS PID", s.username, s.module, st.value/100 as "CPU sec"
FROM v$sesstat st, v$statname sn, v$session s, v$process p
WHERE sn.name = 'CPU used by this session' -- CPU
AND st.statistic# = sn.statistic#
AND st.sid = s.sid
AND s.paddr = p.addr
AND s.last_call_et < 1800 -- active within last 1/2 hour
AND s.logon_time > (SYSDATE - 240/1440) -- sessions logged on within 4 hours
ORDER BY st.value desc;
```

-- para RAC

```
SELECT s.inst_id, s.sid, s.serial#, p.spid as "OS PID", s.username, s.module, st.value/100 as "CPU
sec"
FROM gv$sesstat st, gv$statname sn, gv$session s, gv$process p
WHERE sn.name = 'CPU used by this session' -- CPU
AND st.statistic# = sn.statistic#
AND st.sid = s.sid
AND s.paddr = p.addr
and s.inst_id = st.inst_id
and st.inst_id = p.inst_id
and s.inst_id = sn.inst_id
AND s.last_call_et < 1800 -- active within last 1/2 hour
AND s.logon_time > (SYSDATE - 240/1440) -- sessions logged on within 4 hours
ORDER BY st.value desc;
```

--Problema com CPU

top na shell do SO.

e veja qual processo ta comendo maior %

pegou o pid do cara roda essa query aqui que voce vai pegar a sessão dele !

```
SELECT SID,
       serial#,
       username,
       program,
       sql_hash_value,
       osuser,
       machine,
       taddr,
       lockwait,
       status
FROM v$session
WHERE paddr = (SELECT addr FROM v$process WHERE spid = '&pid')
```

--Tendo a sessão agora é só pegar a query que ta rodando:

```
select
  u.sid,
  substr(u.username,1,12) user_name,
  s.sql_text
from
  v$sqltext s,
  v$session u
where
  s.hash_value = u.sql_hash_value
and sql_text not like '%from v$sqltext s, v$session u%'
and u.sid = '&SID'
order by u.sid;
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

--Fazer um explain plan nela e ver porque ta tão ruim.

```
EXPLAIN PLAN FOR
&SQL_STATEMENT
```

--Verificar o plano de execução:

```
@showplan.sql
```

-- que é:

```
set pages 1000
```

```
set lines 120
```

```
select * from table(dbms_xplan.display)
```

```
/
```

52- Encontra quem está utilizando a tablespace temporária (TEMP)

```
SELECT b.TABLESPACE, b.segfile#, b.segblk#, b.blocks, a.SID, a.serial#,
       a.username, a.osuser, a.status
FROM v$session a, v$sort_usage b
WHERE a.saddr = b.session_addr
ORDER BY b.TABLESPACE, b.segfile#, b.segblk#, b.blocks;
```

```
SELECT sysdate, a.username, a.sid, a.serial#, a.osuser, b.blocks, c.sql_text
FROM v$session a, v$sort_usage b, v$sqlarea c
WHERE b.tablespace = 'TEMP'
and a.saddr = b.session_addr
AND c.address= a.sql_address
AND c.hash_value = a.sql_hash_value
-- AND b.blocks* 8192 > [metrica]
ORDER BY b.tablespace, b.blocks;
```

53- encontrar as 10 tabelas que mais fazem I/O no BD - Oracle 10G

```
select table_name, total_phys_io
from ( select owner || '.' || object_name as table_name,
       sum(value) as total_phys_io
       from v$segment_statistics
       where owner != 'SYS' and object_type = 'TABLE'
       and statistic_name in ('physical reads', 'physical reads direct',
                              'physical writes', 'physical writes direct')
       group by owner || '.' || object_name
       order by total_phys_io desc)
where rownum <= 10;
```

-- POR TABELA:

```
select *
from ( select owner || '.' || object_name as table_name, statistic_name,
       sum(value) as total_phys_io
       from v$segment_statistics
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
where owner!='SYS' and object_type='TABLE'
and object_name='GTT_P13_NFRESUMO'
group by owner||'.'||object_name, statistic_name
order by total_phys_io desc)
where rownum <=10;
```

-- Variação

```
SELECT a.object_name,
       a.object_type,
       b.*
FROM dba_objects a, v$segstat b
WHERE b.statistic_name LIKE 'physi%rea%'
AND a.object_id = b.obj#
ORDER BY b.VALUE DESC
```

54- Encontra sessões ativas executando queries com o maior tempo

```
SELECT inst_id, SID, USERNAME, MACHINE, LOGON_TIME, LAST_CALL_ET
FROM gv$session
WHERE username IS NOT NULL
AND USERNAME NOT IN ('SYS','SYSTEM')
AND status = 'ACTIVE'
AND last_call_et > ( 3600 * 2) -- Maior que 2 horas
```

55- Encontra sessões com o maior tempo de uso de banco

```
SELECT s.sid, s.serial#, p.spid as "OS PID", s.username, s.module, st.value/100 as "DB Time
(sec)"
, stcpu.value/100 as "CPU Time (sec)", round(stcpu.value / st.value * 100,2) as "% CPU"
FROM v$sesstat st, v$statname sn, v$session s, v$sesstat stcpu, v$statname sncpu, v$process
p
WHERE sn.name = 'DB time' -- CPU
AND st.statistic# = sn.statistic#
AND st.sid = s.sid
AND sncpu.name = 'CPU used by this session' -- CPU
AND stcpu.statistic# = sncpu.statistic#
AND stcpu.sid = st.sid
AND s.paddr = p.addr
AND s.last_call_et < 1800 -- active within last 1/2 hour
AND s.logon_time > (SYSDATE - 240/1440) -- sessions logged on within 4 hours
AND st.value > 0;
```

56- Verifica o tempo de backup de cada bd cadastrado no catálogo do RMAN

--Observação efetuar conectado no catálogo

```
SELECT start_time, end_time, status, input_type,
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
ROUND (elapsed_seconds / 60) elapsed_minutes  
FROM V$RMAN_BACKUP_JOB_DETAILS  
WHERE TRUNC (start_time) = TRUNC (SYSDATE)
```

#####57- Verificações básicas RAC

```
crsctl check crs - checks the viability of the CRS stack  
crsctl check cssd - checks the viability of CSS  
crsctl check crsd - checks the viability of CRS  
crsctl check evmd - checks the viability of EVM  
crsctl set css <parameter> <value> - sets a parameter override  
crsctl get css <parameter> - gets the value of a CSS parameter  
crsctl unset css <parameter> - sets CSS parameter to its default  
crsctl query css votedisk - lists the voting disks used by CSS  
crsctl add css votedisk <path> - adds a new voting disk  
crsctl delete css votedisk <path> - removes a voting disk  
crsctl enable crs - enables startup for all CRS daemons  
crsctl disable crs - disables startup for all CRS daemons  
crsctl start crs - starts all CRS daemons.  
crsctl stop crs - stops all CRS daemons. Stops CRS resources in case of cluster.  
crsctl start resources - starts CRS resources.  
crsctl stop resources - stops CRS resources.  
crsctl debug statedump evm - dumps state info for evm objects  
crsctl debug statedump crs - dumps state info for crs objects  
crsctl debug statedump css - dumps state info for css objects  
crsctl debug log css module:level{,module:level} ...  
- Turns on debugging for CSS  
crsctl debug trace css - dumps CSS in-memory tracing cache  
crsctl debug log crs module:level{,module:level} ...  
- Turns on debugging for CRS  
crsctl debug trace crs - dumps CRS in-memory tracing cache  
crsctl debug log evm module:level{,module:level} ...  
- Turns on debugging for EVM  
crsctl debug trace evm - dumps EVM in-memory tracing cache  
crsctl debug log res <resname:level> turns on debugging for resources  
crsctl query crs softwareversion <nodename> - lists the version of CRS software installed  
crsctl query crs activeversion - lists the CRS software operating version  
crsctl lsmodules css - lists the CSS modules that can be used for debugging  
crsctl lsmodules crs - lists the CRS modules that can be used for debugging  
crsctl lsmodules evm - lists the EVM modules that can be used for debugging
```

Para baixar e subir instâncias

==== SHUTDOWN ====

```
./oracle/admin/scripts/tbsrac_env.sh ## Set de ambiente
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
srvctl status database -d ptbs -v
```

```
srvctl stop database -d ptbs -o immediate
```

```
srvctl stop instance -d ptbs -i ptbs2
```

```
srvctl stop instance -d ptbs -i ptbs1
```

```
. /oracle/admin/scripts/env_asm.sh
```

```
srvctl stop asm -n svuxpbs2
```

```
srvctl stop asm -n svuxpbs1
```

```
==== STARTUP ====
```

```
. /oracle/admin/scripts/env_asm.sh
```

```
srvctl start asm -n svuxpbs2
```

```
srvctl start asm -n svuxpbs1
```

```
. /oracle/admin/scripts/tbsrac_env.sh
```

```
srvctl start database -d ptbs
```

```
srvctl start instance -d ptbs -i ptbs1
```

```
srvctl start instance -d ptbs -i ptbs2
```

58- Verificação do plano de execução

1- EXECUTAR

```
select address, hash_value, child_number, sql_text  
from v$sql  
where users_executing <> 0
```

Observação:

-- users_executing é a quantidade de usuários executando o comando

2- COM OS VALORES EXTRAÍDOS DO SCRIPT ACIMA EXECUTE

-- script pedirá os valores conseguidos acima.

SET VERIFY OFF

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
COL id          FORMAT 999
COL parent_id   FORMAT 999 HEADING "PARENT"
COL operation    FORMAT a35 TRUNCATE
COL object_name FORMAT a30
SELECT id, parent_id, LPAD ('', LEVEL - 1) ||
       operation || '' || options operation, object_name
FROM (
  SELECT id, parent_id, operation, options, object_name
  FROM   v$sql_plan
  WHERE  address = '&address'
  AND    hash_value = &hash_value
  AND    child_number = &child_number
)
START WITH id = 0
CONNECT BY PRIOR id = parent_id;
```

59- Liberando espaço com Shrink

```
ALTER TABLE <TABLE_NAME> SHRINK SPACE
```

```
ALTER INDEX <INDEX_NAME> SHRINK SPACE [COMPACT]
```

60- Verifica sessões, e username que estão utilizando determinado objeto

```
select /*+ rule */ a.SID, s.username from gv$access a, gv$session s
where object = '&NOME_OBJETO'
and s.inst_id = a.inst_id
and s.sid = a.sid
```

61- Criando índice exemplo

```
CREATE INDEX TESTE_ARCH.TESTE_DETAIL_IDX_05
ON TESTE_ARCH.TESTE_DETAIL_2016
(ACCOUNT_NO, BILL_REF_NO, BILL_REF_RESETS)
nologging
tablespace TESTE_ARCH_JNL1_IDX
PCTFREE 10
INITRANS 2
MAXTRANS 255
STORAGE (
  INITIAL      128M
  NEXT         512M
  MINEXTENTS   1
  MAXEXTENTS   UNLIMITED
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle


```
PCTINCREASE 0
FREELISTS 1
FREELIST GROUPS 1
BUFFER_POOL DEFAULT
)
PARALLEL 8;
```

62- Verificar onde usuários estão conectando

```
select inst_id, username, count(*) from gv$session
where status = 'ACTIVE'
group by inst_id, username
order by inst_id
```

63- corrigir erro de flashback

Verificar conectado na instância o parâmetro de recovery
Show parameter db_recovery_file

Conecta na instância que tem o asm

Seta ORACLE_SID=+ASM1 (exemplo)

acessar o asmcmd

Ir até o local onde está a área de flashback listada acima no parâmetro.

64- Criação de Tablespace

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
--Pequenos
CREATE TABLESPACE TESTE_DATA2 DATAFILE '/oradata/qabil2/data01/TESTE_DATA2_01.dbf'
SIZE 2048M AUTOEXTEND off
LOGGING
ONLINE
PERMANENT
EXTENT MANAGEMENT LOCAL UNIFORM SIZE 1M
STORAGE (    INITIAL    1M
          NEXT      1M
        )
```

---Pode-se ainda utilizar

```
NOLOGGING
ONLINE
PERMANENT
EXTENT MANAGEMENT LOCAL UNIFORM SIZE 100M;
```

65- Movendo tabela entre tablespaces

```
ALTER TABLE owner.table_name MOVE TABLESPACE tablespace_name --OPCIONAL: storage
(initial 1M next 1M);
```

66- verifica Laudo do ADDM

```
Select a.execution_end, b.type, b.impact, d.rank, d.type,
'Message      : '||b.message MESSAGE,
'Command To correct: '||c.command COMMAND,
'Action Message : '||c.message ACTION_MESSAGE
From dba_advisor_tasks a, dba_advisor_findings b,
Db_a_advisor_actions c, dba_advisor_recommendations d
Where a.owner=b.owner and a.task_id=b.task_id
And b.task_id=d.task_id and b.finding_id=d.finding_id
And a.task_id=c.task_id and d.rec_id=c.rec_id
And a.task_name like 'ADDM%' and a.status='COMPLETED'
Order by b.impact, d.rank
```

66.2- Criando report ADDM com DBMS_ADVISOR#####

DBMS_ADVISOR

The DBMS_ADVISOR package can be used to create and execute any advisor tasks, including ADDM tasks. The following example shows how it is used to create, execute and display a typical ADDM report.

```
BEGIN
```

```
-- Create an ADDM task.
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
DBMS_ADVISOR.create_task (  
  advisor_name => 'ADDM',  
  task_name    => '970_1032_AWR_SNAPSHOT',  
  task_desc    => 'Advisor for snapshots 970 to 1032.');
```

-- Set the start and end snapshots.

```
DBMS_ADVISOR.set_task_parameter (  
  task_name => '970_1032_AWR_SNAPSHOT',  
  parameter => 'START_SNAPSHOT',  
  value     => 970);
```

```
DBMS_ADVISOR.set_task_parameter (  
  task_name => '970_1032_AWR_SNAPSHOT',  
  parameter => 'END_SNAPSHOT',  
  value     => 1032);
```

-- Execute the task.

```
DBMS_ADVISOR.execute_task(task_name => '970_1032_AWR_SNAPSHOT');  
END;  
/
```

-- Display the report.

```
SET LONG 100000
```

```
SET PAGESIZE 50000
```

```
SELECT DBMS_ADVISOR.get_task_report('970_1032_AWR_SNAPSHOT') AS report  
FROM dual;
```

```
SET PAGESIZE 24
```

The value for the SET LONG command should be adjusted to allow the whole report to be displayed.

The relevant AWR snapshots can be identified using the DBA_HIST_SNAPSHOT view.

Related Views

The following views can be used to display the ADDM output without using Enterprise Manager or the GET_TASK_REPORT function.

- DBA_ADVISOR_TASKS - Basic information about existing tasks.
- DBA_ADVISOR_LOG - Status information about existing tasks.
- DBA_ADVISOR_FINDINGS - Findings identified for an existing task.
- DBA_ADVISOR_RECOMMENDATIONS - Recommendations for the problems identified by an existing task.

Rodando ADDM com SCRIPT

-- UNIX

```
@/u01/app/oracle/product/10.1.0/db_1/rdbms/admin/addmrpt.sql
```

-- Windows

```
@d:\oracle\product\10.1.0\db_1\rdbms\admin\addmrpt.sql
```

67- Ver se alguem ficou preso no backup (status active)

```
SELECT d.name, b.status
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
FROM v$datafile d, v$backup b
WHERE d.FILE#=b.FILE#
```

68- Apagar backups obsoletos e fazer crosscheck

--Via script

```
run
{
  allocate channel ch1 device type disk;
  crosscheck archivelog all;
  crosscheck backup;
  delete noprompt expired archivelog all;
  delete noprompt expired backup;
  delete noprompt obsolete;
  release channel ch1;
}
```

-- Conectando no catálogo pra fazer o crosscheck

Setar as variáveis de ambiente da instância

```
connect catalog usuario/senha@rcat
connect target
allocate channel for maintenance type disk;
configure retention policy to recovery window of 15 days;
change archivelog all crosscheck;
delete noprompt expired archivelog all;
```

69- Erro na v\$recovery_file_dest

1- shutdown na instância

2- startup mount;

3- alter database flashback on

4- alter database open

5- Verificar se existe restore point garantidos com

```
select name,scn,time,database_incarnation#,guarantee_flashback_database,storage_size from
v$restore_point;
```

6- Caso haja remove-los com DROP RESTORE POINT NOME_POINT

7- shutdown na instância

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

8- startup mount;

9- alter database flashback off

10- caso haja logs de flashback podem ser excluídos

11- alter database open;

70- Fazer insert as select com hint de append e parallel

1- ALTER SESSION ENABLE PARALLEL DML;

2- insert /*+ append parallel(a,4) */ into arbor.bill_equip_detail a select /*+ parallel(b,4) full(b) */ * from arbor9.bill_equip_detail b;

71- Verificar quantidade de conexões por instância RAC

select count(*) as conectados, inst_id from gv\$session where status = 'ACTIVE' group by inst_id;

#####72- Trace do Listener e do SQL_NET #####

1.Habilitando trace do Listener

LSNRCTL> set trc_level ADMIN

1.1 Desabilitando

LSNRCTL> set trc_level OFF

1.2 Formatando o trace:

trcasst [option] listener.trc

--LOG do LISTENER localiza-se em:
/oracle/product/10.2.0/db_1/network/log

2. Habilitando o SQL_NET TRACE

(1) Open the SQLNET.ORA file typically found in the following location:

ORACLE_BASE\ORACLE_HOME\Network\Admin

(2) Add the following parameters at the end of the file:

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
#CLIENT-SIDE SQL*NET TRACE PARAMETERS
#=====
TRACE_UNIQUE_CLIENT = ON
TRACE_LEVEL_CLIENT = 16
TRACE_DIRECTORY_CLIENT = C:\temp
TRACE_FILE_CLIENT = SQLNetTrace
TRACE_TIMESTAMP_CLIENT = ON
#TRACE_FILELEN_CLIENT = 2048
#TRACE_FILENO_CLIENT = 2
```

Parameters prefixed with a “#” are interpreted as comments and will not affect tracing.

If you want to change the location where the trace file will be saved modify the following two parameters:

(a) TRACE_DIRECTORY_CLIENT

Set this parameter to the folder where you want the trace file to be saved.

WARNING: Do not end the path with a “\”. This is not necessary and may prevent the trace file from being generated.

(b) TRACE_FILE_CLIENT

Set this parameter to the filename of the trace file that will be created. The actual trace filename may contain additional information appended to the end of the filename such as

SQLNetTrace_<#>.trc

where is the application’s process id while it was running and <#> distinguishes different client connections established from the same process id (or session) to the database. This occurs since the parameter TRACE_UNIQUE_CLIENT is set to ON.

The parameter TRACE_LEVEL_CLIENT=16 ensures that maximum trace information is logged.

NOTE: If you think the size of the trace file will exceed the free space on the drive where TRACE_DIRECTORY_CLIENT is saving the file, then uncomment the last two parameters which will have the following affect:

(a) TRACE_FILELEN_CLIENT - will create trace files of the size specified in kilobytes (KB)

(b) TRACE_FILENO_CLIENT - will write to multiple trace files up

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

to the maximum size specified by
TRACE_FILELEN_CLIENT and then cycle
through the files again overwriting
previously written trace information

#####73- LINUX/AIX/HPUX #####

a) Mostrar a quantidade de processos total no servidor para o usuário logado

- a.1) smit chgsys
- a.2) lsattr -El sys0 -a maxuproc
- a.3) ulimit -a

Para saber do sistema logado como root
sysctl -a

a.4) Alterar a quantidade de processos do usuário logado
ulimit -

- c : Core file em blocos de 512 bytes
- d : Segmento de dados em Kbytes
- f : tamanho maximo de cada ficheiro em blocos de 512 bytes
- n : numero maximo de filedescriptor + 1
- s : Stack em Kbytes
- t : tempo de CPU em segundos
- v : tamanho maximo de memoria virtual em Kbytes. ?
- m : tamanho maximo de memoria
- u : numero maximo de processos
- p : tamanho para pipes em blocos de 512
- S : Impoe o soft limit
- H : Impoe o hard limit
- a : mostra os limits correntemente aplicados

a.5) Para alterar do SO.
edito /etc/sysctl.conf salvo e depois /sbin/sysctl -p

VERIFICAR NO /etc/security/limits.conf

b) Contar a quantidade de processos

- b.1) ps -fu oracle | wc -l --> mostra os processos oracle
- b.2) ps -ef | wc -l --> mostra todos os processos

c) localizar processos oracle local=no para kill
ps -fu oracle | grep -i local=no

d) PERMISSÕES:

=====

U G O

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

rwX rwX rwX = 111 111 111
rw- rw- rw- = 110 110 110
rwX --- --- = 111 000 000
U->User (utilizador) G->Group(grupo) O->Other(outros)

A relação dos números em binário com decimal:

rwX = 111 em binário = 7
rw- = 110 em binário = 6
r-x = 101 em binário = 5
r-- = 100 em binário = 4

e)VERIFICAR VERSÕES INSTALADAS PELO SO.

\$ORACLE_HOME/OPatch/patch lsinventory

f)Verificar e apagar arquivos de uma determinada data

find /home/ora/slice/base/load/blbill -mtime +90 -exec rm -f {} \;

g)verificar quantidade de processadores, etc no AIX

nmon

h)Verificar usuários no servidor:

cat /etc/passwd

i) Verificar consumidores de memória

\$ ipcs

\$ ipcs -rm

j)Verificar número de conexões pela porta 1521

\$netstat -Aan |grep 1521|wc -l

DIRETÓRIOS

Para os que estão se iniciando no Linux, conhecer os diretórios pode ser um pouco difícil, as vezes por causa da sua estrutura peculiar (eu diria, mais organizada) ou

por estarem migrando do "outro".

/ : raiz do sistema, o diretório que "guarda" todos os outros diretórios. É um "c: da vida". Uma dica é que ele não possua nenhum arquivo além dos

diretórios/arquivos (default) do sistema.

/bin : arquivos/comandos utilizados durante a inicialização do sistema e por usuários (após a inicialização)

/boot: arquivos utilizados durante a inicialização do sistema

/dev : drivers de controle de dispositivos

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

/etc : arquivos de configurações do computador

/etc/sysconfig : arquivos de configuração do sistema para os dispositivos.

/etc/passwd : dados dos usuários, senhas criptografadas...

/etc/fstab : sistemas de arquivos montados no sistema

/etc/group : grupos

/etc/include : header para programação em C

/etc/inittab : arquivo de configuração do init.

/home : pasta pessoal dos usuários comuns. Um "Meus Documentos da vida"

/lib : bibliotecas compartilhadas

/lib/modules : módulos externos do kernel usados para inicializar o sistema...

/misc : arquivos variados

/mnt : ponto de montagem de sistemas de arquivos (CD, floppy, partições...)

/proc : sistema de arquivos virtual com dados sobre o sistema.

/root : diretório pessoal do root

/sbin : arquivos/comandos especiais (geralmente não são utilizados por usuários comuns)

/tmp: arquivos temporários

/usr : Unix System Resources. Contém arquivos de todos os programas para o uso dos usuários de sistemas UNIX.

/usr/bin : executáveis para todos os usuários

/usr/sbin : executáveis de administração do sistema

/usr/lib : bibliotecas dos executáveis encontrados no /usr/bin

/usr/local : arquivos de programas instalados localmente

/usr/man : manuais

/usr/info : informações

/usr/X11R6 : Arquivos do X Window System e seus aplicativos.

/var : Contém arquivos que são modificados enquanto o sistema está rodando não é compartilhado em rede por ser específico de cada sistema, estando em constantes

modificações.

/var/lib : bibliotecas

/var/local : arquivos variáveis de processos que estão rodando.

/var/log : arquivos de log do sistema

/var/run : arquivos úteis até o próximo boot (atualizações de softwares e kernel)

/var/tmp : arquivos temporários dos programas

O PATH

Para visualizar o path (caminho de procura de arquivos e comandos executáveis), usa-se o comando `% echo $PATH`. Todos os comandos executáveis serão executados "se e

somente se" estiverem armazenados na variável path. O comando que está no PATH é executado a partir de qualquer diretório (não importa se o usuário está no home ou no

mnt ou qualquer outro). Para inserir um arquivo no path usa-se `"/` (Ex: `./arquivo`).

Para inserir o caminho para todos os usuários do sistema edite o arquivo `/etc/profile` ou `/etc/bashrc`.

Para alterar caminhos de um único usuário, edita-se o arquivo `.bash_profile` do diretório `/home/` ou `$HOME/.bashrc`

Para adicionar caminhos novos ao "caminho existente" use e pressione `,` vá até o fim da linha e adicione o novo caminho. Confira utilizando o comando `% env`.

Para adicionar novos comando com o alias faça:

`% alias cdon = "mount /dev/cdrom /mnt/cdrom"` o comando `%cdon` (recém criado) será adicionado ao path e utilizará o script de montagem do CD-ROM.

ATALHOS DO BASH

`Ctrl + Alt + BackSpace` : reinicia o X

`Ctrl + Alt + Del` : reinicia o sistema em modo texto

`Ctrl + Alt + F"x"` : muda o shell (F1 – F6 (texto), F7 (gráfico))

`Alt + F"x"` : alterna o shell (modo texto)

Alt + B : move o cursor do console para a palavra anterior

Alt + F : move o cursor do console para a palavra seguinte

Ctrl + A : vai para o início da linha

Ctrl + E : vai para o fim da linha

Ctrl + C : pára o comando

Ctrl + D : logout

Ctrl + W : apaga do cursor até o início da palavra

Ctrl + U : apaga do cursor até o início da linha

Ctrl + K : apaga do cursor até o fim da linha

Ctrl + L : clear

Ctrl + R : busca incremental no histórico de comandos

Ctrl + T : inverte caractere sob o cursor com o anterior

Ctrl + Y : re-inserir o último texto apagado

Ctrl + Z: colocar processo em background

Shift + Page(UP/DOWN) : navega pelo shell

!"x": repete o último comando (onde "x" é o comando a ser chamado – Exemplo: !mount)

TAB : autocompletar

COMANDOS DE INICIALIZAÇÃO

shutdown :Sintaxe: shutdown [atributo] [tempo]

Atributos: -r : reboot

-h : halt

-n : force (sem sincronizar)

-f : reboot rápido

-c : cancela shutdown corrente

-k : somente simula um shutdown

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

Tempo: now : executar no exato momento

Exemplo: Shutdown -r now

poweroff : halt

init 0 : halt

init 3: derruba o modo gráfico

init 5: iniciar o modo gráfico

init 6 : reboot

init q: examinar o arquivo /etc/inittab

init s: entrar em modo mono usuário (single mode)

exit: termina sessão atual

logout : termina sessão atual

COMANDOS DE SISTEMA

dmsg | less : mostrar lista de inicialização

arch : mostra a arquitetura do computador

uname : informações de sistema

Parâmetros: -m : hardware

-n : nome da máquina na rede

-r : versão do kernel

-s : nome do OS

-v : data de compilação do OS

-a : todas as acima...

uptime : mostra o tempo em que a máquina está no ar

free : mostra o uso da memória

vmstat : Mostra as estatísticas da memória virtual

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

ps -aux | grep tty : lista terminais em uso

tty : mostra em qual terminal o usuário atual está

stty : exibir ou escolher parâmetros do terminal

tset : escolher tipo de terminal

fgconsole : exibe o nº de terminais válidos

cu : chama outro sistema unix

type : informa o que faz um certo comando. Exemplo: type mount

time : mede o tempo gasto para executar um comando

cal : calendário

date: data e hora

who : mostra quem está na máquina no momento

whoami : mostra quem é você

last : informa a entrada e saída de usuários no sistema

lastlog : informa o último usuário que logou no sistema

at : agendar tarefas. Sintaxe : at [hora][data][comando]

crontab : programa tarefas.

Sintaxe: crontab [-parâmetros][usuário][arquivo]

Parâmetros: -l : exibe crontab do usuário atual

-r : remove crontab do usuário atual

São 6 campos separados por sendo que:

1º é o minuto

2º é o hora

3º é o dia do mês

4º é o mês

5º é o dia da semana (que começa com 0 = domingo)

6º é a tarefa ou script a ser executado

O crontab é um programa do Unix que edita o arquivo onde são especificados os comandos a serem executados e a hora e dia de execução pelo cron, um programa que executa

comandos agendados nos sistemas operacionais do tipo Unix (como o Linux ou o Minix, por exemplo). O cron se encarregará de verificar a hora e determinar se existe ou

não algum programa a ser rodado. Caso exista ele o rodará na hora e data solicitada.

Parâmetros

- l usuário -- lista as tarefas agendadas para o usuário
- e usuário -- edita o agendador
- d usuário -- apaga o arquivo do usuário
- c diretório -- especifica um diretório para o Crontab

Para executar basta chamá-lo da seguinte forma:

crontab [parâmetro]

Ex.: crontab -l (para visualizar a lista de tarefas já agendadas).

Sintaxe

mm hh dd MM ss script

Onde:

mm = minuto(0-59)

hh = hora(0-23)

dd = dia(1-31)

MM = mes(1-12)

ss = dia_da_semana(0-6)

script = comando a ser executado.

Obs 1: Em dia_da_Semana, 0 refere-se a domingo; e 6, ao sábado. No caso de dia da semana funciona também as três primeiras letras (em inglês) do dia da semana

(SUN,MON,TUE,WED,THU,FRI,SAT)

Obs 2: Em qualquer posição pode-se usar o * (asterisco) quando não se importar com o campo em questão.

Obs 3: Pode-se utilizar intervalos nesses campos. O caracter para intervalo é o - (hifen).

Obs 4: Pode-se utilizar lista de valores nesses campos. O caracter para a lista é a , (vírgula).

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

Obs 5: Qualquer texto colocado após o programa que será executado será considerado comentário e não será interpretado pelo cron

Exemplo:

```
00 * * * * /bin/script
```

Todo dia de hora em hora (hora cheia)

```
00-59/5 * * * * /bin/script
```

De cinco em cinco minutos todos os dias (note a divisão por 5 do intervalo 00-59)

lspci : lista os dispositivos PCI

setserial : informações sobre as portas seriais

COMANDOS PARA REDES

ifconfig : configura a interface de rede

netconf : Interface gráfica do RedHat para configurar a rede

netconfig : Outra interface gráfica para a configuração da rede

hostname : nome da máquina

netstat : Mostra informações sobre as conexões à rede

domainname: Mostra ou ajusta o nome de domínio do sistema

route : Mostra e manipula a tabela de roteamento

showmount : Mostra informações de sistemas NFS montados sobre a rede

nmap: capta informações em conexões remotas, como o sistema operacional usado, portas abertas...

Sintaxe: nmap -O 200.0.0.5

tracerout : mostra o caminho percorrido do pc até um "tal" servidor da internet

ping : informações de conexão

Parâmetros: -r : avisa se o pacote não chegar ao destino

-c"N" : onde "N" é o nº de pings a serem executados

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

-s : determina o tamanho dos pacotes enviados, por default é 56bytes

rsusers : mostra quem está usando as máquinas na rede

ruptime : informações sobre máquinas da rede local

Parâmetros : -a: usuários inativos a mais de uma hora

-l : ordena saída de informações de cada host

-r : inverte ordem de saída de informações

-t : (uptime)

-u: ordena saída por nº de usuários

rup: semelhante ao uptime, porém para outra máquina na rede. Sintaxe: rup

rwho : mostra o que está "sendo feito" nos computadores da rede local (semelhante ao ruptime)

mail : recebe ou envia e-mail

smbclient : Cliente para redes SAMBA.

smbmount : Monta um sistema de arquivos SAMBA (Windows)

COMANDOS DE AJUDA

man: manual de ajuda. Exemplo : man mount

xman : man em modo gráfico

info : como o man, porém com links nos textos

makewhatis : levanta banco de dados para o comando apropos

apropos:levanta respostas de comandos desconhecidos, antes deve-se executar makewhatis.
Exemplo :

% apropos compiler (serão localizados arquivos que contenham a descrição de compiler).

help: ajuda do shell atual

COMANDOS PARA MANIPULAÇÃO DE DIRETÓRIOS

cd : entra no diretório

Atributos: - : volta ao último diretório buscado

~ [nomedapastadousuario] : vai para a pasta do usuário

/ : diretório raiz

.. : diretório "acima"

ls : lista arquivos e diretórios

Atributos : -C : exibe em colunas

-R : modo recursivo

-a : mostra ocultos

-l : mostra propriedades

mkdir : cria um diretório

rmdir : remove um diretório (atributo : -R para deletar diretórios não vazios)

pwd : mostra o diretório atual

cp : copia diretórios

Atributos: -f : não pergunta para sobrescrever

-i : pergunta para sobrescrever

-r : cópia recursiva

mv : movimenta diretórios (mesmos atributos do cp)

du : mostra hierarquia

Atributos: -a : informações de todos os arquivos

-b : mostra em bytes

-k : mostra em kbytes

-s : lista o total de cada argumento

-x : ignora diretórios com sistemas de arquivos diferentes

-S : apresenta o tamanho de cada diretório separadamente

diff : exibe diferenças entre 2 arquivos ou diretórios

diff3 : exibe diferenças entre 3 arquivos ou diretórios

ln : link simbólico para arquivos e/ou pastas. Exemplo: ln linux-2.6 kernel-atual

COMANDOS PARA MANIPULAÇÃO DE ARQUIVOS

bash : executa o bash

basename : exclui o sufixo do arquivo

fsbrowser : encontra arquivos ou diretórios no pc

head : lista a primeira parte dos arquivos

tail : o contrário do head

cat : exibe conteúdo do arquivo ou direciona-o para outro

Exemplos: cat /home/fulano/contato (exibe o conteúdo do arquivo contato)

cat contato1 > contato2 (faz uma cópia do arquivo contato)

cat contato1 >> contato2 (adiciona um arquivo ao outro)

cat musica.wav > /dev/dsp1 (direciona o som para o dispositivo de audio)

cat arquivo1 arquivo 2 | gzip > arquivo1e2.gz (compacta arquivo1 e arquivo2 num só arquivo)

tac : inverso do cat

file : identifica o tipo de arquivo

find : localiza arquivo ou diretório.

Sintaxe: find [opções] [dados] [opções]

Opções: -name: parâmetro seguido do arquivo a ser procurado

-print : mostrar resultado na tela

-exec "comando" : executa um comando

-ok : idem a exec, porém pergunta antes de executar o comando

awk : procura por um modelo a partir de um arquivo

bdiff : compara 2 arquivos grandes

bfs: procura um arquivo grande

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

diff : exibe diferenças entre 2 arquivos ou diretórios

uniq : compara 2 arquivos, exibe as linhas incomparáveis do arquivo

diff3: exibe diferenças entre 3 arquivos ou diretórios

cmp : compara 2 arquivos, mostra a localização da primeira diferença entre eles

comm: compara 2 arquivos para determina quais linhas são comuns entre eles

split : dividir um arquivo

more e less : filtros depaginação

Exemplos : more /etc/fstab

ps -aux | less

ls : lista arquivos do diretório

Atributos : -C : exibe em colunas

-R : modo recursivo -a : mostra ocultos

-l : mostra propriedades

cp : copia arquivos

Atributos: -f : não pergunta para sobrescrever

-i : pergunta para sobrescrever

-r : cópia recursiva

mv : movimenta arquivos (mesmos atributos do cp)

rm : remove um arquivo

wc: (não cacete! não é banheiro não!) esse aí lista as propriedades dos arquivos

Atributos : -l : nº de linhas do arquivo

-c : n de caracteres

-w : nº de palavras

-b : nº de bytes

grep : uma espécie de busca. Exemplo: % grep tty

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

od : mostra o arquivo em formato octal

ln : link simbólico para arquivos e/ou pastas. Exemplo: ln linux-2.6 kernel-atual

COMANDOS PARA COMPACTAR

Sem descrições... todos fazem a mesma coisa, só que em formatos diferentes, dependendo da ocasião um pode ser melhor que o outro... depende do "gosto" de cada um...

compress/ uncompress : compacta / descompacta em .z

Sintaxe : idem a bzip2

Atributos: -f : compacta em background

-c : descompacta na tela

-V : informações sobre o arquivo

-v : informações sobre a compactação corrente

-d: junto com o compress, descompacta o arquivo

tar : Atributos: -t : lista conteúdo do arquivo tar

-x : extrai arquivos

-c : cria um arquivo tar

-d : compara um arquivo tar com arquivos atuais

-r : anexa arquivos no fim do arquivo tar

-u : analisa arquivos

-A : anexa outros tar

-v : mostra informações de processamento

Exemplos: tar cvM -L 1350 -f /mnt/floppy/partes.tgz grande.tgz : quebra um arquivo grande em

partes menores.

tar pvfx arquivo.tar: extrai tar

tar zpvfx : extrai tar.gz ou tgz

tar jpvfx : extrai tar.bz2

gzip : compacta arquivos e nomes de arquivos com mais de 14 letras

Atributos: -d : descompacta

-f : compacta

-l : informações sobre arquivo compactado

-r : recursivo

gunzip : descompacta

Exemplo: gzip -c arquivo1 > arquivo1e2.gz ; gzip -c arquivo2 > arquivo1e2.gz ou

cat arquivo1 arquivo 2 | gzip > arquivo1e2.gz

(compacta arquivo1 e arquivo2 num só arquivo)

bzip / bunzip: compacta / descompacta

bzcat : descompacta na tela do sistema

bzip2recover : recupera arquivos danificados quando for possível

Atributos : -1 : (1, 2,3...9) sendo que 9 é o maior fator de compactação

-v : informações de processamento

-c : descompacta na tela do sistema

-d : descompacta

-t : verifica integridade do arquivo

-f : sobrescreve

-s : reduz us da memória exigida

cpio: executa funções equivalentes a arquivos de dados em formato cpio ou tar, e contém informações sobre o arquivo. Seus modos de arquivamento são entrada, saída e

passagem.

Modo de entrada: extrai arquivos de um arquivo maior, caso não for inserido um padrão, todos são extraídos.

Modo de saída: copia os arquivos em um arquivo maior

Modo de passagem : combina os modos de entrada e saída sem usar modo de arquivamento, copia

uma árvore de diretórios para outra

Parâmetros: -0 : (saída / passagem) lê arquivo terminado em caractere nulo; ao invés de nova linha

-A : (saída) adiciona um arquivo existente; deve estar especificado -0 ou -F

-d : cria diretório (se for preciso)

-H : indica o formato do arquivo maior

-i : (entrada) extrai

-o : (saída) cria

-t : (entrada) lista tabela

-u : sobrescreve sem perguntar

-v : lista arquivos processados

USUÁRIOS E GRUPOS

useradd : adiciona usuário

userdel : exclui usuário

username : exibe nome do usuário logado no momento

users : exibe os usuários logados no momento

usermod : Modifica um usuário.

id : identificação do usuário e grupos a que pertence

passwd : adiciona senha em usuário já cadastrado. Exemplo: % passwd fulano

chpasswd : muda senha de usuário

groupadd : cria grupo

groupdel : exclui grupo

groups : lista grupos

groupmod : Modifica um grupo

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

newgrp : altera identificação do grupo

chgroup : altera propriedade do grupo

chown : altera usuário/propriedade do grupo de cada arquivo especificado. Ex: % chown -R root /home

sg : executa comando como se fosse de outro grupo

su : usado para ganhar permissão de root.

wall : Manda uma mensagem para todos no terminal.

PERMISSÕES

Primeiramente preciso dar umas breves explicações sobre permissões, dono, grupo e coisara de esquemas... Vou mostrar permissões no modo "literal" e no modo

"octal", que são mais fáceis e mais usadas, porém pode ser feito com aplicativos, cliques no botão direito e coisara, se a distro tiver tal recurso. Para todos os

modos usa-se o comando:

% chmod [opções] [permissão] [diretório]

Atributos: -v : mostra arquivos processados

-f : não mostra mensagens de erro

-c : mostra somente arquivos com permissão

-R : recursivo

Dono: é o usuário que criou o arquivo

Grupo: é um recurso para que vários usuários tenham como "usar" um mesmo arquivo

Outros : não são donos e nem pertencem ao grupo do arquivo

ESQUEMA USADO NO % ls -l

r : permite leitura

w : permite edição/gravação

x : permite executar arquivos executáveis. Se for um diretório, pode ser acessado pelo comando % cd

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

Se houver 1 "d", indica que é um diretório; se houver um "-" indica um arquivo

Exemplo: -rw----- (somente o dono pode ler e gravar no arquivo)

MODO LITERAL :

Esse é o modo em que o sistema nos mostra as permissões num % ls -l, mas esse é o mais "chatinho" pois é feito linha por linha.

[a]LL : todos

[g]roup : grupo

[o]thers : outros

[u]ser : dono

[+] : adiciona permissão

[-] : remove permissão

Com o comando % chmod ficaria assim:

% chmod a -rx /mnt (retira a permissão de leitura e execução para todos no diretório /mnt)

% chmod o +r /mnt (da permissão para "outros", no caso os que não pertencem ao grupo do dono)

MODO OCTAL :

DÍGITO PERMISSÃO (UGOA)

0 Nenhuma ---

1 Executar --x

2 Escrever -w-

3 Escrever/Executar -wx

4 Ler r--

5 Ler/Executar r-x

6 Ler/Escrever rw-

7 Ler/Escrever/Executar rwx

Exemplo: % chmod -R 750 /mnt (Dono: pode Ler, Escrever e Executar; Grupo: pode Ler e Executar, e os outros não podem nada) para o diretório /mnt e todos os seus

subdiretórios.

COMANDOS PARA MANIPULAR IMPRESSÃO

lpq : mostra trabalhos de impressão atual

lpr : copia um arquivo para a linha da impressora

lprm : remove trabalhos de impressora

lpstat: informações

Atributos: -a : lista relatórios enviados para a impressora

-p : lista o estado das impressoras

-s : sumário do estado das impressoras

-R : mostra o nº de cada jobna fila de impressão

COMANDOS PARA MANIPULAR PROCESSOS

ps -aux | less: mostra o PID dos processos

top : mostra todos os processos ativos no pc. (em modo grafico usa-se ktop (do kde) gtop (do gnome)...)

kill: mata um processo pelo PID. Sintaxe: kill [PID]

killall : mata processospelo nome. Sintaxe: killall -HUP [nome]

sleep : torna um processo inativo por determinado tempo

nice : prioridade de execussão (-19 é o máximo e 19 é o mínimo). Sinaxe: nice [opções][comando]

& : quando colocado no fim da linha, executa o processo em 2º plano (ou background se preferir)

jobs : exhibe processos em 2º plano

bg : ao rodar um processo tecle Ctrl + Z para congelar o processo, depois digite bg para o programa voltara "rodar" em 2º plano.

fg: para fazer o processo voltar ao 1º plano. Sintaxe: fg [nº do processo]

nohup : logout sem matar processos no 2º plano. Exemplo: nohup cc program C & ; logout

COMANDOS DE MANIPULAÇÃO DE MÍDIAS

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

cdisk : particionador

mke2fs : formata o hd. Exemplo: mke2fs -j /dev/hda1 (formata a primeira partição do hd primário em ext3)

dosfsck : Verifica e repara sistemas FAT do DOS

dumpe2fs : Faz o backup de blocos do HD e grupos.

e2fsck : Verifica um volume ext2 em busca de erros.

E2label : Muda o nome de uma partição ext2.

dump : Usado para fazer o backup de um sistema ext2. O complemento deste comando é o restore

restore : Usado para restaurar um sistema ext2.

badblock: busca por setores ruins

fsck: faz reparos na partição

df : mostra espaço usado, livre e a capacidade das partições do HD

mount : monta dispositivo. Exemplo: mount -vfat /dev/hda1

umount : desmonta dispositivo. Exemplo: umount /mnt/cdrom

supermount : monta dispositivos automaticamente. Sintaxe: supermount [opções] [/local/do/dispositivo]

mformat : formatação de baixo níveis em disquetes no formato M\$-DOS

fdformat : formatação de baixo níveis em disquetes. Exemplo: fdformat /dev/fd0 H1440

GRAVANDO CDs

Aqui eu vou falar de gravação pelo console mesmo... linhas de comando... as vezes até acaba sendo mais fácil... Ah! Essas aí foram inteiramente retiradas de um e-mail

que recebi da Dicas-L :-)

Criando ..iso:

mkhybrid -r -j -o imagem.iso diretório_dos_arquivos

Testando o .iso antes de gravar

mount -t iso9660 -o loop imagem.iso /mnt/iso

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

Fazendo uma cópia fiel de cd pra cd:

```
cdrecord -v speed=8 dev=1,1,0 -isozsize /dev/cdrom
```

Fazendo CD de Boot:

Crie a imagem apartir de um disquete de boot

```
dd if=/dev/fd0 of=boot.img bs=18k
```

Apagando CD-RW

Para sobrescrever um CD-RW use o parâmetro blank=fast no cdrecord

CD-ROM de audio e dados:

```
cdrecord -v speed=8 dev=1,1,0 -data imagem.iso -audio track1.cdr  
track2.cdr
```

USANDO O RPM

RPM é um pacote (normalmente pré-compilado para cada distro) de fácil instalação. Com uma só linha de comando ou cliques no rato são suficientes para instalar e não se

incomodar com compilações e coisarada...

Sintaxe : rpm [parâmetros][opções][pacote.rpm]

Parâmetros : -i : instala

-e : desinstala

-q : consulta

-v : verbose

-h: exibe "%" como porcentagem a medida que o pacote se instala

showrc : exibe

checksig : verifica assinatura

setugids [especificações do pacote] : ajusta UID

etperms [especificações do pacote] : ajusta permissões

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

+ : após de um parâmetro, indica que pode haver repetição

Obtendo instalações de pacotes instalados:

% rpm -qa | sort | less : lista em ordem alfabética os pacotes instalados

% rpm -qa | wc -1 : mostra a quantidade de pacotes instalados

% rpm -qf /usr/bin/nomedopacote.rpm : mostra os pacotes pertencentes ao arquivo

% rpm -qa | grep gimp: lista todos os arquivos que correspondem ao gimp

Para pacotes não instalados:

% rpm -qip /.../.../pacote.rpm : informações do pacote

% rpm -qdp /.../.../pacote.rpm : encontra documentação

% rpm -qp /.../.../pacote.rpm : mostra como será o nome do pacote

% rpm -qlp /.../.../pacote.rpm : lista os arquivos do pacote

% rpm -qcp /.../.../pacote.rpm : lista arquivos de configuração dos pacotes

% rpm -q --whatrequires /.../.../pacote.rpm : mostra o requerimentos

Desinstalação depacotes:

% rpm -e [opções] pacote.rpm

Opções: --allmatches : remove todas as versões do pacote

--noscripts : não executa scripts de pré/pós instalação

--nodeps : verifica se as dependências serão quebradas

--test : somente simula a desinstalação

Instalação/Atualização de pacotes:

% rpm -ivh [opções] pacote.rpm : instala

% rpm -Uvh [opções] pacote.rpm : atualiza

Opções: --force : sobrescreve

--noscripts : não executa scripts de pré/pós instalação

--nodeps : verifica se as dependências serão quebradas

--excludedocs : não instala arquivos de texto

--test : somente simula a desinstalação

--percent : exibe porcentagem

--oldpkage : atualiza para versão anterior

--replacepkgs : substitui arquivos instalados

--allfiles : instala atualiza pacotes faltantes

% whereis : mostra a localização dos binários do pacote instalado

% rpm -ivh /.../.../gimp-* : instala todos os pacotes do gimp simultaneamente

Instalação por FTP:

% ftp://:@host /.../.../arquivo.rpm

Opções: --ftpproxy : a "máquina" do host citado será o servidor proxy para transferências

--ftpport: Especifica uma porta TCP ao invés de FTP (porta 21). É cara, troca o xxx pelo número da porta.

VERIFICAÇÕES NO SERVIDOR

--Determinando a memória física
grep MemTotal /proc/meminfo

--Verificar o SWAP configurado
grep SwapTotal /proc/meminfo

--Verificar a memória RAM livre
free

-- Verificar a quantidade de processadores e tipo
grep "model name" /proc/cpuinfo

#####74- Copiar de uma máquina para a outra com o Linux #####

scp teste.html oracle@svuxsup2:/home/oracle/.

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

#####75- Estimando espaço para criação de um índice #####

Sintaxe:

```
DBMS_SPACE.CREATE_INDEX_COST (  
ddl IN VARCHAR2,  
used_bytes OUT NUMBER,  
alloc_bytes OUT NUMBER,  
plan_table IN VARCHAR2 DEFAULT NULL);
```

Parâmetros:

ddl The create index DDL statement
used_bytes The number of bytes representing the actual index data
alloc_bytes Size of the index when created in the tablespace
plan_table Which plan table to use, default NULL

Exemplo:

```
declare  
u_bytes number;  
a_bytes number;  
begin  
  
dbms_space.create_index_cost (  
ddl => 'CREATE UNIQUE INDEX TESTE.CRR_DATA_PK ON TESTE.CDR_DATA' ||  
'(MSG_ID, MSG_ID2, MSG_ID_SERV, SPLIT_ROW_NUM, CDR_DATA_PARTITION_KEY)  
tablespace TESTE_INDEX',  
used_bytes => u_bytes,  
alloc_bytes => a_bytes  
);  
dbms_output.put_line ('Used Bytes = ' || u_bytes);  
dbms_output.put_line ('Allocated Bytes = ' || a_bytes);  
end;  
/
```

Resultado:

Used Bytes = 2514576775 bytes ou 2,34Gb à ocupará este valor em bytes para dados
Allocated Bytes = 4160749568 ou 3,87Gb à ocupará este valor dentro do tablespace de índice.

#####76- Trabalhando com tabelas e índices fragmentados (descobrimos e desfragmentamos) #####

--Para verificar quais tabelas atacar primeiro pode-se verificar quais tabelas sofrem mais DML,
--também pode-se utilizar o script do número 53 para ajudar

set serveroutput on

```
declare
    v_unformatted_blocks number;
    v_unformatted_bytes number;
    v_fs1_blocks         number;
    v_fs1_bytes          number;
    v_fs2_blocks         number;
    v_fs2_bytes          number;
    v_fs3_blocks         number;
    v_fs3_bytes          number;
    v_fs4_blocks         number;
    v_fs4_bytes          number;
    v_full_blocks        number;
    v_full_bytes         number;
begin
    dbms_space.space_usage (
        '&OWNER', -----owner
        '&TABLE ou INDEX', -----table_name
        'TIPO:TABLE ou INDEX',
        v_unformatted_blocks,
        v_unformatted_bytes,
        v_fs1_blocks,
        v_fs1_bytes,
        v_fs2_blocks,
        v_fs2_bytes,
        v_fs3_blocks,
        v_fs3_bytes,
        v_fs4_blocks,
        v_fs4_bytes,
        v_full_blocks,
        v_full_bytes);
    dbms_output.put_line('Unformatted Blocks = ' || v_unformatted_blocks);
    dbms_output.put_line('FS1 Blocks      = ' || v_fs1_blocks);
    dbms_output.put_line('FS2 Blocks      = ' || v_fs2_blocks);
    dbms_output.put_line('FS3 Blocks      = ' || v_fs3_blocks);
    dbms_output.put_line('FS4 Blocks      = ' || v_fs4_blocks);
    dbms_output.put_line('Full Blocks     = ' || v_full_blocks);
end;
/
```

segment_owner - Schema name of the segment to be analyzed

segment_name - Name of the segment to be analyzed

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

unformatted_blocks - Total number of blocks that are unformatted
fs1_blocks - Number of blocks that has at least 0 to 25% free space
fs2_blocks - Number of blocks that has at least 25 to 50% free space
fs3_blocks - Number of blocks that has at least 50 to 75% free space
fs4_blocks - Number of blocks that has at least 75 to 100% free space
full_blocks - Total number of blocks that are full in the segment

-- Exemplo para usar com tabelas particionadas:

```
DECLARE
  v_unformatted_blocks number;
  v_unformatted_bytes number;
  v_fs1_blocks number;
  v_fs1_bytes number;
  v_fs2_blocks number;
  v_fs2_bytes number;
  v_fs3_blocks number;
  v_fs3_bytes number;
  v_fs4_blocks number;
  v_fs4_bytes number;
  v_full_blocks number;
  v_full_bytes number;
BEGIN
  dbms_space.space_usage ('LF', 'LF_NF_SAIDA_IMPOSTO', 'TABLE PARTITION',
v_unformatted_blocks, v_unformatted_bytes, v_fs1_blocks, v_fs1_bytes, v_fs2_blocks,
v_fs2_bytes, v_fs3_blocks, v_fs3_bytes, v_fs4_blocks, v_fs4_bytes, v_full_blocks,
v_full_bytes, 'PN_ANO_2008');
  dbms_output.put_line('Unformatted Blocks = ' || v_unformatted_blocks);
  dbms_output.put_line('FS1 Blocks = ' || v_fs1_blocks);
  dbms_output.put_line('FS2 Blocks = ' || v_fs2_blocks);
  dbms_output.put_line('FS3 Blocks = ' || v_fs3_blocks);
  dbms_output.put_line('FS4 Blocks = ' || v_fs4_blocks);
  dbms_output.put_line('Full Blocks = ' || v_full_blocks);
end;
```

```
ALTER TABLE OWNER.TABLE ENABLE ROW MOVEMENT;
--ALTER TABLE OWNER.TABLE DISABLE ROW MOVEMENT;
```

```
ALTER TABLE OWNER.TABLE SHRINK SPACE COMPACT;
ALTER TABLE OWNER.TABLE SHRINK SPACE;
ALTER TABLE OWNER.TABLE SHRINK SPACE CASCADE;
```

-- Para verificar após shirink completo:
set serveroutput on

```
declare
  TOTAL_BLOCKS number;
  TOTAL_BYTES number;
  UNUSED_BLOCKS number;
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle


```

UNUSED_BYTES number;
LAST_USED_EXTENT_FILE_ID number;
LAST_USED_EXTENT_BLOCK_ID number;
LAST_USED_BLOCK number;
begin
  dbms_space.unused_space(
    '&OWNER',
    '&OBJETO',
    '& Tabela ou índice [INDEX|TABLE]',
    TOTAL_BLOCKS,
    TOTAL_BYTES,
    UNUSED_BLOCKS,
    UNUSED_BYTES,
    LAST_USED_EXTENT_FILE_ID,
    LAST_USED_EXTENT_BLOCK_ID,
    LAST_USED_BLOCK);

  dbms_output.put_line('OBJECT_NAME = FREELIST_T');
  dbms_output.put_line('-----');
  dbms_output.put_line('TOTAL_BLOCKS = ' || TOTAL_BLOCKS);
  dbms_output.put_line('UNUSED_BLOCKS = ' || UNUSED_BLOCKS);
  dbms_output.put_line('TOTAL_BYTES = ' || TOTAL_BYTES);
  dbms_output.put_line('UNUSED_BYTES = ' || UNUSED_BYTES);

end;
/

```

#####76.2 Desfragmentando com MOVE

1- Alter table TABLE_NAME move;
 1.2- Se quiser mudar de tablespace faça:
 Alter table TABLE_NAME move tablespace NOME_TABLESPACE;

--Para descobrir quais tabelas fazer o move:
 SELECT DISTINCT 'alter table &OWNER' || t.table_name || ' move;'
 FROM dba_tab_columns c, dba_tables t
 WHERE t.owner = '&OWNER' AND c.data_type NOT IN('LONG', 'LONG RAW', 'CLOB', 'RAW',
 'NCLOB', 'BLOB')
 AND c.table_name = t.table_name;

2- Rebuild dos índices:

alter index INDEX_NAME rebuild online nologin;

#####77- Gerando exp e imp sem gerar arquivo físico

```

#-----
# Desafio
#-----

```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

Devido a falta de disco local no servidor é necessário fazer o export e import sem escrever o arquivo no disco.

```
#-----  
# Servidores  
#-----
```

Origem: svuxweb4 (user usuario_dba)

Destino: svuxpcrs35 (user usuario_dba2)

```
#-----  
# Procedimento  
#-----
```

O teste será realizado com um Owner.

1) Na máquina de destino editar o comando imp escutando na porta 5000 conforme abaixo:

Obs: file=<(nc -l 5000) sinal "<"

```
[oracle@svuxpcrs35 scripts]$ imp usuario_dba2/work file=<(nc -l 5000) fromuser=usuario_dba touser= usuario_dba2
```

2) Na máquina de origem editar o comando exp direcionando a saída na porta 5000 conforme abaixo:

Obs: file=>(nc scuxpcrs35 5000) sinal ">" mais o nome da máquina de deverá conectar

```
[oracle@svuxweb4 lixo]$ exp agerondo file=>(nc svuxpcrs35 5000) owner= usuario_dba compress=n consistent=y
```

Obs:

- Exp/Imp não vai gerar arquivo fisico
- a entrada de dados no comando exp será enviado via rede diretamente para o comando imp
- resultado pode ser visto abaixo

```
#-----  
# Saída dos comandos.  
#-----
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

1) Import

```
[oracle@svuxpcrs35 scripts]$ imp agerondopweb/work file=<(nc -l 5000) fromuser=
usuario_dba touser= usuario_dba2
```

2) Export

```
[oracle@svuxweb4 lixo]$ exp usuario_dba file=>(nc svuxpcrs35 5000) owner= usuario_dba
compress=n consistent=y
```

#####78- DBMS coleta de estatísticas #####

```
--Coleta de estatísticas
--SCHEMA
EXEC dbms_stats.gather_schema_stats(
ownname      => 'SCOTT',
estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE,
method_opt    => 'for all columns size skewonly',
degree        => DBMS_STATS.DEFAULT_DEGREE);
```

Exemplo:

```
EXEC dbms_stats.gather_schema_stats('RANDRADE',estimate_percent =>
DBMS_STATS.AUTO_SAMPLE_SIZE,method_opt =>'for all columns size skewonly',degree =>
DBMS_STATS.DEFAULT_DEGREE);
```

```
--coleta de estatísticas:
--TABELA:
```

```
DBMS_STATS.GATHER_TABLE_STATS (
  ownname      VARCHAR2,
  tabname      VARCHAR2,
  partname     VARCHAR2 DEFAULT NULL,
  estimate_percent NUMBER DEFAULT NULL,
  block_sample  BOOLEAN DEFAULT FALSE,
  method_opt    VARCHAR2 DEFAULT 'FOR ALL COLUMNS SIZE 1',
  degree        NUMBER DEFAULT NULL,
  granularity   VARCHAR2 DEFAULT 'DEFAULT',
  cascade       BOOLEAN DEFAULT FALSE,
  stattab      VARCHAR2 DEFAULT NULL,
  statid        VARCHAR2 DEFAULT NULL,
  statown      VARCHAR2 DEFAULT NULL);
```

-- Boas práticas:

```
exec dbms_stats.GATHER_TABLE_STATS (
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
OWNNAME => 'OWNER',
TABNAME => 'NOME DA TABELA',
ESTIMATE_PERCENT => (null se a tabela em questão for menor que 1Gb, '20' se for maior),
METHOD_OPT => 'FOR ALL INDEXED COLUMNS SIZE 1' -- (coleta sem histogramas)
DEGREE => (2 para tabelas menor que 1Gb, 4 para tabelas entre 1 e 20Gb, 10 para todas as
outras)
GRANULARITY => 'ALL',
CASCADE => TRUE,
NO_INVALIDATE => TRUE);
```

Exemplo:

```
execute dbms_stats.gather_table_stats (ownname => 'ARBOR', tabname => 'DESCRIPTIONS',
estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE, method_opt => 'for all columns
size skewonly', cascade => true);
```

--BANCO TODO:

```
execute dbms_stats.gather_database_stats ( estimate_percent =>
DBMS_STATS.AUTO_SAMPLE_SIZE, method_opt => 'for all columns size skewonly', cascade =>
true);
```

-- Bloquear a estatística:

```
EXEC dbms_stats.lock_table_stats(ownname=> '&SCHEMA', tabname=> '&TABLE');
```

-- Desbloquear:

```
EXEC dbms_stats.unlock_table_stats(ownname=> '&SCHEMA', tabname=> '&TABLE');
```

#####79- Descobrir quais tabelas usam determinada PK como FK #####

--usado para analisar possibilidades de truncates e deletes

```
select * from dba_constraints
where R_constraint_name = 'NOME_DA_PK'
```

manual de como proceder

```
select * from user_constraints where table_name='BSC_NF_ITEM' --achar a constraint da
tabela procurar a (P) primary key
```

```
select * from user_cons_columns where constraint_name in('SYS_C0049147') -- pega coluna da constraint PK abaixo
```

```
select * from user_constraints where R_constraint_name in('SYS_C0049147') order by 4-- pega tabelas dependentes da constraint acima
```

#####80- Conectar via sqlplus sem TNSNAMES registrado #####

```
CONNECT [USER]/[PASS]@[HOSTNAME]:[PORTA_LISTNER]/[SERVICENAME]
```

#####81- Verifica sessões consumidoras de UNDO #####

```
SELECT b.segment_name,
       a.username,
       a.SID,
       a.serial#,
       c.used_ublk,
       c.used_urec,
       c.start_ubafil,
       c.start_ubablk,
       c.start_ubarec,
       b.status,
       b.tablespace_name,
       b.segment_id,
       b.file_id,
       b.block_id
FROM v$session a, dba_rollback_segs b, v$transaction c
WHERE b.segment_id = c.xidusn
AND a.taddr = c.addr
```

#####82- Extrair DDL de um objeto #####

```
select dbms_metadata.get_ddl('OBJECT_TYPE','OBJECT_NAME','OWNER') from dual;
```

Exemplo de uso:

```
spool c:\temp\lf.txt
```

```
SELECT DBMS_METADATA.GET_DDL('TABLE',u.table_name) || ';'
FROM USER_TABLES u;
```

```
spool off
```

--Variação

```
SELECT text
From DBA_SOURCE
```

WHERE upper(name) = 'P_LF_REL_SAIDA_MOD8'

#####83- Verificando traces rodando #####

```
SELECT *
FROM (SELECT SID,
      'SESSION_TRACE' trace_type
      FROM v$session
      WHERE sql_trace = 'ENABLED')
UNION
(SELECT SID,
      t.trace_type
      FROM v$session s, dba_enabled_traces t
      WHERE t.trace_type = 'CLIENT_ID'
      AND s.client_identifier = t.primary_id)
UNION
(SELECT SID,
      t.trace_type
      FROM v$session s, dba_enabled_traces t, v$instance i
      WHERE t.trace_type = 'SERVICE'
      AND s.service_name = t.primary_id
      AND ( t.instance_name IS NULL
      OR t.instance_name = i.instance_name))
UNION
(SELECT SID,
      t.trace_type
      FROM v$session s, dba_enabled_traces t, v$instance i
      WHERE t.trace_type = 'SERVICE_MODULE'
      AND s.service_name = t.primary_id
      AND s.module = t.qualifier_id1
      AND ( t.instance_name IS NULL
      OR t.instance_name = i.instance_name))
UNION
(SELECT SID,
      t.trace_type
      FROM v$session s, dba_enabled_traces t, v$instance i
      WHERE t.trace_type = 'SERVICE_MODULE_ACTION'
      AND s.service_name = t.primary_id
      AND s.module = t.qualifier_id1
      AND s.action = t.qualifier_id2
      AND ( t.instance_name IS NULL
      OR t.instance_name = i.instance_name))
UNION
(SELECT SID,
      t.trace_type
      FROM v$session s, dba_enabled_traces t, v$instance i
      WHERE t.trace_type = 'DATABASE'
      AND ( t.instance_name IS NULL
```

```
OR t.instance_name = i.instance_name))
```

#####84- Criar script de grant por atacado #####

```
SELECT 'grant ' || DECODE (object_type, 'TABLE', 'select,insert,update,delete', 'PACKAGE',
'execute', 'PROCEDURE', 'execute')
|| ' on ' || owner || '.' || object_name || ' to &USER;'
FROM dba_objects
WHERE owner = '&OWMER'
AND object_type IN ('TABLE', 'PACKAGE', 'PROCEDURE');
```

-- Variação:

```
SELECT 'grant ' || DECODE (object_type, 'TABLE', 'select,insert,update,delete', 'VIEW',
'select,insert,update,delete', 'execute')
|| ' on ' || owner || '.' || object_name || ' to &USER;'
FROM dba_objects
WHERE owner = '&OWMER'
AND object_type IN ('TABLE', 'PACKAGE', 'PROCEDURE', 'FUNCTION', 'VIEW');
```

#####85- Descobrir o SCN de uma banco de dados #####

```
SELECT CURRENT_SCN FROM V$DATABASE;
```

```
OU SELECT DBMS_FLASHBACK.GET_SYSTEM_CHANGE_NUMBER FROM DUAL;
```

#####86- Transportar tablespace de uma base para outra #####

Oracle 10G

Origem Plataforma w2k

Destino Plataforma RH 5.2

1 - Logar no SO/Banco da plataforma w2k.

2 - ALTER TABLESPACE VENDAS2009 READ ONLY;

3 - exp USERID=system/xxx AS SYSDBA TRANSPORT_TABLESPACE=y
TABLESPACES=vendas2009 FILE=vendas2009.dmp;

4 - Copiar os arquivos físicos para localidade correta no Red Hat 5.2

5 - logar no SO/Banco do Red Hat 5.2

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
6 - imp USERID=system/xxx AS SYSDBA TRANSPORT_TABLESPACE=y DATAFILES='/locate
    TABLESPACES=vendas2009 FILE=vendas2009.dmp;
```

```
7 - ALTER TABLESPACE VENDAS2009 READ WRITE;
```

#####87-Saber estatisticamente informações de um SQL que esteja rodando #####

```
SELECT sharable_mem,
       persistent_mem,
       runtime_mem,
       sorts,
       loaded_versions,
       open_versions,
       users_opening,
       executions,
       users_executing,
       loads,
       first_load_time,
       invalidations,
       parse_calls,
       disk_reads,
       buffer_gets,
       rows_processed,
       command_type,
       optimizer_mode,
       optimizer_cost,
       parsing_user_id,
       parsing_schema_id,
       kept_versions,
       address,
       hash_value,
       child_number,
       module,
       module_hash,
       action,
       action_hash,
       serializable_aborts,
       type_chk_heap,
       outline_category
FROM v$sql
WHERE (address, hash_value) = (SELECT sql_address,
                                sql_hash_value
                               FROM v$sqlsession
                               WHERE SID = :SID)
```


#####88- Duplicando banco com RMAN usando catálogo #####

RUN

```
{
# The DUPLICATE command uses an automatic sbt channel (se for via disco façaset channel).
# Because the target datafiles are spread across multiple directories,
# run SET NEWNAME rather than DB_FILE_NAME_CONVERT
SET NEWNAME FOR DATAFILE 1 TO '/oradata1/system01.dbf';
SET NEWNAME FOR DATAFILE 2 TO '/oradata2/undotbs01.dbf';
SET NEWNAME FOR DATAFILE 3 TO '/oradata3/cwmlite01.dbf';
SET NEWNAME FOR DATAFILE 4 TO '/oradata4/drsys01';
SET NEWNAME FOR DATAFILE 5 TO '/oradata5/example01.dbf';
SET NEWNAME FOR DATAFILE 6 TO '/oradata6/indx01.dbf';
# Do not set a newname for datafile 7, because it is in the tools tablespace,
# and you are excluding tools from the duplicate database.
SET NEWNAME FOR DATAFILE 8 TO '/oradata7/users01.dbf';
DUPLICATE TARGET DATABASE TO dupdb
SKIP TABLESPACE tools
LOGFILE
GROUP 1 ('/duplogs/redo01a.log',
        '/duplogs/redo01b.log') SIZE 200K REUSE,
GROUP 2 ('/duplogs/redo02a.log',
        '/duplogs/redo02b.log') SIZE 200K REUSE;
```

Referência:

<http://youngcow.net/doc/oracle10g/backup.102/b14191/rcmdupdb005.htm#BGBFADEH>

#####89 - Verificando tamanho máximo do datafile #####

O tamanho máximo de um datafile é em blocos 2^{22} (4194304).

Para saber em bytes multiplica-se pelo tamanho do block (parâmetro db_block_size)

Ex:

$4194304 * 8192 = 34359738368$ (em bytes) ou 32GB.

#####90- Descobrir o valor consumido em bytes por determinada quantidade de linhas

#####

Faz um count na tabela específica e multiplica a quantidade de linha pelo valor do row average size que se encontra na tabela dba_tables no campo:

avg_row_len

#####91- Verificando estatísticas de uma determinada sessão #####

```
SELECT pt.VALUE,  
  
sn.NAME  
  
FROM v$statname sn, SYS.v_$sesstat pt  
  
WHERE sn.statistic# = pt.statistic#  
  
AND pt.SID = &SID  
  
AND UPPER (sn.NAME) IN  
  
( 'RECURSIVE CALLS', 'DB BLOCK GETS', 'CONSISTENT GETS', 'PHYSICAL READS', 'REDO SIZE',  
'BYTES SENT VIA SQL*NET TO CLIENT', 'BYTES RECEIVED VIA SQL*NET FROM CLIENT',  
  
'SQL*NET ROUNDTRIPS TO/FROM CLIENT', 'SORTS (MEMORY)', 'SORTS (DISK)')  
  
ORDER BY pt.statistic#
```

#####92- Fazendo um import em background para não perder caso a sessão caia
#####

1- criar um script assim:

exemplo: raul_imp.sh

conteúdo:

imp parfile=arquivo_parfile.par

2-conceder permissão de execução para ele:

[oracle@SERVIDOR_LINUX]\$ chmod +x raul_imp.sh

3- executar assim:

[oracle@SERVIDOR_LINUX]\$ nohup /diretorio_do_sh/raul_imp.sh &

4- para verificar:

[oracle@SERVIDOR_LINUX]\$ ps -fu oracle | grep -i imp

oracle 24363 1 0 15:24 ? 00:00:00 /bin/sh /bkprman/raul_imp.sh

Ex de parfile:

userid=user/password@instancia

fromuser=nome_owner

touser=nome_owner

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
buffer=16777216
ignore=y
file=nome_arq_dump.dmp
log=nome_arq_log.log
```

#####93-Verificar quais Objetos utilizam determinado DBLINK #####

--views

```
DECLARE
CURSOR c IS
SELECT *
FROM dba_views where text_length <=40000;
BEGIN
FOR r IN c LOOP
IF upper(r.text) LIKE '%BLPBILL%' THEN
DBMS_OUTPUT.put_line (r.owner||'.'||r.view_name);
END IF;
END LOOP;
END;
```

--Demais objetos

```
DECLARE
CURSOR c IS
SELECT owner,name, text
FROM dba_source where type in ('PROCEDURE','PACKAGE','PACKAGE BODY','TYPE
BODY','TRIGGER','FUNCTION','TYPE');
BEGIN
FOR r IN c LOOP
IF upper(r.text) LIKE '%BLPBILL%' THEN
DBMS_OUTPUT.put_line (r.owner||'.'||r.name);
END IF;
END LOOP;
END;
```

#####94 RECUPERANDO DADOS MODIFICADOS ACIDENTALMENTE através da RECYCLEBIN
E FLASHBACK (table e database) #####

--TEST CASE

```
--- create table
create table raul(nome varchar2(30))
```

--- insert

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
insert into raul values('RAUL')
```

```
insert into raul values('FRANCISCO')
```

```
insert into raul values('ANDRADE')
```

```
delete from raul where nome='RAUL'
```

```
commit;
```

```
SELECT * FROM RAUL
```

```
--- drop table
```

```
drop table RAUL
```

```
---recyclebin
```

```
select * from dba_recyclebin where owner='RANDRADE'
```

```
--Select na tabela é possível a partir do nome dela na recyclebin
```

```
SELECT * FROM "BIN$h5KjKjFI4FLgQwopFo/gUg==$0"
```

```
--1 Utilizando FLASHBACK (Oracle 10G em diante)
```

```
--1.1 Flashback table
```

```
--1.1.1 Tabela toda
```

```
flashback table "BIN$gzC35HeHwELgQwopFo/AQg==$0" to before drop;
```

```
--1.1.2 Determinado tempo
```

```
--Precisa estar habilitado
```

```
alter table raul enable row movement
```

```
flashback table RAUL to timestamp to_timestamp('2010-05-26 20:43:00', 'YYYY-MM-DD  
HH:MI:SS');
```

```
--1.1.3 Determinado SCN
```

```
flashback table RAUL TO SCN 204142131000;
```

```
--Pegar o SCN Atual
```

```
select * from v$database
```

```
--1.2 Flashback database
```

```
--Procedimentos para ativar flashback database
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

01. Desative o Banco;
 02. Monte o banco, se for do tipo RAC, monte no modo exclusivo;
 03. Dê o seguinte comando: `alter system db_flashback_retention_target = 5000;`
Esse comando definirá até onde poderá recuperar o banco. É quantificado em minutos.
 04. Ative o Flashback Database: `alter database flashback on;`
 05. Abra o banco (`alter database open;`) e anote o SCN (System Change Number) na view `V$database` no `Current_SCN` e faça algumas alterações no banco;
 06. Dê um Shutdown e monte o banco novamente;
 07. Agora utilize o comando `flashback database to scn` (SCN anotado antes das alterações sem o parêntese);
 08. Abra o banco no modo read only (`alter database open read only;`) e veja se as alterações que você tinha feito sumiram;
 09. Depois que você confirmar se é nesse ponto que você gostaria de voltar mesmo, abra o banco com `resetlogs` (`alter database open resetlogs;`);
- OBS: Você pode fazer uma recuperação tanto pra frente quanto pra trás no tempo e quantas vezes desejar até abrir o banco com `RESETLOGS`.

--2 VIEWS IMPORTANTES

- * `V$database`: Coluna `Flashback_on`, que mostra se o recurso está ativado.
- * `V$flashback_database_log`: Permite monitorar o `RETENTION_TARGET`, que pode ajudar a estimar a quantidade de espaço exigida na área de recuperação.

Colunas importantes: `FLASHBACK_SCN` (menor SCN permitido), `RETENTION_TARGET`, `FLASHBACK_SIZE` (tamanho da área de recuperação ocupada) e `ESTIMATED_FLASHBACK_SIZE` (Tamanho

estimado de dados).

- * `V$flashback_database_stat`: Manutenção dos dados de logs. Estima a quantidade total de espaço para recuperações futuras.

#####95- Verificando latch free

We can get information about "latch free" or any wait for that matter from `V$EVENT_NAME`. `V$EVENT_NAME` gives brief descriptions of the fields P1, P2 and P3 for a wait

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

event:

```
select * from v$event_name
where name = 'latch free'
```

```
PARAMETER1 PARAMETER2 PARAMETER3
-----
address    number    tries
```

In other words the latch# is P2 aka PARAMETER2. We can go into ASH and find out the id, the latch#, of all latch free waits we had:

```
select p2, count(*)
from v$active_session_history
where event='latch free'
group by p2
```

```
P2 COUNT(*)
-----
127    3556
```

In this case I was only getting waits on latch# 127. What is latch# 127? We can look it up in v\$latchname:

```
select * from v$latchname where latch#=127
```

```
LATCH# NAME
-----
127    simulator lru latch
```

#####96- Aplicando patch do Oracle #####

- 1- Baixar da internet (metalink)
- 2- Salvar em algum local ex /oracle/app/patches
- 3- acessar ele cd /oracle/app/patches
- 4- verificar versão do opatch no prompt digitar
\$ORACLE_HOME/OPatch/opath lsinventory
- 5- Para aplicar
\$ORACLE_HOME/OPatch/ opath apply ou
- # \$ORACLE_HOME/OPatch/ opath auto /endereço do path

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

Observações:

=====

- 1- Se a instância for ASM setar primeiro as variáveis de ambiente do ASM e aplicar nele, depois setar da instância e aplicar nela também.
- 2- Caso não seja aplicado o patch em base apenas com os binários, ou seja, se a base tiver banco, deve-se rodar o DBUA ao final

#####97- Configurando manualmente o Oracle Enterprise Manager #####

```
$ emca -repos create
$ emca -config dbcontrol db
$ emca -config dbcontrol db -repos recreate
$ emca -deconfig dbcontrol db -repos drop -- dropando se necessário
```

RESPONDA AS PERGUNTAS

=====

Listener port number: 1521
Database SID: {sid}
Service name: {serviceName}
Email address for notification: {email@site.com.br}
Email gateway for notification: {smtp}
Password for dbsnmp: xxxxxxx
Password for sysman: xxxxxxx
Password for sys: xxxxxxx

#####98- Contar quantidade de Commits em pl com LOOP #####

```
SELECT a.SID,
       DECODE (b.CLASS, 1, 'User', 2, 'Redo', 4, 'Enqueue', 8, 'Cache', 16, 'OS', 32, 'ParallelServer',
       64, 'SQL', 128, 'Debug', 72, 'SQL & Cache', 40, 'ParallelServer
       & Cache')
CLASS,b.NAME,
       a.VALUE
FROM v$sesstat a, v$statname b
WHERE (a.statistic# = b.statistic#)
      AND SID = '&SID'
      and name = 'user commits'
```

#####99- Voltando Flashback database #####

SQL> shutdown immediate;

Database closed.

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

Database dismounted.

ORACLE instance shut down.

SQL> startup mount exclusive;

ORACLE instance started.

Database mounted.

--Com SQL

SQL> FLASHBACK DATABASE to timestamp to_timestamp('01-01-2007 10:00', 'DD-MM-YYYY HH24:MI:SS');

Flashback complete.

SQL> alter database open resetlogs;

--Com RMAN

1) conectar no servidor (Linux, Unix , AIX) e emitir o seguinte comando:

export NLS_DATE_FORMAT='YYYY-MM-DD HH24:MI:SS'

2) conectar no rman (catálogo + banco) e enviar o comando abaixo, não esquecer de alterar o horário e a string NSR para o cliente correto:

```
run{
allocate channel ch1 type 'sbt_tape' maxpiecesize 10G;
send 'NSR_ENV=(NSR_SERVER=svuxnsr1.teste.net.br,NSR_CLIENT=svuxp1b4.teste.net.br)';
FLASHBACK DATABASE TO TIME='2010-06-29 16:10:00';
release channel ch1;
}
```

#####100- Determina o período de Flashback atual (até onde pode-se voltar o banco)

```
select oldest_flashback_scn,
oldest_flashback_time
from v$flashback_database_log;
```

#####101- Exibe a cota de disco da área de recuperação flash

```
select estimated_flashback_size,
flashback_size
from V$FLASHBACK_DATABASE_LOG;
```


#####102- Mostra as alterações de taxa na geração de dados de flashback

```
select begin_time, end_time, flashback_data, db_data, redo_data, estimated_flashback_size
as EST_FB_SIZE
from V$FLASHBACK_DATABASE_STAT;
```

```
-- flashback_data: numero de bytes de dados de flashback;
-- redo_data:    numero de bytes de dados de redo;
-- db_data:     numero de bytes de blocos de dados lidos e gravados;
-- estimated_flashback_size: espaço de flashback estimado para o período;
-- ambos gravados durante o intervalo de tempo citado.
```

#####103- Gerando AWR pelo SO (Sem EM) e gerando relatório comparativo

```
cd $ORACLE_HOME/rdbms/admin
```

```
sqlplus "/ as sysdba"
```

```
SQL>@awrrpt.sql;
```

```
$ORACLE_HOME/rdbms/admin/awrrpt.sql;
```

depois preenche os dados que ele pede e irá criar um HTML no endereço indicado.

```
-- Caso queira gerar comparativo
```

```
cd $ORACLE_HOME/rdbms/admin
```

```
sqlplus "/ as sysdba"
```

```
SQL>@$ORACLE_HOME/rdbms/admin/awrddrpt.sql;
```

#####104- LOCK DE ENQUEUE

```
--Tipo TM
```

```
SELECT l.sid, s.blocking_session blocker, s.event, l.type, l.lmode, l.request, o.object_name,
o.object_type
FROM v$lock l, dba_objects o, v$session s
WHERE UPPER(s.username) = UPPER('&SCHEMA')
AND l.id1 = o.object_id (+)
AND l.sid = s.sid
ORDER BY sid, type;
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
--Tipo ITL
select s.sid, e.event, s.row_wait_obj#, o.object_name
from v$session s, v$enqueue_lock l, v$session_event e, dba_objects o
where e.sid = l.sid
and e.sid = s.sid
and s.row_wait_obj# = o.object_id (+)
and e.event like 'enq: TX%';
```

#####105- gc current block 2-way

```
SELECT sql_id, COUNT (*)
FROM dba_hist_active_sess_history
WHERE wait_class_id = (SELECT wait_class_id
                       FROM v$event_name
                       WHERE NAME = 'gc current block 2-way')
GROUP BY sql_id
ORDER BY 2 DESC
```

#####106- Verificar a seletividade de uma coluna

-- rodado com script seletividade.sql

```
SET VERIFY OFF
set lines 180 feedback off
```

```
alter session set nls_numeric_characters = '.,'
/
```

accept samples number prompt 'Sample: '

```
column sum for 9999999999 head 'Total #Rows'
column cnt for 9999999999 head 'Total # Dist Values'
column min for 999999 head 'Min # of Rows'
column avg for 999999 head 'Avg # of Rows'
column max for 999999 head 'Max # of Rows'
column bsel for 999999.9999 head 'Best Selectivity[%]'
column asel for 999999.9999 head 'Avg Selectivity[%]'
column wsel for 999999.9999 head 'Worst selectivity[%]'
```

```
select sum(a) sum,
       count(a) cnt,
       min(a) min,
       round(avg(a),1) avg,
       max(a) max,
       round(min(a)/sum(a)*100,4) bsel,
       round(avg(a)/sum(a)*100,4) asel,
       round(max(a)/sum(a)*100,4) wsel
from (select count(1) a from &table_name where rownum <= &samples group by
&column_name)
/
```

set lines 120 feedback on

#####107- Verificar o plano de execução

-- rodado com script: showplan.sql

```
set pages 1000
set lines 120
select * from table(dbms_xplan.display)
/
```

set lines 80

#####108- Verificação de qual objeto de uma sessão vem um wait event

-- Selecciona os p1 e p2 da sessão, sendo p1 o file e p2 o block

```
select
  username,
  event,
  p1,
  p2
from
  v$session_wait
where
  sid = &SID;
```

-- depois encontra o objeto

```
select
  segment_name,
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
segment_type
from
  dba_extents
where
  file_id = &p1
and
  &p2 between
  block_id and block_id + blocks - 1;
```

#####109- Verificar a quantidade de archive gerado por thread e hora
-- Por thread

```
SELECT thread#,
       TRUNC (firsttime),
       TO_CHAR (firsttime, 'Dy'),
       SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '00', SUBSTR (totalsizehora, 4, 10), 0))) "h0",
       SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '01', SUBSTR (totalsizehora, 4, 10), 0))) "h1",
       SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '02', SUBSTR (totalsizehora, 4, 10), 0))) "h2",
       SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '03', SUBSTR (totalsizehora, 4, 10), 0))) "h3",
       SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '04', SUBSTR (totalsizehora, 4, 10), 0))) "h4",
       SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '05', SUBSTR (totalsizehora, 4, 10), 0))) "h5",
       SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '06', SUBSTR (totalsizehora, 4, 10), 0))) "h6",
       SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '07', SUBSTR (totalsizehora, 4, 10), 0))) "h7",
       SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '08', SUBSTR (totalsizehora, 4, 10), 0))) "h8",
       SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '09', SUBSTR (totalsizehora, 4, 10), 0))) "h9",
       SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '10', SUBSTR (totalsizehora, 4, 10), 0)))
"h10",
       SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '11', SUBSTR (totalsizehora, 4, 10), 0)))
"h11",
       SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '12', SUBSTR (totalsizehora, 4, 10), 0)))
"h12",
       SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '13', SUBSTR (totalsizehora, 4, 10), 0)))
"h13",
       SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '14', SUBSTR (totalsizehora, 4, 10), 0)))
"h14",
       SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '15', SUBSTR (totalsizehora, 4, 10), 0)))
"h15",
       SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '16', SUBSTR (totalsizehora, 4, 10), 0)))
"h16",
       SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '17', SUBSTR (totalsizehora, 4, 10), 0)))
"h17",
       SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '18', SUBSTR (totalsizehora, 4, 10), 0)))
"h18",
       SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '19', SUBSTR (totalsizehora, 4, 10), 0)))
"h19",
       SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '20', SUBSTR (totalsizehora, 4, 10), 0)))
"h20",
```

```

SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '21', SUBSTR (totalsizehora, 4, 10), 0)))
"h21",
SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '22', SUBSTR (totalsizehora, 4, 10), 0)))
"h22",
SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '23', SUBSTR (totalsizehora, 4, 10), 0))) "h23"
FROM (SELECT b.thread# "THREAD#",
TO_DATE (b.first_time, 'dd/mm/yy') "FIRSTTIME",
TO_CHAR (b.first_time, 'hh24') || '-' || SUM (ROUND ((b.blocks * b.block_size) /
1024 / 1024)) "TOTALSIZEHORA"
FROM v$archived_log b
GROUP BY b.thread#, TO_DATE (b.first_time, 'dd/mm/yy'), TO_CHAR (b.first_time,
'hh24')) a
GROUP BY thread#, TRUNC (firsttime), TO_CHAR (firsttime, 'Dy')
ORDER BY 2, 1

```

-- Por hora

```

SELECT TRUNC (firsttime) "Data",
TO_CHAR (firsttime, 'Dy') "Dia",
SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '00', SUBSTR (totalsizehora, 4, 10), 0))) "h0",
SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '01', SUBSTR (totalsizehora, 4, 10), 0))) "h1",
SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '02', SUBSTR (totalsizehora, 4, 10), 0))) "h2",
SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '03', SUBSTR (totalsizehora, 4, 10), 0))) "h3",
SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '04', SUBSTR (totalsizehora, 4, 10), 0))) "h4",
SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '05', SUBSTR (totalsizehora, 4, 10), 0))) "h5",
SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '06', SUBSTR (totalsizehora, 4, 10), 0))) "h6",
SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '07', SUBSTR (totalsizehora, 4, 10), 0))) "h7",
SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '08', SUBSTR (totalsizehora, 4, 10), 0))) "h8",
SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '09', SUBSTR (totalsizehora, 4, 10), 0))) "h9",
SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '10', SUBSTR (totalsizehora, 4, 10), 0)))
"h10",
SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '11', SUBSTR (totalsizehora, 4, 10), 0)))
"h11",
SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '12', SUBSTR (totalsizehora, 4, 10), 0)))
"h12",
SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '13', SUBSTR (totalsizehora, 4, 10), 0)))
"h13",
SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '14', SUBSTR (totalsizehora, 4, 10), 0)))
"h14",
SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '15', SUBSTR (totalsizehora, 4, 10), 0)))
"h15",
SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '16', SUBSTR (totalsizehora, 4, 10), 0)))
"h16",
SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '17', SUBSTR (totalsizehora, 4, 10), 0)))
"h17",
SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '18', SUBSTR (totalsizehora, 4, 10), 0)))
"h18",
SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '19', SUBSTR (totalsizehora, 4, 10), 0)))
"h19",

```

```
SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '20', SUBSTR (totalsizehora, 4, 10), 0)))
"h20",
SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '21', SUBSTR (totalsizehora, 4, 10), 0)))
"h21",
SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '22', SUBSTR (totalsizehora, 4, 10), 0)))
"h22",
SUM ((DECODE (SUBSTR (totalsizehora, 1, 2), '23', SUBSTR (totalsizehora, 4, 10), 0))) "h23"
FROM (SELECT b.thread# "THREAD#",
TO_DATE (b.first_time, 'dd/mm/yy') "FIRSTTIME",
TO_CHAR (b.first_time, 'hh24') || '-' || SUM (ROUND ((b.blocks * b.block_size) /
1024 / 1024)) "TOTALSIZEHORA"
FROM v$log b
GROUP BY b.thread#, TO_DATE (b.first_time, 'dd/mm/yy'), TO_CHAR (b.first_time,
'hh24')) a
GROUP BY TRUNC (firsttime), TO_CHAR (firsttime, 'Dy')
ORDER BY 1
```

Variação

```
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') "01",
to_char(sum(decode(to_char(first_time, 'HH24'), '02', 1, 0)), '99') "02",
to_char(sum(decode(to_char(first_time, 'HH24'), '03', 1, 0)), '99') "03",
to_char(sum(decode(to_char(first_time, 'HH24'), '04', 1, 0)), '99') "04",
to_char(sum(decode(to_char(first_time, 'HH24'), '05', 1, 0)), '99') "05",
to_char(sum(decode(to_char(first_time, 'HH24'), '06', 1, 0)), '99') "06",
to_char(sum(decode(to_char(first_time, 'HH24'), '07', 1, 0)), '99') "07",
to_char(sum(decode(to_char(first_time, 'HH24'), '08', 1, 0)), '99') "08",
to_char(sum(decode(to_char(first_time, 'HH24'), '09', 1, 0)), '99') "09",
to_char(sum(decode(to_char(first_time, 'HH24'), '10', 1, 0)), '99') "10",
to_char(sum(decode(to_char(first_time, 'HH24'), '11', 1, 0)), '99') "11",
to_char(sum(decode(to_char(first_time, 'HH24'), '12', 1, 0)), '99') "12",
to_char(sum(decode(to_char(first_time, 'HH24'), '13', 1, 0)), '99') "13",
to_char(sum(decode(to_char(first_time, 'HH24'), '14', 1, 0)), '99') "14",
to_char(sum(decode(to_char(first_time, 'HH24'), '15', 1, 0)), '99') "15",
to_char(sum(decode(to_char(first_time, 'HH24'), '16', 1, 0)), '99') "16",
to_char(sum(decode(to_char(first_time, 'HH24'), '17', 1, 0)), '99') "17",
to_char(sum(decode(to_char(first_time, 'HH24'), '18', 1, 0)), '99') "18",
to_char(sum(decode(to_char(first_time, 'HH24'), '19', 1, 0)), '99') "19",
to_char(sum(decode(to_char(first_time, 'HH24'), '20', 1, 0)), '99') "20",
to_char(sum(decode(to_char(first_time, 'HH24'), '21', 1, 0)), '99') "21",
to_char(sum(decode(to_char(first_time, 'HH24'), '22', 1, 0)), '99') "22",
to_char(sum(decode(to_char(first_time, 'HH24'), '23', 1, 0)), '99') "23"
from v$log_history group by to_char(first_time, 'YYYY-MON-DD');
```

#####109.1 Demonstra o Redolog size recomendado

SELECT

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
(SELECT ROUND(AVG(BYTES) / 1024 / 1024, 2) FROM V$LOG) AS "Redo size (MB)",  
ROUND((20 / AVERAGE_PERIOD) * (SELECT AVG(BYTES)  
FROM V$LOG) / 1024 / 1024, 2) AS "Recommended Size (MB)"  
FROM (SELECT AVG((NEXT_TIME - FIRST_TIME) * 24 * 60) AS AVERAGE_PERIOD  
FROM V$ARCHIVED_LOG  
WHERE FIRST_TIME > SYSDATE -7  
AND TO_CHAR(FIRST_TIME, 'HH24:MI') BETWEEN  
'17:00' AND '20:00');
```

#####110- Conectar como outro usuário

```
-----  
--  
-- Script: su.sql  
-- Purpose: to connect as another user without knowing their password  
--  
-- Author:  
-----
```

```
set termout off  
set verify off  
set echo off  
set pagesize 0  
set linesize 80  
column line1 format a79  
column line2 format a79  
column line3 format a79
```

```
spool su.tmp  
select  
'alter user &1 identified by pwsenha;' line1,  
'connect &1/pwsenha@'||i.instance_name|| chr(10) line2,  
'alter user &1 identified by values "' || u.password || "';' line3  
from  
sys.dba_users u,  
sys.v_$instance i  
where  
u.username = upper('&1')  
/  
spool off
```

```
@su.tmp  
--host rm -f su.tmp -- for Unix  
--host del su.tmp -- for others
```

```
define Prompt = "SQL> "
```

```
set termout off
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
select
  user || ' @ ' || instance_name || ':' || chr(10) || 'SQL> ' prompt
from
  sys.v_$instance
/
set termout on
set sqlprompt "&Prompt"
show user
set termout on
```

-- para executar é:
SQL> @su NOME_DO_USUARIO

#####111- Armazenar tabela no Buffer para resolver problemas de I/O
-- pode ser que alguma tabela seja responsável por grande volume de I/O na base de dados
então pode-se "piná-la" armazená-la em cache para que não se tenha I/O.

```
alter system set db_keep_cache_size = tamanho_do_cache scope=both;
alter table activia.act_ordem_servico storage (buffer_pool keep);
```

--Obs: o SGA_MAX_SIZE deve ser suficiente para comportar o db_keep_cache_size

#####112- Verificar registros duplicados

```
select CORRIDOR_PLAN_ID, TYPE_ID_USG, UNITS_INDICATOR, ELEMENT_ID, RATE_CLASS,
EQUIP_TYPE_CODE, EQUIP_CLASS_CODE, CLASS_OF_SERVICE_CODE, PROVIDER_CLASS,
BILL_CLASS,
JURISDICTION, CURRENCY_CODE, RATE_PERIOD, ACTIVE_DT, DISTANCE_BAND_ID,
COMPONENT_ID --> columns from your PK
from ARBOR.RATE_USAGE_OVERRIDES --> your source data that you use for update
group by CORRIDOR_PLAN_ID, TYPE_ID_USG, UNITS_INDICATOR, ELEMENT_ID, RATE_CLASS,
EQUIP_TYPE_CODE, EQUIP_CLASS_CODE, CLASS_OF_SERVICE_CODE, PROVIDER_CLASS,
BILL_CLASS,
JURISDICTION, CURRENCY_CODE, RATE_PERIOD, ACTIVE_DT, DISTANCE_BAND_ID,
COMPONENT_ID --> columns from your PK
having count(*)>1;
```

#####113- Verificar usuário tentando conectar com senha incorreta

1- Alter user xx account unlock;

2- Set trace para o evento:

ORA-01017: invalid username/password; logon denied

```
ALTER SYSTEM SET EVENTS='1017 trace name errorstack level 3';
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

3- Acompanha alert e trace

4- Disable.

```
alter system set events '1017 trace name errorstack off';
```

#####114- Atualizar Materialized View (SNAPSHOT)

```
execute dbms_snapshot.refresh('NOME' VARCHAR2)
```

-- para verificar a atualização da view e o SCN do refresh

```
select * from DBA_MVIEW_ANALYSIS where MVIEW_NAME='&MVIEW_NAME'
```

#####115- Verificar JOB's rodando

```
select a.username, a.osuser, a.machine, a.sid, a.serial#, b.spid, c.job, c.what  
from v$session a, v$process b, dba_jobs c, dba_jobs_running d  
where a.paddr=b.addr and c.job=d.job and a.sid in (select sid from dba_jobs_running);
```

--Variação:

```
select j.job, j.what, s.sid, s.serial#, p.spid, s.username, s.osuser from dba_jobs j  
inner join dba_locks l on j.job = l.lock_id2 and l.lock_type = 'JQ'  
inner join v$session s on s.sid = l.session_id inner join v$process p on p.addr = s.paddr
```

#####116- Resolver problema quando sessão cai enquanto está criando o índice e ele aparece na dba_indexes e não na dba_segments e demais (índice é criado pela

metade) Oracle 10G 10.2.0.X

--1. Localizar o OBJECT_ID

```
select i.obj#, i.flags, u.name, o.name, o.type#  
from sys.obj$ o, sys.user$ u, sys.ind$ idx, sys.ind_online$ i  
where o.obj#=idx.obj# and o.owner# = u.user# and idx.obj#=i.obj#;
```

--2. rodar a function abaixo com o valor do OBJECT_ID pego pela query acima

```
DECLARE  
retval BOOLEAN;
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
object_id BINARY_INTEGER;  
wait_for_lock BINARY_INTEGER;  
BEGIN  
object_id := &OBJECT_ID;  
wait_for_lock := NULL;  
retval := SYS.Dbms_repair.online_index_clean(object_id);  
COMMIT;  
END;  
/
```

#####117- Comando para manutenção de banco bloqueando acessos sem parar o banco.

1) ALTER SYSTEM ENABLED RESTRICTED SESSION

2) ALTER SYSTEM QUIESCE DATABASE

O primeiro comando bloqueia sessões que não possuem privilegio de restricted session.

A sessões que já estão conectadas permanecem conectadas e podendo fazer qualquer operação.

O dba portanto deverá fazer um kill nestas sessões ou então fazer um re-start do banco.

O segundo comando impede qualquer transação não dba, bloco PL-sql e consultas de iniciarem.

A vantagem do quiesce é exatamente não precisar desligar o banco, uma restrição para ambientes 24x7, sem falar que no startup database buffer cache e shared pool teriam que novamente ser carregados.

#####118- Habilitando Block change tracking para otimizar backups incrementais

```
ALTER DATABASE  
ENABLE BLOCK CHANGE TRACKING  
USING FILE os_file_name;
```

Se omitir a cláusula USING_FILE habilita o OMF (Oracle Manage Files).

-- Para verificar:

```
Select * from v$block_change_tracking
```

#####119- verifica a porcentagem de processos em relação à quantidade máxima de processos permitidos pelo parâmetro de inicialização PROCESSES

```
SELECT TRUNC (100 * (SELECT COUNT (1) FROM v$process) / (SELECT CAST (VALUE AS
INTEGER)
FROM v$parameter WHERE NAME = 'processes'), 2) AS pct_processes
FROM DUAL;
```

119.1- Para matar sessões rodando a muito tempo:

```
SELECT 'kill -9 ' || SPID
FROM V$PROCESS WHERE ADDR IN
(SELECT PADDR FROM V$SESSION WHERE USERNAME = 'FOR_WEB' and last_call_et>=3600)--
em segundos
```

--Variação que gera o kill -9

```
select
  '/' * ' ' || a.username || ' Login: ' || to_char(a.logon_time,'DD/MM/YYYY HH24:MI') ||
  ' Idle Time: ' || round(last_call_et/60,2) || 'M */' || ' '
alter system kill session ' ' || ' ' || a.sid || ' ' || a.serial# || ' ' || ' ' || ' '
!kill -9 ' ' || b.spid
from v$session a, v$process b
where a.paddr = b.addr
and a.username not in ('SYSTEM','SYS')
and length(rtrim(a.username)) > 0
and last_call_et/60/60 >= 1;
```

-- Para contar as sessões ativas:

```
select count(1) sessoes_ativas
from gv$session
where username is not null
and status = 'ACTIVE'
and sid in (select sid
            from gv$session_wait
            where event not like 'Streams AQ%'
            and event not like 'queue messages%'
            and event not like 'SQL*Net%'
            and event not like 'rdbms ipc message');
```

#####120- Descobrir os SPID e gerar kill -9 para usuários que estejam paradas a mais de 60 minutos (3600 segundos)

```
SELECT 'kill -9 ' || SPID
FROM V$PROCESS WHERE ADDR IN
(SELECT PADDR FROM V$SESSION WHERE USERNAME = 'FOR_WEB' and last_call_et>=3600)
```

-- pelo SERIAL#

```
SELECT 'kill -9 ' || SPID
FROM V$PROCESS WHERE SERIAL# IN
(SELECT serial# FROM V$SESSION WHERE last_call_et>=3600 and username <>'ORACLE' and
status<>'KILLED')
```

#####121- Mostra os objetos que estão sendo lidos fisicamente e causando o db sequential read

```
select b.sid,
       nvl(substr(a.object_name,1,30),
           'P1='||b.p1||' P2='||b.p2||' P3='||b.p3) object_name,
       a.subobject_name,
       a.object_type
from   dba_objects a, v$session_wait b, x$bh c
where  c.obj = a.object_id(+)
and    b.p1 = c.file#(+)
and    b.p2 = c.dbablk(+)
and    b.event = 'db file sequential read'
union
select b.sid,
       nvl(substr(a.object_name,1,30),
           'P1='||b.p1||' P2='||b.p2||' P3='||b.p3) object_name,
       a.subobject_name,
       a.object_type
from   dba_objects a, v$session_wait b, x$bh c
where  c.obj = a.data_object_id(+)
and    b.p1 = c.file#(+)
and    b.p2 = c.dbablk(+)
and    b.event = 'db file sequential read'
order by 1;
```

#####122- Encontrar qual o valor atribuído para determinada bind

-- Serve para verificar quais os valores o Oracle está usando de BIND variable.

-- Modelo 1

```
select * from DBA_HIST_SQLBIND
where sql_id in (select sql_id from v$session where SID = 3822)
order by last_captured
```

-- Modelo 2

```
select * from sys.V_$SQL_BIND_CAPTURE
where sql_id in (select sql_id from v$session where SID = 1660)
order by last_captured
```

```
SELECT *
FROM v$sql_bind_capture
WHERE address = (SELECT sql_address
                 FROM v$session
                 WHERE SID = 1617)
```

--Variação

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
SELECT b.name, b.value_string, sq.sql_text
FROM v$sql_bind_capture b, v$sql sq, v$session s
WHERE sq.sql_id = b.sql_id
AND sq.address = b.address
AND sq.child_address = b.child_address
AND sq.parsing_user_id = s.user#
AND s.username = '&USERNAME'
and s.OSUSER = '&OSUSER'
and sq.sql_id = '&SQL_ID's
ORDER BY sq.sql_id, sq.address, sq.child_address, b.position;
```

#####123- Trigger para monitorar registro modificado via rowid

Case:

Criação da tabela verifica

```
CREATE TABLE RANDRADE.VERIFICA
(
  ROW_ID VARCHAR2(4000 BYTE)          NULL,
  DATA  DATE                        NULL
)
```

--Criação da trigger:

```
CREATE OR REPLACE TRIGGER verifica_update
AFTER UPDATE
ON randrade.raul
REFERENCING NEW AS NEW OLD AS OLD
FOR EACH ROW
DECLARE
  v_rowid ROWID;
BEGIN
  INSERT INTO verifica
    (row_id,
     DATA
    )
  VALUES (:OLD.ROWID,
          SYSDATE
          );
END;
```

--Utilização:

update raul set nome = 'RAUL' where codigo = 001

--Resultado:

AAAZLQAAEAAAABcAAA 01/11/2010 10:38:30

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

#####124- Melhora de performance com objetos colocados keep na shared_pool

Certas vezes é performático deixar os objetos mais acessados "pinados" na shared_pool, evitando-se assim que o banco precise procurá-los fazendo parse toda

vez que eles forem solicitados.

a- Para verificar os comandos que estão na shared_pool no momento user a view v\$sql_area;

b- Para verificar os objetos de banco de dados que estão em cache "pinados" use a view v\$db_object_cache;

QUERY:select owner,name from v\$db_object_cache where kept='YES';

c- Para verificar os objetos mais acessados e o quanto eles gastam em memória use também a view v\$db_objects_cache, onde:

executions; indica a quantidade de acessos que essa query/comando teve.

sharabale_man; tamanho do comando em memória, em bytes. Mostra quanto de espaço o comando usa na memória.

d- Limpara a SHARED_POOL

ALTER SYSTEM FLUSH SHARED_POOL;

--d1.Flush mantendo keep sequences

declare

cursor cached_sequences is

select

sequence_owner,

sequence_name

from

sys.dba_sequences

where

cache_size > 0;

sequence_owner varchar2(30);

sequence_name varchar2(30);

begin

open cached_sequences;

loop

fetch cached_sequences into sequence_owner, sequence_name;

exit when cached_sequences%notfound;

sys.dbms_shared_pool.keep(sequence_owner || '.' || sequence_name, 'Q');

end loop;

end;

/

e- Colocar objetos na SHARED_POOL:

EXECUTE DBMS_SHARED_POOL.KEEP('OWNER.OBJETO');

f- Tirar objetos da SHARED_POOL:

```
EXECUTE DBMS_SHARED_POOL.UNKEEP('OWNER.OBJETO');
```

g- Para ver o tamanho da shared_pool faça:

```
show parameter shared_pool_size
```

h- Para alterar o tamanho da SHARED_POOL

```
alter system set shared_pool_size=300M
```

i- Para colocar uma query pinada:

```
EXECUTE DBMS_SHARED_POOL.KEEP('address,hash_value','C');
```

Ex:

```
EXECUTE DBMS_SHARED_POOL.KEEP('AB5CC7C4,1616276262','C');
```

Para procurar qual o address e hash_value:

```
select address, hash_value from v$sqlarea where sql_id='SQL_ID *'
```

* Escolhido pelo relatório do AWR por exemplo.

j- Para verificar a utilização da SGA faça o comando do item 27

#####125- Verificar principais recursos da máquina

```
select *
from v$resource_limit
where resource_name in ('enqueue_resources','enqueue_locks',
'dml_locks','processes','sessions');
```

#####126- Verificar início e final de processo para tomada de tempo

---Método 1

```
set timing on;
```

```
SQL> SELECT order_id
FROM po_orders
WHERE branch_id = 6
```

```
ORDER_ID
```

```
-----
```

```
232
```

```
Elapsed: 00:00:13.26
```

---Método 2

```
BEGIN
```

```
DBMS_OUTPUT.put_line ('INICIO = ' || TO_CHAR (SYSDATE, 'YYYY/MM/DD HH24:MI:SS'));
```

```
END;
```

```
--INSTRUÇÃO QUE VC DESEJA, EX:
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
SELECT * FROM HR.EMPLOYEES;
```

```
BEGIN
```

```
DBMS_OUTPUT.put_line ('FIM = ' || TO_CHAR (SYSDATE, 'YYYY/MM/DD HH24:MI:SS'));
```

```
END;
```

-- Com isto, rodando em um script com Spool, ou via toad ele demonstra o horário inicial e o final.

#####127- Trigger para detectar tentativas de logon frustradas

-- Cria-se a tabela:

-- LOGONUNSUCCESSFUL (Table)

--

```
CREATE TABLE SYS.LOGONUNSUCCESSFUL
```

```
(
```

```
  HOST  VARCHAR2(150 BYTE)          NULL,
```

```
  IP    VARCHAR2(15 BYTE)           NULL,
```

```
  OSUSER VARCHAR2(150 BYTE)         NULL,
```

```
  DATA DATE                        NULL
```

```
);
```

-- Depois a trigger

```
CREATE OR REPLACE TRIGGER SYS.LOGON_UNSUCCESSFUL AFTER SERVERERROR ON
DATABASE
```

```
DECLARE
```

```
  IP varchar2(15);
```

```
  os_user varchar2(80);
```

```
  host varchar2(100);
```

```
BEGIN
```

```
  SELECT sys_context('userenv','host') into host FROM dual;
```

```
  SELECT sys_context('userenv','ip_address') into IP FROM dual;
```

```
  SELECT sys_context('userenv','os_user') into os_user FROM dual;
```

```
  IF (ora_is_servererror(1017)) THEN
```

```
    insert into teste.logonunsuccessful(host,ip,osuser,data) values(host, ip || ora_login_user,
os_user, sysdate);
```

```
  END IF;
```

```
END;
```

```
/
```

#####128- Criar script com todos os grants da base por usuários e base toda

```
SELECT 'grant ' || PRIVILEGE || ' on ' || owner || '.' || table_name || ' to ' || grantee || ';'
FROM dba_tab_privs
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle


```
WHERE owner NOT IN ('SYS', 'CTXSYS', 'SYSTEM', 'OUTLN', 'DBSNMP', 'WMSYS', 'EXFSYS',  
'DMSYS', 'XDB', 'ORDSYS', 'ORDPLUGINS', 'MDSYS', 'OLAPSYS', 'SYSMAN')
```

#####129- Trabalhando com sequence

--- Alterando a sequence (increment)

--1- coloca o increment no valor desejado:

```
alter sequence arborgvt_billing.SQ_GVT_EXEC_ARG increment by &valor_desejado;
```

--2- faz o select para verificar e assumir o valor:

```
select arborgvt_billing.SQ_GVT_EXEC_ARG.nextval from dual;
```

--3-Volta o increment para 1 para que vá de 1 em 1

```
alter sequence arborgvt_billing.SQ_GVT_EXEC_ARG increment by 1;
```

--4- faz o select para verificar e assumir o valor:

```
select arborgvt_billing.SQ_GVT_EXEC_ARG.nextval from dual
```

--- Resetar a sequence:

```
ALTER SEQUENCE arborgvt_billing.SQ_GVT_EXEC_ARG INCREMENT BY -&valor_do_incr  
MINVALUE 0;
```

#####130- Procedimento para compilar objeto em uso

-- Checar dependencias

```
select owner, name, type  
from dba_dependencies  
where referenced_owner = 'SIEBPRD'  
and referenced_name = 'GVT_PKG_PACOTE'  
and referenced_type = 'PACKAGE';
```

-- 1. Identificar o SID de quem vai compilar o pkg

```
select s.sid, s.serial#, p.spid, s.status, s.username, s.osuser, s.machine, s.program  
from v$session s, v$process p  
where s.paddr = p.addr  
and s.username = 'RANDRADE'  
and s.machine = 'WORKGROUP/CTHQ5-2836';
```

-- 2. Validar exclusao do SID de quem vai compilar o pkg

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
select s.sid, s.serial#, p.spid, s.status, s.username, s.osuser, s.machine, s.program
from v$session s, v$process p
where s.paddr = p.addr
and s.sid <> 2915      /* Desconsiderar este SID -- Id de quem esta compilando o pkg */
and s.sid in (select sid from v$access
where owner = '&OWNER'
and type = '&TIPO_OBJETO'
and sid <> 2915      /* Desconsiderar este SID -- Idem acima */
and object = '&NOME_OBJETO');
```

-- 3. Proceder a compilacao a partir de outra sessão

-- 4. Executar kill nas demais sessões

```
select 'kill -9 ' || p.spid
from v$session s, v$process p
where s.paddr = p.addr
and s.sid <> 2915      /* Desconsiderar este SID -- Id de quem esta compilando o pkg em
outra sessao */
and s.sid in (select sid from v$access
where owner = '&OWNER'
and type = '&TIPO_OBJETO'
and sid <> &SID_VAI_EXECUTAR      /* Desconsiderar este SID -- Idem acima */
and object = '&NOME_OBJETO');
```

131- Verificar quantidade média (não exata) de registros inseridos durante um insert.

-- Se faz necessário ter estatísticas coletadas.

-- O valor informado não será exato e sim estimado

-- serve para tabelas vazias, caso contrário proceder antes de iniciar o insert guardar o valor depois subtrair e considerar a diferença

```
SELECT
trunc((SELECT BYTES
FROM dba_segments
WHERE segment_name = 'ANALISE_RECEITAS_TERCEIRO') /
(SELECT avg_row_len FROM dba_tables WHERE table_name =
'ANALISE_RECEITAS_TERCEIRO'))
FROM DUAL;
```

#####132- Fazer carga de arquivo texto em tabela do banco.

-- Test Case:

-- Tabela:

```
CREATE TABLE teste_cluster_uso_a9
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
( bill_period          VARCHAR2 (003),
  account_category     NUMBER,
  element_id           NUMBER,
  component_id         NUMBER,
  type_code            NUMBER,
  subtype_code         NUMBER,
  tipo_registro        NUMBER,
  description_code     NUMBER,
  amount              NUMBER,
  icms                 NUMBER,
  piscofins            NUMBER,
  data_ref             VARCHAR2 (006),
  external_id_conta    VARCHAR2 (012),
  external_id_fone     VARCHAR2 (020));
```

-- PL/Lê do arquivo texto em um Oracle Directory e grava no banco

```
/* Formatted on 2010/12/15 14:01 (Formatter Plus v4.8.6) */
```

```
SET verify      off;
SET serverout   on;
SET feed        off;
SET space       0;
SET pagesize    0;
SET line        500;
SET wrap        on;
SET heading     off;
```

DECLARE

```
  v_dirspool      VARCHAR2 (60) := 'VERIFICA_DADOS'; --
/app/billing/svuxd1b4/cobilling/Interfaces/log';
  v_nome_arquivo  VARCHAR2 (60) := 'A9_CLUSTER_RC_NRC_A_FATURAR.TXT'; --
'CLUSTER_RC_NRC_A_FATURAR_20101102234524.txt';
-- v_nome_arquivos VARCHAR2 (60) :=
'RM_CLUSTER_RC_NRC_A_FATURAR_20101102234524.txt';
  v_arquivo       UTL_FILE.file_type;
-- v_arquivos     UTL_FILE.file_type;
  v_linha         VARCHAR2 (200);
  v_lidos         NUMBER := 0;
  v_gravados      NUMBER := 0;
  v_correcao      NUMBER := 0;
  v_bill_period   VARCHAR2 (003);
  v_account_category NUMBER;
  v_element_id    NUMBER;
  v_component_id  NUMBER;
  v_type_code     NUMBER;
  v_subtype_code  NUMBER;
  v_tipo_registro NUMBER;
  v_description_code NUMBER;
  v_amount        NUMBER;
  v_icms          NUMBER;
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```

v_pisconfins          NUMBER;
v_data_ref            VARCHAR2 (006);
v_external_id_conta    VARCHAR2 (012);
v_external_id_fone     VARCHAR2 (020);
BEGIN
  DBMS_OUTPUT.put_line ('VAI ABRIR ' || v_dirspool || '/' || v_nome_arquivo);
  v_arquivo             := UTL_FILE.fopen (v_dirspool, v_nome_arquivo, 'r', 10000);
  DBMS_OUTPUT.put_line ('ABRIU ENTRADA ');
  v_lidos               := 0;

  --- DBMS_OUTPUT.put_line ('VAI ABRIR ' || v_dirspool || '/' || v_nome_arquivos);
  --- v_arquivos          := UTL_FILE.fopen (v_dirspool, v_nome_arquivos, 'w', 10000);
  --- DBMS_OUTPUT.put_line ('ABRIU SAIDA ');
  LOOP
    UTL_FILE.get_line (v_arquivo, v_linha);
    --dbms_output.put_line(v_linha);
    --dbms_output.put_line('v_tamanho linha ' || length(v_linha));
    v_lidos              := v_lidos + 1;
    v_bill_period        := SUBSTR (v_linha, 1, 3);
    --dbms_output.put_line('1 ' || chr(39) || v_bill_period || chr(39) || ' tam = '
    ' || length(v_bill_period));
    v_account_category   := SUBSTR (v_linha, 4, 4);
    --dbms_output.put_line('2 ' || chr(39) || v_account_category || chr(39) || ' tam = '
    ' || length(v_account_category));
    v_element_id         := SUBSTR (v_linha, 8, 6);
    --dbms_output.put_line('3 ' || chr(39) || v_element_id || chr(39) || ' tam = '
    ' || length(v_element_id));
    v_component_id       := SUBSTR (v_linha, 14, 10);
    --dbms_output.put_line('4 ' || chr(39) || v_component_id || chr(39) || ' tam = '
    ' || length(v_component_id));
    v_type_code          := SUBSTR (v_linha, 24, 4);
    --dbms_output.put_line('5 ' || chr(39) || v_type_code || chr(39) || ' tam = '
    ' || length(v_type_code));
    v_subtype_code       := SUBSTR (v_linha, 28, 8);
    --dbms_output.put_line('6 ' || chr(39) || v_subtype_code || chr(39) || ' tam = '
    ' || length(v_subtype_code));
    v_tipo_registro      := SUBSTR (v_linha, 36, 1);
    --dbms_output.put_line('7 ' || chr(39) || v_tipo_registro || chr(39) || ' tam = '
    ' || length(v_tipo_registro));
    v_description_code    := SUBSTR (v_linha, 37, 8);
    --dbms_output.put_line('8 ' || chr(39) || v_description_code || chr(39) || ' tam = '
    ' || length(v_description_code));
    v_amount             := SUBSTR (v_linha, 45, 17);
    --dbms_output.put_line('9 ' || chr(39) || v_amount || chr(39) || ' tam = ' || length(v_amount));
    v_icms               := SUBSTR (v_linha, 62, 10);
    --dbms_output.put_line('10 ' || chr(39) || v_icms || chr(39) || ' tam = ' || length(v_icms));
    v_pisconfins         := SUBSTR (v_linha, 72, 10);
    --dbms_output.put_line('11 ' || chr(39) || v_pisconfins || chr(39) || ' tam = '
    ' || length(v_pisconfins));
    v_data_ref           := SUBSTR (v_linha, 82, 6);

```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
--dbms_output.put_line('12 ' || chr(39) || v_data_ref || chr(39) || ' ' tam =
'||length(v_data_ref));
    v_external_id_conta      := SUBSTR (v_linha, 88, 12);
--dbms_output.put_line('13 ' || chr(39) || v_external_id_conta || chr(39) || ' ' tam =
'||length(v_external_id_conta));
    v_external_id_fone       := SUBSTR (v_linha, 100);

--dbms_output.put_line('14 ' || chr(39) || v_external_id_fone || chr(39) || ' ' tam =
'||length(v_external_id_fone));
    BEGIN
--dbms_output.put_line('antes insert ');
    INSERT INTO gvt_cluster_rc_nrc_a9
        VALUES (v_bill_period,
            v_account_category,
            v_element_id,
            v_component_id,
            v_type_code,
            v_subtype_code,
            v_tipo_registro,
            v_description_code,
            v_amount,
            v_icms,
            v_piscofins,
            v_data_ref,
            v_external_id_conta,
            v_external_id_fone
        );

--    DBMS_OUTPUT.put_line ('antes commit ');
    COMMIT;
    v_gravados              := v_gravados + 1;
---    UTL_FILE.fflush (v_arquivos);
    EXCEPTION
        WHEN OTHERS
        THEN
            DBMS_OUTPUT.put_line ('ORA = ' || SQLERRM);
            DBMS_OUTPUT.put_line ('REG = ' || v_lidos);
    END;
END LOOP;

    UTL_FILE.fclose_all;
EXCEPTION
    WHEN UTL_FILE.invalid_path
    THEN
        DBMS_OUTPUT.put_line ('Verif a exist do param. UTL_FILE_DIR no init');
    WHEN UTL_FILE.invalid_operation
    THEN
        DBMS_OUTPUT.put_line ('invalid_operation');
    WHEN NO_DATA_FOUND
    THEN
```

```

DBMS_OUTPUT.put_line ('LIDOS  = ' || v_lidos);
DBMS_OUTPUT.put_line ('GRAVADOS = ' || v_gravados);
DBMS_OUTPUT.put_line ('CORRECAO = ' || v_correcao);
WHEN OTHERS
THEN
    DBMS_OUTPUT.put_line (v_linha);
    DBMS_OUTPUT.put_line ('sqlerrm ' || SQLERRM);
    DBMS_OUTPUT.put_line ('linha lidos = ' || v_lidos || ' gravados ' || v_gravados);
END;
```

#####133- DEBUG Oracle

1- através do SO quando não consegue conectar

a) Pega o PID da Sessão

```
$ps -ef |grep $ORACLE_SID ## Use PID of any User Process instead of BG Process
```

a.1) Pode também pegar com o comando top do SO

b) Roda o GDB

```
$ /usr/bin/gdb gdb $ORACLE_HOME/bin/oracle <PID_from_earlier_step>
```

c) Printa o relatório

```
print ksudss(10)
```

2- Conectado no banco

```
export ORACLE_SID=PROD ## Replace PROD with the SID you want to trace
```

```
sqlplus -prelim / as sysdba
```

```
oradebug setmypid
```

```
oradebug unlimit;
```

```
oradebug dump systemstate 10
```

3- Via SCRIPT

```
script stack.log
```

```
pstack <pid_of_pmon>
```

```
pstack <pid_of_smon>
```

#####134- Segment advisor (verifica a necessidade de shrink)

```
-- Description : Displays segment advice for the specified segment.
```

```
-- Requirements : Access to the DBMS_ADVISOR package.
```

```
-- Call Syntax : Object-type = "tablespace":
```

```
--          @segment_advisor.sql tablespace (tablespace-name) null
```

```
--          Object-type = "table" or "index":
```

```
--          @segment_advisor.sql (object-type) (object-owner) (object-name)
```

```
SET SERVEROUTPUT ON SIZE 1000000
```

```
SET LINESIZE 200
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

SET VERIFY OFF

DECLARE

```
l_object_id  NUMBER;
l_task_name  VARCHAR2(32767) := 'SEGMENT_ADVISOR_TASK';
l_object_type VARCHAR2(32767) := UPPER('&1');
l_attr1      VARCHAR2(32767) := UPPER('&2');
l_attr2      VARCHAR2(32767) := UPPER('&3');
```

BEGIN

```
IF l_attr2 = 'NULL' THEN
  l_attr2 := NULL;
END IF;
```

```
DBMS_ADVISOR.create_task (
  advisor_name => 'Segment Advisor',
  task_name    => l_task_name);
```

```
DBMS_ADVISOR.create_object (
  task_name => l_task_name,
  object_type => l_object_type,
  attr1      => l_attr1,
  attr2      => l_attr2,
  attr3      => NULL,
  attr4      => 'null',
  attr5      => NULL,
  object_id  => l_object_id);
```

```
DBMS_ADVISOR.set_task_parameter (
  task_name => l_task_name,
  parameter => 'RECOMMEND_ALL',
  value     => 'TRUE');
```

```
DBMS_ADVISOR.execute_task(task_name => l_task_name);
```

```
FOR cur_rec IN (SELECT f.impact,
                      o.type,
                      o.attr1,
                      o.attr2,
                      f.message,
                      f.more_info
                  FROM   dba_advisor_findings f
                  JOIN   dba_advisor_objects o ON f.object_id = o.object_id AND f.task_name =
o.task_name
                  WHERE f.task_name = l_task_name
                  ORDER BY f.impact DESC)
LOOP
  DBMS_OUTPUT.put_line('..');
  DBMS_OUTPUT.put_line('Type      : ' || cur_rec.type);
  DBMS_OUTPUT.put_line('Attr1    : ' || cur_rec.attr1);
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```

DBMS_OUTPUT.put_line('Attr2      : ' || cur_rec.attr2);
DBMS_OUTPUT.put_line('Message    : ' || cur_rec.message);
DBMS_OUTPUT.put_line('More info   : ' || cur_rec.more_info);
END LOOP;

DBMS_ADVISOR.delete_task(task_name => l_task_name);
EXCEPTION
WHEN OTHERS THEN
    DBMS_OUTPUT.put_line('Error      : ' || DBMS_UTILITY.format_error_backtrace);
    DBMS_ADVISOR.delete_task(task_name => l_task_name);
END;
/

```

#####135- Descobrir os Hight Water Mark do banco

```

-----
-- Description : Displays high water mark statistics.
-- Requirements : Access to the DBA views.
-- Call Syntax : @db_usage_hwm
-----

```

```

COLUMN name FORMAT A40
COLUMN highwater FORMAT 999999999999
COLUMN last_value FORMAT 999999999999
SET PAGESIZE 24

SELECT hwm1.name,
       hwm1.highwater,
       hwm1.last_value
FROM dba_high_water_mark_statistics hwm1
WHERE hwm1.version = (SELECT MAX(hwm2.version)
FROM dba_high_water_mark_statistics hwm2
WHERE hwm2.name = hwm1.name)
ORDER BY hwm1.name;

```

COLUMN FORMAT DEFAULT

#####136- Exemplo de PL para criar commit a cada quantidade específica de registros

```

set verify      off;
set serverout   on size 1000000;
set feed        off;
set space       0;
set pagesize    0;
set line        500;
set wrap        on;
set heading     off;

```



```
--
*****
*****
--Parametros: sem
--
*****
*****
```

DECLARE

```
wProc      varchar2(20);    -- controle erro: nome proced. em execucao
v_cont     NUMBER:=0;
V_CONT_TOT NUMBER:=0;
V_CONT_ERR NUMBER:=0;
```

CURSOR c1 IS

```
SELECT JD.rowid row_id FROM ARBOR.JNL_DETAIL JD
WHERE JD.JNL_REF_NO in(2663,2664);
```

BEGIN

```
dbms_output.put_line('abriu o arquivo de entrada');
```

FOR r1 IN c1 LOOP

BEGIN

```
DELETE ARBOR.JNL_DETAIL JD
WHERE r1.row_id =jd.rowid;
```

```
v_cont:=v_cont+1;
v_cont_tot := v_cont_tot+1;
```

```
if v_cont > 5000 then
  v_cont:=0;
  commit;
end if;
```

EXCEPTION

```
WHEN NO_DATA_FOUND then
  V_CONT_ERR:=V_CONT_ERR+1;
  dbms_output.put_line('Erro Loop - Nco encontrado dados');
WHEN OTHERS then
  V_CONT_ERR:=V_CONT_ERR+1;
  dbms_output.put_line('Erro Loop ->' || sqlerrm || wproc);
```

END;

END LOOP;

commit;

```
dbms_output.put_line(' TERMINO DO PROCESSAMENTO OK - ' || SYSDATE || 'Total de linhas:
' || v_cont_tot);
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
dbms_output.put_line(' Erros encontrados      - ' || SYSDATE || 'Total de erros : ' ||
v_cont_err);
```

EXCEPTION

```
WHEN UTL_FILE.INVALID_PATH then
    dbms_output.put_line('Erro 6 -> UTL_FILE.INVALID_PATH');
WHEN UTL_FILE.INVALID_OPERATION then
    dbms_output.put_line('Erro 7 -> INVALID_OPERATION');
WHEN UTL_FILE.WRITE_ERROR then
    dbms_output.put_line('Erro 8 -> WRITE_ERROR');
when OTHERS then
    dbms_output.put_line('Erro 9 ->' || sqlerrm || wproc);
END;
/
```

#####137- instalando Auditoria de banco de dados (by Burlesson com adaptações):

Oracle auditing setup & configuration

The Oracle audit command is used to to create specific audit trails for Oracle tables.

To enable Oracle auditing you must set the initialization parameter "audit_trail = true" and run the cataudit.sql scripts (as SYS).

Observar os parâmetros:

Exemplos abaixo:

```
audit_file_dest      string    /oracle/PRD/audit
audit_sys_operations  boolean  FALSE
audit_syslog_level   string
audit_trail          string    OS
```

No exemplo acima vai gerar logs de auditoria no diretório físico /oracle/PRD/audit e não nas tabelas de log de auditoria.

É importante verificar que os logs devem ser apagados ou copiados para outro local de tempos em tempos evitando estouro de área física.

Managing Oracle audit trails

The Oracle audit command write the audit information to specific data dictionary views.

BEWARE: These audit trail rows can quickly fill-up your SYSTEM tablespace, and special care needs to be taken to ensure that you do not "lock-up" your database, by filling the SYSTEM tablespace. Check these tables:

```
?    dba_audit_exists
?    dba_audit_object
?    dba_audit_session
?    dba_audit_statement
?    dba_audit_trail
```

We also have these metadata views for Oracle auditing options:

```
?    dba_obj_audit_opts
?    dba_priv_audit_opts
?    dba_stmt_audit_opts
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

Auditing the audit trail

Today, we need to audit the audit trail itself to prevent "inside jobs" and this Oracle command will record all changes to the audit trail.

```
audit
  delete,
  insert,
  update
on
  sys.aud$
by access;
```

Auditing user activity with the Oracle audit command

Oracle has syntax for auditing specific user activity. To audit the activity of user FRED we could issue these audit commands:

Audit all Oracle user activity.

This audits everything including DDL (create table), DML (inserts, updates, deletes) and login/logoff events:

```
audit all by FRED by access;
```

Audit all Oracle user viewing activity:

```
audit select table by FRED by access;
```

Audit all Oracle user data change activity:

```
audit update table, delete table,
insert table by FRED by access;
```

Audit all Oracle user viewing activity:

```
audit execute procedure by FRED by access;
```

We can also query the dba_audit_trail view. Here are the column descriptions from the Oracle documentation:

Column	Datatype	NULL	Description
OS_USERNAME	VARCHAR2(255)		Operating system login username of the user whose actions were audited
USERNAME	VARCHAR2(30)		Name (not ID number) of the user whose actions were audited
USERHOST	VARCHAR2(128)		Client host machine name
TERMINAL	VARCHAR2(255)		Identifier of the user's terminal
TIMESTAMP	DATE		Date and time of the creation of the audit trail entry (date and time of user login for entries created by AUDIT SESSION) in the local database session time zone
OWNER	VARCHAR2(30)		Creator of the object affected by the action
OBJ_NAME	VARCHAR2(128)		Name of the object affected by the action
ACTIONNUMBER		NOT NULL	Numeric action type code. The corresponding name of the action type is in the ACTION_NAME column.
ACTION_NAME	VARCHAR2(28)		Name of the action type corresponding to the numeric code in the ACTION column
NEW_OWNER	VARCHAR2(30)		Owner of the object named in the NEW_NAME column
NEW_NAME	VARCHAR2(128)		New name of the object after a RENAME or the name of the underlying object
OBJ_PRIVILEGE	VARCHAR2(16)		Object privileges granted or revoked by a GRANT or REVOKE statement

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

SYS_PRIVILEGE	VARCHAR2(40)	System privileges granted or revoked by a GRANT or REVOKE statement
ADMIN_OPTION	VARCHAR2(1)	Indicates whether the role or system privilege was granted with the ADMIN option
GRANTEE	VARCHAR2(30)	Name of the grantee specified in a GRANT or REVOKE statement
AUDIT_OPTION	VARCHAR2(40)	Auditing option set with the AUDIT statement
SES_ACTIONS	VARCHAR2(19)	Session summary (a string of 16 characters, one for each action type in the order ALTER, AUDIT, COMMENT, DELETE, GRANT, INDEX, INSERT, LOCK, RENAME, SELECT, UPDATE, REFERENCES, and EXECUTE. Positions 14, 15, and 16 are reserved for future use. The characters are:
?	--	None
?	S	Success
?	F	Failure
?	B	Both
LOGOFF_TIME	DATE	Date and time of user log off
LOGOFF_LREAD	NUMBER	Logical reads for the session
LOGOFF_PREAD	NUMBER	Physical reads for the session
LOGOFF_LWRITE	NUMBER	Logical writes for the session
LOGOFF_DLOCK	VARCHAR2(40)	Deadlocks detected during the session
COMMENT_TEXT	VARCHAR2(4000)	Text comment on the audit trail entry, providing more information about the statement audited

Also indicates how the user was authenticated. The method can be one of the following:

?	DATABASE	- Authentication was done by password
?	NETWORK	- Authentication was done by Oracle Net Services or the Advanced Security option
?	PROXY	- Client was authenticated by another user; the name of the proxy user follows the method type

SESSIONID	NUMBER	NOT NULL	Numeric ID for each Oracle session
ENTRYID	NUMBER	NOT NULL	Numeric ID for each audit trail entry in the session
STATEMENTID	NUMBER	NOT NULL	Numeric ID for each statement run
RETURNCODE	NUMBER	NOT NULL	Oracle error code generated by the action.

Some useful values:

?	0	- Action succeeded
?	2004	- Security violation

PRIV_USED	VARCHAR2(40)	System privilege used to execute the action
CLIENT_ID	VARCHAR2(64)	Client identifier in each Oracle session
SESSION_CPU	NUMBER	Amount of CPU time used by each Oracle session
EXTENDED_TIMESTAMP	TIMESTAMP(6) WITH TIME ZONE	Timestamp of the creation of the audit trail entry (timestamp of user login for entries created by AUDIT SESSION) in UTC (Coordinated Universal Time) time zone
PROXY_SESSIONID	NUMBER	Proxy session serial number, if an enterprise user has logged in through the proxy mechanism
GLOBAL_UID	VARCHAR2(32)	Global user identifier for the user, if the user has logged in as an enterprise user
INSTANCE_NUMBER	NUMBER	Instance number as specified by the INSTANCE_NUMBER initialization parameter
OS_PROCESS	VARCHAR2(16)	Operating System process identifier of the Oracle process

TRANSACTIONID	RAW(8)	Transaction identifier of the transaction in which the object is accessed or modified
SCN	NUMBER	System change number (SCN) of the query
SQL_BIND	NVARCHAR2(2000)	Bind variable data of the query
SQL_TEXT	NVARCHAR2(2000)	SQL text of the query

--Verificação pode ser feita pela query abaixo:

```
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;
```

#####138- Verificações de Response-time

-- Verificando como o banco está rodando inicialmente:

```
SELECT metric_name, VALUE
FROM SYS.v_$sysmetric
WHERE metric_name IN('Database CPU Time Ratio', 'Database Wait Time Ratio')
AND intsize_csec = (SELECT MAX(intsize_csec)
FROM SYS.v_$sysmetric);
```

O Oracle Database 10g visão V \$ SYSMETRIC contém várias métricas excelente tempo de resposta, das quais duas são a base de dados Tempo de espera da relação(Database Wait Time Ratio) e Banco de DadosTempo de CPU Ratio(Database CPU Time Ratio).

A consulta acima mostra a última snapshot das duas estatísticas, que ajudarão a determinar se seu banco de dados está atualmente a atravessar uma elevada percentagem de espera /gargalos contra operações funcionam adequadamente.

A base de dados Tempo de CPU Ratio é calculado dividindo a quantidade de CPU gasto na base de dados pela quantidade de "tempo de banco de dados", que é definido como o tempogasto pelo banco de dados em nível de chamadas de usuário (com processos de background da instância sendo excluídos) .

Os valores elevados (90-95 por cento +) são bons e indicam pouca espera ações / gargalo, mas não tomar esse limite apenas como uma regra geral, porque cada sistema é diferente.

--Você também pode dar uma olhada rápida sobre a última hora para ver se o banco sofreu qualquer baixa no desempenho global, usando esta consulta:

```
SELECT end_time, VALUE
FROM SYS.v_$sysmetric_history
WHERE metric_name = 'Database CPU Time Ratio'
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

ORDER BY 1;

--E você pode ter uma boa idéia do mínimo, máximo, e valores médios de eficiência global do banco de dados, consultando a visão V \$ SYSMETRIC_SUMMARY com uma consulta como esta:

```
SELECT CASE metric_name
      WHEN 'SQL Service Response Time' THEN 'SQL Service Response Time (secs)'
      WHEN 'Response Time Per Txn' THEN 'Response Time Per Txn (secs)'
      ELSE metric_name
    END metric_name,
CASE metric_name
      WHEN 'SQL Service Response Time' THEN ROUND( (minval / 100), 2)
      WHEN 'Response Time Per Txn' THEN ROUND( (minval / 100), 2)
      ELSE minval
    END minimum,
CASE metric_name
      WHEN 'SQL Service Response Time' THEN ROUND( (maxval / 100), 2)
      WHEN 'Response Time Per Txn' THEN ROUND( (maxval / 100), 2)
      ELSE maxval
    END maximum,
CASE metric_name
      WHEN 'SQL Service Response Time' THEN ROUND( (average / 100), 2)
      WHEN 'Response Time Per Txn' THEN ROUND( (average / 100), 2)
      ELSE average
    END average
  FROM SYS.v_$sysmetric_summary
 WHERE metric_name IN
      ('CPU Usage Per Sec', 'CPU Usage Per Txn', 'Database CPU Time Ratio', 'Database Wait
Time Ratio',
      'Executions Per Sec', 'Executions Per Txn', 'Response Time Per Txn', 'SQL Service
Response Time',
      'User Transaction Per Sec')
 ORDER BY 1
```

Onde Response Time Per Txn (secs) corresponde ao tempo máximo de resposta por transação de usuário, sendo assim, este valor sendo baixo, em caso de lentidão, não tem o que se falar em problemas de banco de dados.

Com a query abaixo é possível verificar quais atividades dos usuários estão onerando o banco de dados:

```
SELECT CASE db_stat_name
      WHEN 'parse time elapsed' THEN 'soft parse time'
      ELSE db_stat_name
    END db_stat_name,
CASE db_stat_name
      WHEN 'sql execute elapsed time' THEN time_secs - plsql_time
      WHEN 'parse time elapsed' THEN time_secs - hard_parse_time
      ELSE time_secs
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```

END time_secs,
CASE db_stat_name
  WHEN 'sql execute elapsed time' THEN ROUND(100 *(time_secs - plsql_time) / db_time,
2)
  WHEN 'parse time elapsed' THEN ROUND(100 *(time_secs - hard_parse_time) /
db_time, 2)
  ELSE ROUND(100 * time_secs / db_time, 2)
END pct_time
FROM (SELECT stat_name db_stat_name, ROUND( (VALUE / 1000000), 3) time_secs
      FROM SYS.v_$sys_time_model
      WHERE stat_name NOT IN('DB time', 'background elapsed time', 'background cpu time',
'DB CPU') ),
      (SELECT ROUND( (VALUE / 1000000), 3) db_time
      FROM SYS.v_$sys_time_model
      WHERE stat_name = 'DB time'),
      (SELECT ROUND( (VALUE / 1000000), 3) plsql_time
      FROM SYS.v_$sys_time_model
      WHERE stat_name = 'PL/SQL execution elapsed time'),
      (SELECT ROUND( (VALUE / 1000000), 3) hard_parse_time
      FROM SYS.v_$sys_time_model
      WHERE stat_name = 'hard parse elapsed time')
ORDER BY 2 DESC;

```

Além do tempo de atividade, um DBA vai querer saber o tempo de espera global. Agora a Oracle fornece waits /wait class:

```

SELECT wait_class, total_waits, ROUND(100 *(total_waits / sum_waits), 2) pct_waits,
      ROUND( (time_waited / 100), 2) time_waited_secs, ROUND(100 *(time_waited /
sum_time), 2) pct_time
FROM (SELECT wait_class, total_waits, time_waited
      FROM v$system_wait_class
      WHERE wait_class != 'Idle'),
      (SELECT SUM(total_waits) sum_waits, SUM(time_waited) sum_time
      FROM v$system_wait_class
      WHERE wait_class != 'Idle')
ORDER BY 5 DESC;

```

Tal como acontece com as métricas de tempo de resposta, você também pode olhar para trás no tempo durante a última hora com uma consulta como esta:

```

SELECT TO_CHAR(a.end_time, 'DD-MON-YYYY HH24:MI:SS') end_time, b.wait_class,
      ROUND( (a.time_waited / 100), 2) time_waited
FROM SYS.v_$waitclassmetric_history a, SYS.v_$system_wait_class b
WHERE a.wait_class# = b.wait_class# AND b.wait_class != 'Idle'
ORDER BY 1, 2;

```

Você pode, naturalmente, apenas se concentrar em um único SID com a visão V \$ SESS_TIME_MODEL e obter dados para todas as áreas estatísticas de uma sessão. Você

também pode ver a atividade esperar sessão atual usando as classes de esperar novas usando a seguinte consulta:

```
SELECT a.SID, b.username, a.wait_class, a.total_waits, ROUND( (a.time_waited / 100), 2)
time_waited_secs
FROM SYS.v_$session_wait_class a, SYS.v_$session b
WHERE b.SID = a.SID AND b.username IS NOT NULL AND a.wait_class != 'Idle'
ORDER BY 5 DESC;
```

Se você precisa de olhar para trás no tempo para descobrir o que as sessões foram registradas em e consumindo mais recursos, você pode usar a seguinte consulta. No exemplo abaixo, nós estamos olhando para a atividade a partir das três da tarde do dia 15 de fevereiro de 2011, que envolveu usuário I/O espera:

```
SELECT sess_id, username, program, wait_event, sess_time, ROUND(100 *(sess_time /
total_time), 2) pct_time_waited
FROM (SELECT a.session_id sess_id, DECODE(session_type, 'background', session_type,
c.username) username,
a.program program, b.NAME wait_event, SUM(a.time_waited) sess_time
FROM SYS.v_$active_session_history a, SYS.v_$event_name b, SYS.dba_users c
WHERE a.event# = b.event# AND a.user_id = c.user_id AND sample_time > '15-FEB-11
03:00:00 PM'
AND sample_time < '15-FEB-11 03:30:00 PM' AND b.wait_class = 'User I/O'
GROUP BY a.session_id, DECODE(session_type, 'background', session_type, c.username),
a.program, b.NAME),
(SELECT SUM(a.time_waited) total_time
FROM SYS.v_$active_session_history a, SYS.v_$event_name b
WHERE a.event# = b.event# AND sample_time > '15-FEB-11 03:00:00 PM' AND
sample_time < '15-FEB-11 03:30:00 PM'
AND b.wait_class = 'User I/O')
group by sess_id, username, program, wait_event, sess_time, total_time
ORDER BY 6 DESC
```

Uma consulta que você pode utilizar para encontrar os cinco principais comandos SQL com o maiores I/O de usuários em espera seria:

```
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;
```

Por exemplo, digamos que você encontrou uma instrução SQL específica que parece ser extremamente deficiente em termos de usuários I / O tempo de espera. Você pode emitir a seguinte consulta para obter a eventos espera individuais associado com a consulta, juntamente com os tempos de espera correspondente, arquivos e objetos que eram a fonte dessesespera:

```
SELECT event, time_waited, owner, object_name, current_file#, current_block#
FROM SYS.v_$active_session_history a, SYS.dba_objects b
```



```
WHERE sql_id = 'ID_sql' AND a.current_obj# = b.object_id AND time_waited <> 0;
```

#####139- Aplicando patch com o Opatch Unix/Linux

1) Abra seu navegador de preferência e acesse

<http://support.oracle.com> ou metalink.oracle.com (vai redirecionar)

2) Entre com sua conta no metalink e forneça o número do patch necessário, assim como a plataforma exata, pois o Oracle roda em várias plataformas de S.O, escolha a sua.

[https://support.oracle.com/CSP/ui/flash.html#tab=PatchHomePage\(page=PatchHomePage&id=gcwbmiiv\(\)\)](https://support.oracle.com/CSP/ui/flash.html#tab=PatchHomePage(page=PatchHomePage&id=gcwbmiiv()))

3) Faça download do mesmo e transporte para o servidor do modo mais adequado (ftp, sftp, scp, winscp e outros)

4) Descompacte o patch (exemplo usando o gunzip)

```
$ cd /home/oracle
```

```
$ gunzip XXXXXX.zip
```

5) Baixe todas as instâncias que estão usando aquela Oracle home que vai aplicar patch (a maioria dos patches solicita isso);

Esse “for” em shell script faz isso para nós:

```
$ for i in `ps -ef | grep smon | grep -v grep | awk -F"_" '{print $3}'`; do
ORACLE_SID=$i
sqlplus "/as sysdba" <<eof
shutdown immediate;
EOF
done;</eof
```

6) Baixe o listener

```
$ lsnrctl stop
```

7) Verifique se realmente não existe nada rodando

```
$ ps -ef | grep oracle | grep -v grep | grep -v ps | grep -v sshd | grep -v sh
```

8) Faça backup dos binários com o seu comando predileto (eu vou usar o tar) com esse shell abaixo:

```
$ DATA=`date +%F`
```

```
$ HOST=`hostname`
```

```
$ tar -czvf bkp_OH_RDBMS_${HOST}_${DATA}.tar.gz /oracle/product/10.2.0/db_1
```

Ps.: Verifique o caminho de sua Oracle Home antes de rodar o comando acima para ter certeza de um backup dos binários. O comando acima vai criar o backup no diretório corrente, então tenha certeza de estar em um fs com tamanho suficiente para abrigar esse backup.

9) Entre na pasta do patch
\$ cd /home/oracle/XXXXXX

\$ opatch apply

\$ opatch apply -local (caso seja RAC e queira fazer um nó de cada vez para manter disponibilidade).

10) Podemos subir as instâncias rodando agora

```
$ export ORACLE_SID=XXXXXX
$ sqlplus "/as sysdba" <<eof
startup;
EOF</eof
```

11) Para verificar se o seu patch foi aplicado com sucesso consulte o inventory do Oracle com o próprio comando opatch:

\$ opatch lsinventory

Podemos agora ver o número do patch que acabamos de aplicar na saída do comando acima.

#####140- Relink – Oracle Home

Todas as vezes que os administradores de S.O fazem mudanças de versão de kernel, aplicações de patches no sistema Operacional, nós DBAs temos que fazer relink dos executáveis do Oracle. Abaixo segue um roteiro do que precisa ser feito para tal tarefa no Linux e Unix.

1) Descubra o path de sua(s) Oracle Homes
[oracle@srv1] \$ echo \$ORACLE_HOME

2) Faça backup de sua Oracle Home
[oracle@srv1] \$ tar -czf backup_OH_18072010.tar \$ORACLE_HOME

3) Verifique bases que estão rodando
[oracle@srv1] \$ ps -ef | grep smon | grep -v grep | awk -F"_" '{print \$3}'

4) Baixe uma a uma

```
[oracle@srv1] $ export ORACLE_SID=ORCL
[oracle@srv1] $ sqlplus "/as sysdba"
SQL> shutdown immediate;
```

5) Verifique a saída do comando umask (tem que ser 0022)

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
[oracle@srv1] $ umask
```

6) Agora segue o relink da Oracle Home

```
[oracle@srv1] $ script /home/oracle/relink_srv1_18072010.log #Gerar log
[oracle@srv1] $ cd $ORACLE_HOME
[oracle@srv1] $ $ORACLE_HOME/bin/relink all;
[oracle@srv1] $ exit
```

7) Procurando erros no Log do relink

```
[oracle@srv1] $ grep -i "Code 1?" /home/oracle/relink_srv1_18072010.log
[oracle@srv1] $ grep -i "Fatal" /home/oracle/relink_srv1_18072010.log
[oracle@srv1] $ grep -i "Error" /home/oracle/relink_srv1_18072010.log
```

Caso os comandos grep acima não retorne nada, o relink está pronto e você já pode subir a(s) base(s) de dados novamente. Caso encontre algum erro sugiro dar uma consultada no Metalink e se não tiver nenhum workaround para solucionar, abra uma SR.

Obs.: No caso de RAC a CRS_HOME não necessita de relink.

#####141- Instalando APEX 3.2 no Oracle 10g R2

Este é um tutorial para instalação do ambiente de desenvolvimento APEX 3.2 no Oracle 10g R2 (Patch 3 – 10.2.0.4).

1. Pré-requisitos

Para instalação do ambiente de desenvolvimento de aplicações rápidas (Application Express 3.2) em um banco de dados Oracle 10g devemos atender aos pré-requisitos seguir:

1.1 Oracle versão 9.2.0.3 ou Superior.

1.2 Oracle Text

Para instalação do Oracle Text precisamos executar o script abaixo logado como SYS no banco onde queremos instalar a opção Oracle Text, e devemos usar os parâmetros (schema|tbs_default|tbs_temporaria|account_status) no script:

```
SQL> @$ORACLE_HOME/ctx/admin/catctx.sql ctxsys DRSYS TEMP NOLOCK
```

1.3 Oracle XMLDB

Para instalação do Oracle XMLDB precisamos executar o script abaixo logado como SYS no banco onde queremos instalar a opção Oracle XMLDB, e devemos usar os parâmetros (senha|tbs_default|tbs_temporaria) no script:

```
SQL> @$ORACLE_HOME/rdbms/admin/catqm.sql "SUA_SENHA" XDB TEMP
```

1.4 OHS (Oracle http Server) e HTMLDB

Para realizar a instalação dessas opções precisaremos realizar o download do companion cd da versão 10g R2 diretamente do site da Oracle no link

http://download.oracle.com/otn/linux/oracle10g/10201/10201_companion_linux_x86_64.cpio.gz.

Crie uma nova Home para o OHS antes de chamar o OUI e designe a instalação para utilizar essa nova home. Siga os passos do OUI escolhendo a opção (HTMLDB + OHS).

Quando chegar ao final do processo de instalação no RedHat AS 4 falta uma lib que foi renomeada, visto que o pacote compat-db-4.1.25-9, que é pré-requisito de instalação do banco, nas versões 32 e 64 bits já haviam sido instalados antes de iniciar esse processo.

Seguindo a Nota do Metalink 784162.1

```
# ln -s /usr/lib/libgdbm.so.2.0.0 /usr/lib/libdb.so.2
```

Obs.: Se executar esse passo antes da instalação será melhor, pois a instalação não irá falhar (Aqui está o pulo do Gato).

2. Procedimentos

2.1 Baixe o Apex 3.2

http://download.oracle.com/otn/java/appexpress/apex_3.2.zip

```
# unzip apex_3.2.zip
```

```
# cd apex
```

2.2 Instalação do ambiente de desenvolvimento APEX 3.2

Conecte-se na base de dados onde será instalado o APEX como SYS e execute o script abaixo com os parâmetros (tbs_def|tbs_deft_files|tbs_temp|Virtual_directory_http_server) no script:

```
SQL>@apexins SYSAUX SYSAUX TEMP /img/
```

2.3 Alteração de senha do ADMIN e apex_public_user

O script abaixo é utilizado para trocar a senha do ADMIN que é o administrador de todo o ambiente do Apex, ele é quem pode criar as Workspaces e gerir todo esse ambiente.

```
SQL>@apxchpwd
```

```
SQL>alter user apex_public_user account unlock;
```

```
SQL>alter user apex_public_user identified by NOVA_SENHA;
```

2.4 Configurando os arquivos do APEX

Copie as imagens do diretório do apex para o diretório virtual do OHS

```
# cp -r images /u01/app/oracle/product/10.2.0/comp/htmldb/
```

2.5 Configurando o OHS para rodar o APEX 3.2

Abra o arquivo de configuração dads.conf

```
# vi /u01/app/oracle/product/10.2.0/comp/Apache/modplsql/conf/dads.conf
```

Adicione as linhas abaixo no arquivo:

```
Alias /img/ /u01/app/oracle/product/10.2.0/comp/htmldb/images/
```

```
Order deny,allow
```

```
PlsqlDocumentPath docs
```

```
AllowOverride None
```

```
PlsqlDocumentProcedure www_flow_file_manager.process_download
```

```
PlsqlDatabaseConnectionString seuhost:1521:service_name.suaempresa.com
```

```
ServiceNameFormat
```

```
PlsqlNLSLanguage AMERICAN_AMERICA.WE8ISO8859P1
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
PlsqlAuthenticationMode Basic
SetHandler pls_handler
PlsqlDocumentTablename www_flow_file_objects$
PlsqlDatabaseUsername APEX_PUBLIC_USER
PlsqlDefaultPage htmldb
PlsqlDatabasePassword apex
Allow from all
```

O que precisamos alterar de ambiente para ambiente são as configurações:

Linha 1 “Alias” → Mude para o seu alias de diretório virtual que você apontou no apexins, passo 5.2, e o caminho no seu servidor onde você copiou os arquivos referente ao passo 5.4.

Linha 8 “PlsqlDatabaseConnectionString” → Substitua afim de apontar a connect string do seu banco de dados onde está instalado o apex IP_SERVIDOR:PORTA_LISTENER:SERVICE_NAME;

Linha 15 “PlsqlDatabasePassword” → Substitua pela senha do ADMIN informada no passo 5.3;

Altere o arquivo marvel.conf para que o mesmo reflita o diretório virtual informador no passo 5.2 conforme demonstrado abaixo;

```
# vi /u01/app/oracle/product/10.2.0/comp/Apache/modplsql/conf/marvel.conf
Alias /img/ /u01/app/oracle/product/10.2.0/comp/htmldb/images/
```

2.6 Reiniciando o OHS (opmnctl)

\$(Parar)

```
#!/u01/app/oracle/product/10.2.0/comp/opmn/bin/opmnctl stopall
```

\$(Iniciar)

```
#!/u01/app/oracle/product/10.2.0/comp/opmn/bin/opmnctl startall
```

3. Verificações após a Instalação do Apex 3.2

3.1 Acesso à página de administração está na porta que

Verifique o arquivo portlist.ini dentro da home onde foi instalada o OHS para saber que porta o Oracle HTTP Server está usando. No nosso caso a porta foi 7700, então a página de administração é acessada através da URL:

http://seuhost.suaempresa.com:7700/pls/apex/apex_admin

3.2 Acesso à página de desenvolvimento de aplicações

Podemos acessar as workspaces de desenvolvimento através da URL

http://seuhost.suaempresa.com:7700/pls/apex

3.3 E-mail de notificação de requisição de workspace

Todas as vezes que uma Workspace é criada, automaticamente o Apex manda um e-mail para a conta do Analista que solicitou, conforme o exemplo abaixo:

Application Express – New Workspace Request

Workspace Name: RANDRADE

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

Schema Name: RANDRADE_APEX
Userid: RANDRADE
Email: Raul.andrade@clickdba.com
URL: http://seuhost:7700/pls/apex/

#####142- Sessões com DB links - resolvendo problemas de excesso de sessões
--Caso apareça algum problema e necessite fechar a conexão do dblink após execução. Veja exemplo:

Comandos Executados:

```
select sysdate from dual@HR_LK1 <<< abre dblink PBCT1  
select sysdate from dual@HR_LK2 <<< abre dblink PBCT2
```

Verificação do dblink para a sessão:

```
select * from V$DBLINK <<< neste ponto tem dois DBLINKs na tua sessão >>>
```

commit <<< é obrigatório e antes do alter session >>>

Derrubando a conexão do dblink da sessão:

```
ALTER SESSION CLOSE DATABASE LINK teste_pbct1 <<< derruba o dblink PBCT1. Não é  
necessário colocar dono do dblink.
```

Verificação do dblink para a sessão:

```
select * from V$DBLINK <<< neste ponto tem somente o dblink PBCT2
```

#####143- Aplicando archives em dataguard quando ocorre perda de sequência

Pode ocorrer em um sistema configurado Dataguard, que a sequencia de archives fique desigual entre as base de contingência e a produção.

1- Deve-se primeiramente verificar no alert log contingência da instância "master", pois é uma delas que aplicará os archives.

Exemplo:

RFS[2]: Archived Log:

```
'+CCME_RECV01/ccme/archivelog/2011_03_09/thread_1_seq_17715.1079.745290407'
```

Wed Mar 9 01:07:13 2011

Fetching gap sequence in thread 1, gap sequence 17709-17714

Wed Mar 9 01:07:48 2011

FAL[client]: Failed to request gap sequence

GAP - thread 1 sequence 17709-17714

DBID 1609635303 branch 681410474

FAL[client]: All defined FAL servers have been attempted.

Com isto percebe-se que faltaram a aplicar as sequencias de 17709-17714 da thread 1.

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

2- Pode-se também verificar na view:

```
select * from v$archive_gap -- rodar na contingência
```

ou ainda rodando a query abaixo tanto na contingência quanto na produção:

```
select THREAD#, sequence#, to_char(FIRST_TIME,'day') day, to_char(first_time,'dd/mm/yyyy  
hh24:mi:ss') time  
from v$log_history  
where first_time > to_date('01/01/2011 15:00:00','dd/mm/yyyy hh24:mi:ss')  
order by 1,2;
```

A query acima mostra os archives aplicados, caso haja diferença haverá archives faltando.

3- Identificados para aplicar faça:

```
connect catalog user_catalog/senha_catalog@catalog --caso utilize catálogo  
connect target /
```

```
run{  
  allocate channel ch1 type 'sbt_tape'; -- caso utilize fita  
  send 'NSR_ENV=(NSR_SERVER=svuxnsr1.net.br,NSR_CLIENT=racvipcmd_1_2.net.br)';--  
  parâmetro de configuração de fita.  
  restore archivelog from sequence 13441 until sequence 13441 thread 2;  
  release channel ch1;  
}
```

#####144- Executar comando shell de um servidor para outro

```
ssh -n svuxhost1 "kill -9 pid"
```

#####145- Criando tabelas externas

--145.1 Criando tabela externa com ORACLE_LOADER

-- Criando o arquivo dat (SO LINUX/AIX/HPUX) (caso 1 - arquivo com campos alinhados):

-- vi emp.dat

"emp.dat" 4 lines, 118 characters

```
RAUL          ,ANDRADE  
JOAO          ,ANDRADE  
ELIS          ,NOGUEIRA
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

AMARILDO ,SANCHES

-- Criando o arquivo dat (SO LINUX/AIX/HPUX) (caso 2 - arquivo com campos desalinhados):

-- vi emp.dat

"emp.dat" 4 lines, 118 characters

RAUL,ANDRADE

JOAO,ANDRADE

ELIS,NOGUEIRA

AMARILDO,SANCHES

--Cria tabela (caso 1 - arquivo com campos alinhados):

```
CREATE TABLE oldemp (fname char(25), lname char(25))
ORGANIZATION EXTERNAL
(TYPE ORACLE_LOADER
DEFAULT DIRECTORY data_pump_dir
ACCESS PARAMETERS
(RECORDS DELIMITED BY NEWLINE
NOBADFILE
NOLOGFILE
FIELDS TERMINATED BY ','
(fname POSITION(1:20) CHAR,
lname POSITION(22:41) CHAR))
LOCATION('emp.dat'))
PARALLEL 5
REJECT LIMIT 200
```

--Cria tabela (caso 2 - arquivo com campos desalinhados):

```
CREATE TABLE oldemp (fname char(25), lname char(25))
ORGANIZATION EXTERNAL
(TYPE ORACLE_LOADER
DEFAULT DIRECTORY data_pump_dir
ACCESS PARAMETERS
(RECORDS DELIMITED BY NEWLINE
NOBADFILE
NOLOGFILE
FIELDS TERMINATED BY ','
(fname CHAR,
lname CHAR))
LOCATION('emp.dat'))
PARALLEL 5
REJECT LIMIT 200
```

--Verificando valores inseridos:

```
select * from oldemp
```


-----X-----X-----X-----

--145.2 Criando tabela externa com driver ORACLE_DATAPUMP

-- Cria tabela interna para o test case:

```
create table employees (employee_id number(10),
first_name varchar2(50),
last_name varchar2(50))
```

-- insere dados:

```
insert into employees values('01','RAUL','ANDRADE');
insert into employees values('02','ELIS','NOGUEIRA');
insert into employees values('03','JOAO','ANDRADE');
insert into employees values('01','LOUISE','ANDRADE');
insert into employees values('01','SABRINA','ANDRADE');
Commit;
```

--Verifica diretorios

```
select * from dba_directories
DATA_PUMP_DIR
```

-- cria tabela externa:

```
CREATE TABLE emp_ext
(employee_id, first_name, last_name)
ORGANIZATION EXTERNAL
(TYPE ORACLE_DATAPUMP
DEFAULT DIRECTORY data_pump_dir
LOCATION
('emp1.exp','emp2.exp'))
PARALLEL
as select * from employees;
```

-- Verifica valores incluídos:

```
select * from emp_ext
```

-- Observação: caso seja deletado valor na tabela principal ele não será alterado na externa.

--Diferença básica entre elas:

Na que utiliza o ORACLE_DATAPUMP, depende de uma tabela interna do sistema porém grava os dados de leitura em um arquivo externo.

Na que utiliza o ORACLE_LOADER, você pode acrescentar/diminuir dados, alterando o arquivo .dat(não obrigatório esta extensão).

#####146- Rodando o hanganalyze

Caso tenha problemas e precise rodar um trace de memória (hanganalyze):

```
sqlplus -prelim / as sysdba
oradebug setmypid
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
oradebug unlimit
oradebug hanganalyze 4
wait 90 seconds
oradebug hanganalyze 4
exit
```

```
sqlplus -prelim / as sysdba
oradebug setmypid
oradebug unlimit
set time on
oradebug dump systemstate 266
wait 90 seconds
oradebug dump systemstate 266
wait 90 seconds
oradebug dump systemstate 266
exit
```

#####147- Habilitando o Resource Manager para evitar o paralelismo.

```
BEGIN
  DBMS_RESOURCE_MANAGER.CLEAR_PENDING_AREA();
  DBMS_RESOURCE_MANAGER.CREATE_PENDING_AREA();

  DBMS_RESOURCE_MANAGER.CREATE_PLAN(PLAN =>
'LIMIT_PARALLELISM_PLAN',COMMENT =>'Resource plan for parallel execution CLICKDBA');

  DBMS_RESOURCE_MANAGER.CREATE_CONSUMER_GROUP(CONSUMER_GROUP
=>'PARALLEL_DEGREE', COMMENT => 'Limit the degree of parallelism CLICKDBA');

  DBMS_RESOURCE_MANAGER.CREATE_PLAN_DIRECTIVE(PLAN =>
'LIMIT_PARALLELISM_PLAN',GROUP_OR_SUBPLAN => 'PARALLEL_DEGREE', COMMENT =>
'Limit the degree of parallelism CLICKDBA',PARALLEL_DEGREE_LIMIT_P1 => 3);
  DBMS_RESOURCE_MANAGER.CREATE_PLAN_DIRECTIVE(PLAN =>
'LIMIT_PARALLELISM_PLAN',GROUP_OR_SUBPLAN => 'OTHER_GROUPS', COMMENT => 'Leave
others alone');

  DBMS_RESOURCE_MANAGER.SET_CONSUMER_GROUP_MAPPING(ATTRIBUTE =>
DBMS_RESOURCE_MANAGER.ORACLE_USER,VALUE => 'AGERONDO',CONSUMER_GROUP =>
'PARALLEL_DEGREE');
  DBMS_RESOURCE_MANAGER.VALIDATE_PENDING_AREA();
  DBMS_RESOURCE_MANAGER.SUBMIT_PENDING_AREA();

  DBMS_RESOURCE_MANAGER_PRIVS.GRANT_SWITCH_CONSUMER_GROUP(GRANTEE_NAME
=> 'AGERONDO', CONSUMER_GROUP => 'PARALLEL_DEGREE', GRANT_OPTION => FALSE);

  DBMS_RESOURCE_MANAGER.SET_INITIAL_CONSUMER_GROUP(USER =>
'AGERONDO',CONSUMER_GROUP => 'PARALLEL_DEGREE');

END;
```

```
alter system set resource_manager_plan = 'LIMIT_PARALLELISM_PLAN' scope=both sid='*';
```

```
user
```

```
BEGIN
SYS.DBMS_RESOURCE_MANAGER.clear_pending_area();
SYS.DBMS_RESOURCE_MANAGER.create_pending_area();
SYS.DBMS_RESOURCE_MANAGER_PRIVS.GRANT_SWITCH_CONSUMER_GROUP
('TESTE','PARALLEL_DEGREE',FALSE);
SYS.DBMS_RESOURCE_MANAGER.submit_pending_area();
END;
/
```

```
BEGIN
SYS.DBMS_RESOURCE_MANAGER.SET_INITIAL_CONSUMER_GROUP
('TESTE','PARALLEL_DEGREE');
END;
/
```

#####148 - Verificando a utilização da PGA estatisticamente

```
SELECT NAME, VALUE FROM v$pgastat;
```

#####149 – Verificando como setar o OPTIMIZER_INDEX_COST_ADJ

```
SELECT SUM(a.time_waited_micro) / SUM(a.total_waits) / 1000000 c1,
       SUM(b.time_waited_micro) / SUM(b.total_waits) / 1000000 c2,
       (SUM(a.total_waits) / SUM(a.total_waits + b.total_waits) ) * 100 c3,
       (SUM(b.total_waits) / SUM(a.total_waits + b.total_waits) ) * 100 c4,
       (SUM(b.time_waited_micro) / SUM(b.total_waits) ) / (SUM(a.time_waited_micro) /
SUM(a.total_waits) ) * 100 c5
FROM dba_hist_system_event a, dba_hist_system_event b
WHERE a.snap_id = b.snap_id AND a.event_name = 'db file scattered read' AND
b.event_name = 'db file sequential read';
```

Observação: colocar o valor do C5 como valor para o parametro de inicialização

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

optimizer_index_cost_adj

#####150- Alterações em REDOLOG

--Verifica endereço dos logs

select * from v\$logfile;

-- Adiciona novos logfiles

ALTER DATABASE

ADD LOGFILE GROUP 6 ('C:\ORACLE\ORADATA\CLICKMAR\REDO08.LOG',

'C:\ORACLE\ORADATA\CLICKMAR\REDO09.LOG')

SIZE 500M;

--Verifica qual está ativo e drop

select * from v\$log;

--Dropa os inativos

alter database drop logfile 'C:\ORACLE\ORADATA\CLICKMAR\REDO01.LOG'

--Forçar a alternância de log para poder deletar

alter system switch logfile;

alter system checkpoint;

#####151- Verificar a quantidade de commits das sessões;

select sid, value from v\$sesstat

where statistic# = (select statistic#

from v\$statname where name = 'user commits')order by value desc;

#####152- Verificando quais tabelas sofreram modificações e não tem estatísticas

select * from ALL_TAB_MODIFICATION;

#####153- Query para detectar excessos de archivelog – Query Chiappa

Schema aonde criar Tab temporaria default=SYSTEM :

Segundos de Intervalo entre execs default=1 a 60

Condições no result : usar Name, Valor1, Valor2 ou Diferença (opcional) =>name like '%redo%'
and diferença > 0

Lista de SIDs (opcional):

set autoprint off echo off feed off show off trim off verify off pages 9999 lines 300

```

ACCEPT v_usuario Char DEFAULT 'SYSTEM' PROMPT "Schema aonde criar Tab temporaria
default=SYSTEM : "
ACCEPT v_seconds Number DEFAULT '1' PROMPT "Segundos de Intervalo entre execs
default=1 : "
accept v_cond DEFAULT '1=1' prompt "Condições no result : usar Name, Valor1,
Valor2 ou Diferença (opcional) =>"
accept v_sid_list DEFAULT ss.sid PROMPT "Lista de SIDs (opcional):"
create table &&v_usuario..TEMP_STATS_TBL as
select sid, name, value, 1 exec#
  from v$sesstat ss, v$statname sn
 where sn.statistic#=ss.statistic#
 and sid <> (select sid from v$mystat where rownum = 1)
 and ss.sid in (&v_sid_list);
--
exec dbms_lock.sleep(&v_seconds);
--
insert into &&v_usuario..TEMP_STATS_TBL
(select sid, name, value, 2 exec#
  from v$sesstat ss, v$statname sn
 where sn.statistic#=ss.statistic#
 and ss.sid <> (select sid from v$mystat where rownum = 1)
 and ss.sid in (&v_sid_list));
--
column SID format 99999
column NAME format A44
column Valor1 format 99999999999
column Valor2 format 99999999999
PROMPT
PROMPT
PROMPT "Estats Alteradas"
PROMPT "===== "
break on SID skip page
select * from
( select t1.sid, t1.name, t1.value Valor1, t2.value Valor2, t2.value - t1.value Diferença from
&&v_usuario..TEMP_STATS_TBL t1, &&v_usuario..TEMP_STATS_TBL t2
  where t1.sid = t2.sid
    and t1.name = t2.name
    and t1.exec# = 1
    and t2.exec# = 2
    and t1.value <> t2.value
  order by 1, 5 desc)
where &v_cond;
commit;
drop table &&v_usuario..TEMP_STATS_TBL;
undefine v_usuario
undefine v_seconds
undefine v_sid_list
undefine v_cond

```

Variação Query do Beto:

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

1) Para descobrir qual segmento que mais foi alterado e portanto mais gerou "redo information" execute a seguinte query:

```
SELECT dhso.object_name, dhso.object_type,
       sum(db_block_changes_delta)
FROM dba_hist_seg_stat dhss,
     dba_hist_seg_stat_obj dhso,
     dba_hist_snapshot dhs
WHERE dhs.snap_id = dhss.snap_id
AND dhs.instance_number = dhss.instance_number
AND dhss.obj# = dhso.obj#
AND dhss.dataobj# = dhso.dataobj#
AND begin_interval_time BETWEEN to_date('2013_11_01 00','YYYY_MM_DD HH24') AND
to_date('2013_11_13 00','YYYY_MM_DD HH24')
GROUP BY dhso.object_name, dhso.object_type
order by sum(db_block_changes_delta) desc
```

Exemplo:

O resultado foi o seguinte:

```
SYS_IOT_TOP_13274    INDEX 20270580576
AQ$_SYS$SERVICE_METRICS_TAB_L TABLE 16468474336
SYS_IOT_TOP_13282    INDEX 555419840
TEMPHIST            TABLE 80661296
V_WSALUNO           TABLE 49376896
GT_CATALOGA         TABLE 43265472
```

Daí verifica quais queries estão usando ele

#####154- Mostrar SQL master and slaves e pra fazer monitoração de Diferença de waits entre sessões

Foi lá : salva esses caras em arquivos .SQL e roda de dentro do sql*plus, com um usuário privilegiado (ie, que acessa packages do SYS, DBA_xx e V\$xxx)
o primeiro vc pode dar ENTER quando ele perguntar qual SID coordenadora/master vc quer ver, que aí ele mostra todas, o que a intenção
e o segundo vc dá ENTER no nome da tablespace aonde ele cria a tab temporária (se vc não tiver nenhuma restrição a usar temporariamente a SYSTEM),
e depois informa quantos segundos de intervalo entre as monitorações (eu costumo usar 60)
ENTER, e finalmente vc indica Diferença > 0
aí ele vai mostrar todas as SIDs que tiveram processamento no intervalo, E o quanto teve de diferença
fatalmente os eventos de I/O vão mostrar boas diferenças
aí vc relaciona pro cliente as SIDs do segundo script com as do primeiro, a maioria das SIDs slaves deve estar aparecendo no segundo script
=> script para consultar sessões master & slaves de parallel SQL :

```
accept sid_list DEFAULT QC_SID prompt "Lista de QC SIDs (opcional):"
accept slave_sids DEFAULT SID prompt "Lista de Slaves (opcional):"
select * from (
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
select
  decode(px.qcinst_id,NULL,username,
    ' - ' || lower(substr(s.program,length(s.program)-4,4) ) ) "Username",
  decode(px.qcinst_id,NULL, 'QC', '(Slave)') "QC/Slave" ,
  to_char( px.server_set) "Slave Set",
  to_char(s.sid) "SID",
  decode(px.qcinst_id, NULL ,to_char(s.sid) ,px.qcsid) QC_SID,
  px.req_degree "Requested DOP",
  px.degree "Actual DOP"
from
  v$px_session px,
  v$session s
where
  px.sid=s.sid (+)
and
  px.serial#=s.serial#
order by 5 , 1 desc
)
where QC_SID in (&SID_LIST)
and (SID in (&slave_sids) OR QC_SID=SID)
/
```

=> script de consultar diferença entre duas tomadas de waits/stats entre sessões :

```
/* Script para comparar estatísticas ENTRE todas as sessões ativas, IGNORANDO
  apenas a sessão que está EXECUTANDO o script.
  O objetivo é permitir facilmente descobrir QUAIS estatísticas
  estão variando na sessão, de modo a ter uma noção do que a sessão está fazendo.
  OBS : deve rodar como usuário privilegiado, e exige a package DBMS_LOCK criada.
*/
set autoprint off echo off feed off show off trim off verify off
ACCEPT v_usuario Char DEFAULT 'SYSTEM' PROMPT "Schema aonde criar Tab temporária
default=SYSTEM : "
ACCEPT v_seconds Number DEFAULT '1' PROMPT "Segundos de Intervalo entre execs
default=1 : "
accept v_cond DEFAULT '1=1' prompt "Condição no result : usar Name, Valor1, Valor2
ou Diferença (opcional) =>"
accept v_sid_list DEFAULT ss.sid PROMPT "Lista de SIDs (opcional):"
create table &&v_usuario..TEMP_STATS_TBL as
select sid, name, value, 1 exec#
  from v$sesstat ss, v$statname sn
 where sn.statistic#=ss.statistic#
    and sid <> (select sid from v$mystat where rownum = 1)
    and ss.sid in (&v_sid_list);
--
exec dbms_lock.sleep(&&v_seconds);
--
insert into &&v_usuario..TEMP_STATS_TBL
```

```
(select sid, name, value, 2 exec#
  from v$sesstat ss, v$statname sn
 where sn.statistic#=ss.statistic#
   and ss.sid <> (select sid from v$mystat where rownum = 1)
   and ss.sid in (&v_sid_list)
);
--
column SID format 999
column NAME format A44
column Valor1 format 99999999999
column Valor2 format 99999999999
PROMPT
PROMPT
PROMPT "Estatísticas Alteradas"
PROMPT "=====
break on SID skip page
select * from
(
  select t1.sid, t1.name, t1.value Valor1, t2.value Valor2, t2.value - t1.value Diferença
    from &v_usuario..TEMP_STATS_TBL t1, &v_usuario..TEMP_STATS_TBL t2
   where t1.sid = t2.sid
     and t1.name = t2.name
     and t1.exec# = 1
     and t2.exec# = 2
     and t1.value <> t2.value
   order by 1, 5 desc
)
where &v_cond
;
commit;
drop table &v_usuario..TEMP_STATS_TBL;
undefine v_usuario
undefine v_seconds
undefine v_sid_list
undefine v_cond
clear breaks
```

#####155- Criando Script para startup automático do Oracle no Linux

1- CRIAR ARQUIVO DE INICIALIZAÇÃO DO ORACLE

Logado como ROOT crie um script de inicialização

vi /usr/local/sbin/initora

Nele coloque:

#####

#!/bin/bash

Data: 05/07/2011

Desc.: Script para inicialização automática do banco Oracle 10g

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle


```
# Setando variáveis de ambiente
ORACLE_HOME=/u01/app/oracle/product/10.2.0/db_1; export ORACLE_HOME
ORACLE_SID="linuxmar";export ORACLE_SID
TNS_ADMIN=$ORACLE_HOME/network/admin/listener.ora; export TNS_ADMIN
```

```
# Startup do Listener
su oracle -c -l "/u01/app/oracle/product/10.2.0/db_1/bin/lsnrctl start"
# Startup do Banco de Dados
su oracle -c -l "/u01/app/oracle/product/10.2.0/db_1/bin/dbstart"
```

```
#####
```

```
2- Troque as permissões do arquivo
# chmod +x /usr/local/sbin/initora
# chown oracle:dba /usr/local/sbin/initora
```

3- NO etc/oratab colocar Y no nome do banco.

```
4- Logado como ROOT
echo /usr/local/sbin/initora >> /etc/rc.local
```

156- Verificação dos privilégios de um determinado usuário inclusive com os herdados

```
set echo off
set verify off
set pages 9999
col granted_role form a25
col owner form a15
col table_name form a33
col privilege form a33
ACCEPT username prompt 'Enter Username : '
PROMPT Roles granted to user
SELECT granted_role,admin_option,default_role
FROM dba_role_privs
WHERE grantee=UPPER('&username')
ORDER BY 1;
PROMPT Table Privileges granted to a user through roles
SELECT granted_role, owner, table_name, privilege
FROM ( SELECT granted_role
FROM dba_role_privs WHERE grantee=UPPER('&username')
UNION
SELECT granted_role
FROM role_role_privs
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
WHERE role in (SELECT granted_role
FROM dba_role_privs WHERE grantee=UPPER('&username')
)
) roles, dba_tab_privs
WHERE granted_role=grantee
ORDER BY 1,2,3,4;
PROMPT System Privileges assigned to a user through roles
SELECT granted_role, privilege
FROM ( SELECT granted_role
FROM dba_role_privs WHERE grantee=UPPER('&username')
UNION
SELECT granted_role
FROM role_role_privs
WHERE role in (SELECT granted_role
FROM dba_role_privs WHERE grantee=UPPER('&username')
)
) roles, dba_sys_privs
WHERE granted_role=grantee
ORDER BY 1,2;
PROMPT Table privileges assigned directly to a user
SELECT owner, table_name, privilege
FROM dba_tab_privs
WHERE grantee=UPPER('&username')
ORDER BY 1,2,3;
PROMPT System privileges assigned directly to a user
SELECT privilege, admin_option
FROM dba_sys_privs
WHERE grantee=UPPER('&username');
undefine username
```

#####157- Trabalhando com DATAPUMP

1- CREATE DIRECTORY dump AS '/oracle/dump/';

2- Após criar o Diretório precisamos conceder os privilégios de leitura e gravação. Em nosso exemplo iremos disponibilizar este diretório para todos os usuários do Banco de Dados concedendo o privilégio para o schema Public;

- GRANT read, write ON DIRECTORY dump TO PUBLIC;

3- Iremos conceder o Privilégio IMP_FULL_DATABASE para o usuário teste que irá Importar:

- GRANT IMP_FULL_DATABASE to teste;

4- Após isso basta exportar e depois importar

5- Exportando com usuario teste com Oracle Data Pump:

- Expdp teste/teste directory=dump dumpfile=teste.dmp logfile=teste.log

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

6- Importando com usuario TESTE no Oracle Data Pump, remapeando por schemas(schema TESTE para o TESTE2) e remapeando por Tablespaces(Tablespace TESTE para o TESTE2):

- Impdp teste/teste directory=dump dumpfile=teste.dmp logfile=testeimp.log
remap_schema=teste:teste2 remap_tablespace=teste:teste2

(Obs: Caso queira remapear mais de uma tablespace, utilizem o seguinte parâmetro setado abaixo:

- REMAP_TABLESPACE = teste0:teste2, teste1:teste2

***** EXP

expdp lf/lfvb@clickmar OWNER=LF VERSION=10.0.0 DIRECTORY=dump
DUMPFILE=bkp300611dp.dmp LOGFILE=bkpdmp.log

***** IMP

Impdp lf/lfvb@linuxmar VERSION=10.0.0 FROMUSER=LF TOUSER=LF directory=dump
dumpfile=bkp300611dp.dmp logfile=imp_300611.log

EXEMPLO USANDO TIMESTAMP usando parfile

Executa no prompt

expdp nfe/nfe@nfe47p parfile=exp_nfe.par

##conteudo do exp_nfe.par

OWNER=nfe

DIRECTORY=ORACLE_DIR

DUMPFILE=bkp_nfe47p_timed.dmp

LOGFILE=bkpnfed_timed.log

CONSISTENT=Y

FLASHBACK_TIME="TO_TIMESTAMP('25-08-2003 14:35:00', 'DD-MM-YYYY HH24:MI:SS')"

BUFFER A parameter comparable to BUFFER is not needed.

COMPRESS A parameter comparable to COMPRESS is not needed.

CONSISTENT A parameter comparable to CONSISTENT is not needed. Use FLASHBACK_SCN and FLASHBACK_TIME for this functionality.

CONSTRAINTS EXCLUDE=CONSTRAINT

DIRECT A parameter comparable to DIRECT is not needed. Data Pump Export automatically chooses the best method (direct path mode or external tables mode).

EXCLUDE=table:"IN('CUSTOMER','ORDOR)'"

FEEDBACK
STATUS

FILE
DUMPFIL

FILESIZE

FILESIZE

FLASHBACK_SCN

FLASHBACK_SCN

FLASHBACK_TIME

FLASHBACK_TIME

FULL

FULL

GRANTS

EXCLUDE=GRANT

HELP

HELP

INDEXES

EXCLUDE=INDEX

LOG

LOGFILE

OBJECT_CONSISTENT

A parameter comparable to OBJECT_CONSISTENT is not needed.

OWNER

SCHEMAS

PARFILE

PARFILE

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

QUERY

QUERY

RECORDLENGTH

A parameter comparable to RECORDLENGTH is not needed because sizing is done automatically.

RESUMABLE

A parameter comparable to RESUMABLE is not needed. This functionality is automatically provided for privileged users.

RESUMABLE_NAME

A parameter comparable to RESUMABLE_NAME is not needed. This functionality is automatically provided for privileged users.

RESUMABLE_TIMEOUT

A parameter comparable to RESUMABLE_TIMEOUT is not needed. This functionality is automatically provided for privileged users.

ROWS=N

CONTENT=METADATA_ONLY

ROWS=Y

CONTENT=ALL

STATISTICS

A parameter comparable to STATISTICS is not needed. Statistics are always saved for tables.

TABLES

TABLES

TABLESPACES

TABLESPACES (Same parameter; slightly different behavior)

TRANSPORT_TABLESPACE

TRANSPORT_TABLESPACES (Same parameter; slightly different behavior)

TRIGGERS

EXCLUDE=TRIGGER

TTS_FULL_CHECK

TRANSPORT_FULL_CHECK

USERID

A parameter comparable to USERID is not needed. This information is supplied as the username/password when you invoke Export.

VOLSIZE

A parameter comparable to VOLSIZE is not needed.

#####158- Verificação das configurações NLS da banco de dados

```
SELECT Parameter, Value FROM NLS_DATABASE_PARAMETERS;  
SELECT Parameter, Value FROM NLS_SESSION_PARAMETERS;  
SELECT Parameter, Value FROM NLS_INSTANCE_PARAMETERS;
```

#####159- Habilitar/desabilitar paralelos para a sessão.

```
ALTER SESSION {ENABLE | DISABLE | FORCE} PARALLEL {DML|DDL|QUERY}
```

#####160- Habilitando e desabilitando o job de coleta automática de estatísticas.
exec dbms_scheduler.disable('SYS.GATHER_STATS_JOB');

#####161- verificar JOB's rodando.

```
select * from DBA_SCHEDULER_JOB_RUN_DETAILS order by ACTUAL_START_DATE desc;
```

```
select * from DBA_SCHEDULER_JOBS where JOB_NAME = 'GATHER_STATS_JOB';
```

#####162- Criando JOB para coleta de estatísticas:

--Setamos parâmetros

```
begin
  DBMS_STATS.SET_PARAM('CASCADE','TRUE');
  DBMS_STATS.SET_PARAM('DEGREE','4');
  DBMS_STATS.SET_PARAM('METHOD_OPT','FOR ALL INDEXED COLUMNS SIZE 1');
  DBMS_STATS.SET_PARAM('ESTIMATE_PERCENT','30');
end;
SQL> /
```

PL/SQL procedure successfully completed.

SQL>

--Então nós criamos um schedule e setamos o JOB para ele:

```
SQL> begin
  dbms_scheduler.create_schedule(
    schedule_name => 'STATS_COLLECTION',
    repeat_interval=>'freq=monthly;bymonthday=5;byhour=12;byminute=30',
    comments => 'Schedule to collect statistics');
  dbms_scheduler.set_attribute(
    name=>'GATHER_STATS_JOB',
    attribute=>'SCHEDULE_NAME',
    value=>'STATS_COLLECTION');
end;
/
```

--Habilitamos o JOB

```
exec dbms_scheduler.enable('SYS.GATHER_STATS_JOB');
```

-- caso precise apagar apagar

```
exec dbms_scheduler.drop_schedule(schedule_name => 'STATS_COLLECTION', force=>true);
```

-- remover estatísticas e rodar de novo.

```
SQL> analyze table scott.emp delete statistics;
```

Table analyzed.

```
SQL> begin
2  dbms_stats.gather_database_stats_job_proc;
3  end;
4  /
```

PL/SQL procedure successfully completed.

#####163- Verificando a geração das estatísticas:

```
select * from DBA_OPTSTAT_OPERATIONS order by start_time desc
```

#####164- Haduken

1- conecte no SQLPLUS e faça:

```
shutdown immediate;  
startup mount restrict;
```

2- No RMAN e faça:

```
drop database including backups noprompt;  
exit;  
--Já era
```

165- Criando procedure para usuário executar kill session

```
create or replace procedure kill_session  
(pn_sid number  
,pn_serial number)  
as  
lv_user varchar2(30);  
begin  
select username into lv_user from v$session where sid = pn_sid and serial# = pn_serial;  
if lv_user is not null and lv_user not in ('SYS','SYSTEM') then  
execute immediate 'alter system kill session ''' || pn_sid || ',' || pn_serial || ''''';  
else  
raise_application_error(-20000,'Attempt to kill protected system session has been blocked.');
```

#####166- Movendo LOB

```
ALTER TABLE table_name MOVE LOB(lob_columnname) STORE AS (TABLESPACE  
new_tablespace_name);
```


#####167- Encontrando HOTBLOCK

```
select
  count(*),
  sql_id,
  nvl(o.object_name,ash.current_obj#) objn,
  substr(o.object_type,0,10) otype,
  CURRENT_FILE# fn,
  CURRENT_BLOCK# blockn
from v$active_session_history ash
  , all_objects o
where event like 'latch: cache buffers chains'
  and o.object_id (+)= ash.CURRENT_OBJ#
  and sql_id='5k948g3mv551g'
group by sql_id, current_obj#, current_file#,
         current_block#, o.object_name,o.object_type
order by count(*)
```

--Usar junto com o número 14

#####168- Debug complete Oracle

```
select sid,p1,p2,event, wait_class, seconds_in_wait, state from v$session_wait where sid=<sid
  fof hung process>;
select p.pid,p.SPID,s.SID
from v$process p,v$session s
where s.paddr = p.addr and s.sid=<sid fof hung process>;
oradebug hanganalyze 5
oradebug setospid <spid from previous query>
oradebug unlimit
oradebug event 10046 trace name context forever, level 12
Note:
Let it run 10 min
oradebug tracefile_name
oradebug event 10046 trace name context off
oradebug dump errorstack 3
```

#####169- Criando partição por data

```
alter table LF_NF_SAIDA
add partition NFS_2011_12 values less than (TO_DATE(' 2012-01-01 00:00:00', 'SYYYY-MM-DD
HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN'))
tablespace TS_DATA_2011_12
pctfree 10
initrans 1
maxtrans 255
storage
(INITIAL 64K
NEXT 1M
MINEXTENTS 1
MAXEXTENTS unlimited
);
```

Table Partitioning

Now a days enterprises run databases of hundred of Gigabytes in size. These databases are known as

Very Large Databases (VLDB). From Oracle Ver. 8.0 Oracle has provided the feature of table partitioning i.e. you can partition a table according to some criteria . For example you have a SALES table with the following structure

Suppose this table contains millions of records, but all the records belong to four years only i.e. 1991, 1992, 1993 and 1994. And most of the time you are concerned about only one year i.e. you give queries like the following

```
select sum(amt) from sales where year=1991;
select product,sum(amt) from sales where year=1992
Group by product;
```

Now whenever you give queries like this Oracle will search the whole table. If you partition this table according to year, then the performance is improve since oracle will scan only a single partition instead of whole table.

CREATING PARTITION TABLES

To create a partition table give the following statement

```
create table sales (year number(4),
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
product varchar2(10),  
amt number(10,2))  
partition by range (year)  
partition p1 values less than (1992) tablespace u1,  
partition p2 values less than (1993) tablespace u2,  
partition p3 values less than (1994) tablespace u3,  
partition p4 values less than (1995) tablespace u4,  
partition p5 values less than (MAXVALUE) tablespace u5;
```

In the above example sales table is created with 5 partitions. Partition p1 will contain rows of year 1991 and

it will be stored in tablespace u1. Partition p2 will contain rows of year 1992 and it will be stored in tablespace u2.

Similarly p3 and p4.

In the above example if you don't specify the partition p4 with values less than MAXVALUE, then you will not be able to insert any row with year above 1994.

Although not required, you can place partitions in different tablespaces. If you place partitions in different tablespaces then you can isolate problems due to failures as only a particular partition will not be available and rest of the partitions will still be available.

The above example the table is partition by range.

In Oracle you can partition a table by

- Range Partitioning
- Hash Partitioning
- List Partitioning
- Composite Partitioning

Range Partitioning

This type of partitioning is useful when dealing with data that has logical ranges into which it can be distributed;

for example, value of year. Performance is best when the data evenly distributes across the range

Hash partitioning

Use hash partitioning if your data does not easily lend itself to range partitioning, but you would like to partition for performance and manageability reasons. Hash partitioning provides a method of evenly distributing data across a specified number of partitions. Rows are mapped into partitions based on a hash value of the partitioning key

The following example shows how to create a hash partition table.

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

The following example creates a hash-partitioned table. The partitioning column is partno, four partitions are created and assigned system generated names, and they are placed in four named tablespaces (tab1,tab2, ...).

```
CREATE TABLE products
(partno NUMBER,
description VARCHAR2 (60))
PARTITION BY HASH (partno)
PARTITIONS 4
STORE IN (tab1, tab2, tab3, tab4);
```

List Partitioning

Use list partitioning when you require explicit control over how rows map to partitions. You can specify a list of discrete values for the partitioning column in the description for each partition. This is different from range partitioning, where a range of values is associated with a partition, and from hash partitioning, where the user has no control of the row to partition mapping.

List partitioning allows unordered and unrelated sets of data to be grouped and organized together very naturally

The following example creates a table with list partitioning

```
Create table customers (custcode number(5),
Name varchar2(20),
Addr varchar2(10,2),
City varchar2(20),
Bal number(10,2))
Partition by list (city),
Partition north_India values ('DELHI','CHANDIGARH'),
Partition east_India values ('KOLKOTA','PATNA'),
Partition south_India values ('HYDERABAD','BANGALORE',
'CHENNAI'),
Partition west India values ('BOMBAY','GOA');
```

If a row is inserted in the above table then oracle maps the value of city column and whichever partition list matches the city column the row is stored in that partition.

COMPOSITE PARTITIONING

Composite partitioning partitions data using the range method, and within each partition, subpartitions it using the hash method. Composite partitions are ideal for both historical data and striping, and provide improved

manageability of range partitioning and data placement, as well as the parallelism advantages of hash partitioning.

When creating composite partitions, you specify the following:

- Partitioning method: range
- Partitioning column(s)
- Partition descriptions identifying partition bounds
- Subpartitioning method: hash
- Subpartitioning column(s)
- Number of subpartitions for each partition or descriptions of subpartitions

The following statement creates a composite-partitioned table. In this example, three range partitions are created, each

containing eight subpartitions. Because the subpartitions are not named, system generated names are assigned, but the

STORE IN clause distributes them across the 4 specified tablespaces (tab1, ...,tab4).

```
CREATE TABLE PRODUCTS (partno NUMBER,  
    description VARCHAR(32),  
    costprice NUMBER)  
PARTITION BY RANGE (partno)  
    SUBPARTITION BY HASH(description)  
    SUBPARTITIONS 8 STORE IN (tab1, tab2, tab3, tab4)  
    (PARTITION p1 VALUES LESS THAN (100),  
    PARTITION p2 VALUES LESS THAN (200),  
    PARTITION p3 VALUES LESS THAN (MAXVALUE));
```

ALTERING PARTITION TABLES

To add a partition

You can add add a new partition to the "high" end (the point after the last existing partition).

To add a partition

at the beginning or in the middle of a table, use the SPLIT PARTITION clause.

For example to add a partition to sales table give the following command.

```
alter table sales add partition p6 values less than (1996);
```

To add a partition to a Hash Partition table give the following command.

Alter table products add partition;

Then Oracle adds a new partition whose name is system generated and it is created in the default tablespace.

To add a partition by user define name and in your specified tablespace give the following command.

Alter table products add partition p5 tablespace u5;

To add a partition to a List partition table give the following command.

```
alter table customers add partition central_India  
values ('BHOPAL','NAGPUR');
```

Any value in the set of literal values that describe the partition(s) being added must not exist in any of the other partitions of the table.

Coalescing Partitions

Coalescing partitions is a way of reducing the number of partitions in a hash-partitioned table, or the number of subpartitions in a composite-partitioned table. When a hash partition is coalesced, its contents are redistributed into one or more remaining partitions determined by the hash function. The specific partition that is coalesced is selected by Oracle, and is dropped after its contents have been redistributed.

To coalesce a hash partition give the following statement.

```
Alter table products coalesce partition;
```

This reduces by one the number of partitions in the table products.

DROPPING PARTITIONS

To drop a partition from Range Partition table, List Partition or Composite Partition table give the following command.

```
Alter table sales drop partition p5;
```

Once you have drop the partition and if you have created a global index on the table. Then you have to rebuild the global index after dropping the partition by giving the following statement.

```
Alter index sales_ind rebuild;
```

To avoid rebuilding of indexes after dropping of the partitions you can also first delete all the records and then drop the partition like this

```
Delete from sales where year=1994;
```

```
Alter table sales drop partition p4;
```

This method is most appropriate for small tables, or for large tables when the partition being dropped contains a small percentage of the total data in the table.

Another method of dropping partitions is give the following statement.

```
ALTER TABLE sales DROP PARTITION p5 UPDATE GLOBAL INDEXES;
```

This causes the global index to be updated at the time the partition is dropped.

Exchanging a Range, Hash, or List Partition

To exchange a partition of a range, hash, or list-partitioned table with a nonpartitioned table, or the reverse, use the ALTER TABLE ... EXCHANGE PARTITION statement. An example of converting a partition into a nonpartitioned table follows. In this example, table stocks can be range, hash, or list partitioned.

```
ALTER TABLE stocks
```

```
EXCHANGE PARTITION p3 WITH stock_table_3;
```

Merging Partitions

Use the ALTER TABLE ... MERGE PARTITIONS statement to merge the contents of two partitions into one partition. The two original partitions are dropped, as are any corresponding local indexes.

You cannot use this statement for a hash-partitioned table or for hash subpartitions of a composite-partitioned table.

You can only merged two adjacent partitions, you cannot merge non adjacent partitions.

For example the merge the partition p2 and p3 into one partition p23 give the following statement.

Alter table sales merge partition p2 and p3 into partition p23;

Modifying Partitions: Adding Values

Use the MODIFY PARTITION ... ADD VALUES clause of the ALTER TABLE statement to extend the value list of an existing partition. Literal values being added must not have been included in any other partition's value list. The partition value list for any corresponding local index partition is correspondingly extended, and any global index, or global or local index partitions, remain usable.

The following statement adds a new set of cities ('KOCHI', 'MANGALORE') to an existing partition list.

```
ALTER TABLE customers
  MODIFY PARTITION south_india
    ADD VALUES ('KOCHI', 'MANGALORE');
```

Modifying Partitions: Dropping Values

Use the MODIFY PARTITION ... DROP VALUES clause of the ALTER TABLE statement to remove literal values from the value list of an existing partition. The statement is always executed with validation, meaning that it checks to see if any rows exist in the partition that correspond to the set of values being dropped. If any such rows are found then Oracle returns an error message and the operation fails. When necessary, use a DELETE statement to delete corresponding rows from the table before attempting to drop values.

You cannot drop all literal values from the value list describing the partition. You must use the ALTER TABLE ... DROP PARTITION statement instead.

The partition value list for any corresponding local index partition reflects the new value list, and any global index, or global or local index partitions, remain usable.

The statement below drops a set of cities ('KOCHI' and 'MANGALORE') from an existing partition value list.

```
ALTER TABLE customers
  MODIFY PARTITION south_india
    DROP VALUES ('KOCHI','MANGALORE');
```

SPLITTING PARTITIONS

You can split a single partition into two partitions. For example to split the partition p5 of sales table into two partitions give the following command.

```
Alter table sales split partition p5 into
  (Partition p6 values less than (1996),
  Partition p7 values less then (MAXVALUE));
```

TRUNCATING PARTITON

Truncating a partition will delete all rows from the partition.

To truncate a partition give the following statement

Alter table sales truncate partition p5;

LISTING INFORMATION ABOUT PARTITION TABLES

To see how many partitioned tables are there in your schema give the following statement

```
Select * from user_part_tables;
```

To see on partition level partitioning information

```
Select * from user_tab_partitions;
```

#####170- Monitora status do RMAN enquanto o backup está rodando

```
SELECT to_char(start_time,'DD-MON-YY HH24:MI') "BACKUP STARTED",
       sofar, totalwork,
       elapsed_seconds/60 "ELAPSE (Min)",
       round(sofar/totalwork*100,2) "Complete%"
FROM   sys.v_$session_longops
WHERE  opname = 'dbms_backup_restore'
```

--Completa informação de backups

```
SELECT sid, opname, context, sofar, totalwork, ROUND(sofar/totalwork, 2) "%Complete"
FROM   V$SESSION_LONGOPS
WHERE  opname LIKE 'RMAN%'
AND    opname LIKE '%aggregate%'
AND    totalwork != 0
AND    sofar <> totalwork;
```

-- MODELO 2

```
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"
col event              for a40
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle


```
col client_info for a30
select client_info
      , event
      from v$session
     where event like 'Backup%'
     order by client_info;
-- MOSTRA TOTAL E ESTIMATIVA DE TÉRMINO
select recid
      , output_device_type
      , dbsize_mbytes
      , input_bytes/1024/1024 input_mbytes
      , output_bytes/1024/1024 output_mbytes
      , (output_bytes/input_bytes*100) compression
      , (mbytes_processed/dbsize_mbytes*100) complete
      , to_char(start_time + (sysdate-start_time)/(mbytes_processed/dbsize_mbytes), 'DD-MON-
YYYY HH24:MI:SS') est_complete
     from v$rman_status rs
      , (select sum(bytes)/1024/1024 dbsize_mbytes from v$datafile)
     where status='RUNNING'
      and output_device_type is not null
```

#####171- Testando a eficiência da Library Cache

```
SELECT ROUND(SUM(PINHITS)/SUM(PINS) * 100,2) "Eficiência - Library Cache"
FROM V$LIBRARYCACHE;
```

```
--- Variação
select sum(pins) "Executions",
       sum(reloads) "Cache Misses While Executing",
       (sum(reloads)/sum(pins))*100 "Ratio"
from v$librarycache;
```

Obs.: Caso seja maior aumentar a SHARED_POOL_AREA

```
#####172- Setando coleta de estatística automática
BEGIN
dbms_stats.set_global_prefs('estimate_percent', '30');
END;
```

#####173-Criando Job para coleta de estatísticas de schema:

```
BEGIN
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
sys.dbms_scheduler.create_job(
job_name => ""SYS"."COLETA_STATS",
job_type => 'PLSQL_BLOCK',
job_action => 'begin
dbms_stats.gather_schema_stats(
ownname      => 'LF',
estimate_percent  => DBMS_STATS.AUTO_SAMPLE_SIZE,
method_opt    => 'for all columns size 1",
degree        => 8);
end;',
repeat_interval => 'FREQ=DAILY;BYHOUR=4;BYMINUTE=0;BYSECOND=0',
start_date => to_timestamp_tz('2011-12-07 -4:00', 'YYYY-MM-DD TZh:TzM'),
job_class => ""DEFAULT_JOB_CLASS",
comments => 'coleta estatísticas de tabelas LF',
auto_drop => FALSE,
enabled => FALSE);
sys.dbms_scheduler.set_attribute( name => ""SYS"."COLETA_STATS", attribute =>
'job_weight', value => 1);
sys.dbms_scheduler.set_attribute( name => ""SYS"."COLETA_STATS", attribute => 'restartable',
value => TRUE);
sys.dbms_scheduler.enable( ""SYS"."COLETA_STATS" );
END;
```

-- variação:

```
BEGIN
SYS.DBMS_SCHEDULER.DROP_JOB
(job_name => 'SYSTEM.STATISTICS_LF');
END;
/

BEGIN
SYS.DBMS_SCHEDULER.CREATE_JOB
(
job_name      => 'SYSTEM.STATISTICS_LF'
,start_date   => TO_TIMESTAMP_TZ('2011/08/19 17:25:07.495033 -03:00','yyyy/mm/dd
hh24:mi:ss.ff tzh:tzm')
,repeat_interval => 'FREQ=WEEKLY;BYDAY=FRI;BYHOUR=19;BYMINUTE=0;BYSECOND=0'
,end_date      => NULL
,job_class     => 'DEFAULT_JOB_CLASS'
,job_type      => 'PLSQL_BLOCK'
,job_action    => 'begin
dbms_stats.gather_schema_stats(ownname => ""LF"", estimate_percent => 30, cascade =>
TRUE);
end;'
,comments      => 'Statistics of owner LF'
);
SYS.DBMS_SCHEDULER.SET_ATTRIBUTE
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
( name    => 'SYSTEM.STATISTICS_LF'
,attribute => 'RESTARTABLE'
,value    => TRUE);
SYS.DBMS_SCHEDULER.SET_ATTRIBUTE
( name    => 'SYSTEM.STATISTICS_LF'
,attribute => 'LOGGING_LEVEL'
,value    => SYS.DBMS_SCHEDULER.LOGGING_RUNS);
SYS.DBMS_SCHEDULER.SET_ATTRIBUTE_NULL
( name    => 'SYSTEM.STATISTICS_LF'
,attribute => 'MAX_FAILURES');
SYS.DBMS_SCHEDULER.SET_ATTRIBUTE_NULL
( name    => 'SYSTEM.STATISTICS_LF'
,attribute => 'MAX_RUNS');
BEGIN
  SYS.DBMS_SCHEDULER.SET_ATTRIBUTE
  ( name    => 'SYSTEM.STATISTICS_LF'
  ,attribute => 'STOP_ON_WINDOW_CLOSE'
  ,value    => FALSE);
EXCEPTION
  -- could fail if program is of type EXECUTABLE...
  WHEN OTHERS THEN
    NULL;
END;
SYS.DBMS_SCHEDULER.SET_ATTRIBUTE
( name    => 'SYSTEM.STATISTICS_LF'
,attribute => 'JOB_PRIORITY'
,value    => 3);
SYS.DBMS_SCHEDULER.SET_ATTRIBUTE_NULL
( name    => 'SYSTEM.STATISTICS_LF'
,attribute => 'SCHEDULE_LIMIT');
SYS.DBMS_SCHEDULER.SET_ATTRIBUTE
( name    => 'SYSTEM.STATISTICS_LF'
,attribute => 'AUTO_DROP'
,value    => FALSE);
END;
/
```

#####174- Script para contagem de linha de todas as tabelas de um owner (faz em massa)

```
DECLARE
t_command varchar2(200);
t_total_records integer;
t_limit integer := 0; -- Mostra as tabelas que tenham mais linhas do que 0
BEGIN
for r in (select OWNER, TABLE_NAME from dba_tables where owner = '&OWNER' order by
table_name)
loop
t_command := 'SELECT COUNT(*) FROM '||r.OWNER||'.'||r.TABLE_NAME;
EXECUTE IMMEDIATE t_command INTO t_total_records;
if t_total_records > t_limit then
DBMS_OUTPUT.PUT_LINE(rpad(r.table_name,55,' ')||to_char(t_total_records,'99999999')||'
record(s)');
end if;
end loop;
END;
```

#####175- Mostrar hidden parameters (parâmetros Ocultos Oracle) conectado como SYS

```
select
a.ksppinm "Parameter",
b.kspstvl "Session Value",
c.kspstvl "Instance Value"
from
x$kspai a, x$kspcv b, x$kspsv c
where a.indx = b.indx
and a.indx = c.indx
and a.ksppinm like '/_%' escape '/';
```

--Variação

```
COLUMN parameter          FORMAT a37
COLUMN description        FORMAT a30 WORD_WRAPPED
COLUMN "Session VALUE"    FORMAT a10
COLUMN "Instance VALUE"   FORMAT a10
SET LINES 100 PAGES 0
```

```
SELECT
a.ksppinm "Parameter",
a.kspdesc "Description",
b.kspstvl "Session Value",
c.kspstvl "Instance Value"
FROM
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
x$ksppi a,  
x$ksppcv b,  
x$ksppsv c  
WHERE  
a.indx = b.indx  
AND  
a.indx = c.indx
```

#####176- Verificar objetos em cache.

```
select * from v$db_object_cache
```

#####177 – Descobrir objetos com erros de compilação e os erros

```
select * from dba_errors  
where owner = 'LF'  
and text not in ('PL/SQL: SQL Statement ignored','PL/SQL: Item ignored','PL/SQL: Statement  
ignored')  
and text not like 'PLS-00341%'  
and text not like 'PLS-00320%'  
and text not like 'PLS-00905%'  
and text not like 'PLS-00364%'  
order by type, name, sequence, line
```

#####178 – Trigger que registra CREATE ou DROP

```
CREATE OR REPLACE TRIGGER LF.tr_sati_create_object  
AFTER CREATE OR DROP ON SCHEMA  
DECLARE  
v_error VARCHAR2(200) := NULL;  
v_event varchar2(200) := NULL;  
BEGIN  
  
SELECT ora_sysevent  
INTO v_event  
from dual;  
  
INSERT INTO sati_object_control (object_name,  
event,  
timestamp,  
hour,  
object_type,  
os_user,  
bd_user,  
terminal,  
application#)  
VALUES (sys.dictionary_obj_name,
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```

        v_event,
        sysdate,
        to_char(sysdate, 'hh24:mi:ss'),
        sys.dictionary_obj_type,
        sys_context('USERENV', 'OS_USER'),
        sys_context('USERENV', 'SESSION_USER')
    ,
        sys_context('USERENV', 'TERMINAL'),
        1);
EXCEPTION
WHEN DUP_VAL_ON_INDEX THEN
    UPDATE sati_object_control
    SET timestamp    = sysdate,
        hour         = to_char(sysdate, 'hh24:mi:ss'),
        os_user      = sys_context('USERENV', 'OS_USER'),
        bd_user      = sys_context('USERENV', 'SESSION_USER'),
        terminal      = sys_context('USERENV', 'TERMINAL'),
        application# = nvl(application#,1) + 1
    WHERE object_name = sys.dictionary_obj_name
    AND object_type = sys.dictionary_obj_type
    AND event         = v_event;
WHEN OTHERS THEN
    v_error := SQLERRM;
    RAISE_APPLICATION_ERROR(-20001, v_error);
END;
/

```

--Criação da tabela SATI_OBJECT_CONTROL:

```

CREATE TABLE LF.SATI_OBJECT_CONTROL
(
    OBJECT_NAME VARCHAR2(500 BYTE)      NOT NULL,
    TIMESTAMP    DATE,
    HOUR         VARCHAR2(8 BYTE),
    OBJECT_TYPE  VARCHAR2(200 BYTE)     NOT NULL,
    OS_USER      VARCHAR2(200 BYTE),
    BD_USER      VARCHAR2(200 BYTE),
    TERMINAL     VARCHAR2(30 BYTE),
    EVENT        VARCHAR2(200 BYTE)     DEFAULT ' '      NOT NULL,
    APPLICATION# NUMBER
)
TABLESPACE TS_SATP_DAT_1M
PCTUSED 0
PCTFREE 10
INITTRANS 1
MAXTRANS 255
STORAGE (
    INITIAL      3M
    NEXT         1M
    MINEXTENTS   1
)

```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
MAXEXTENTS    UNLIMITED
PCTINCREASE    0
BUFFER_POOL    DEFAULT
);
```

COMMENT ON TABLE LF.SATI_OBJECT_CONTROL IS 'Tabela de Controle de Atualizacao de Objetos no Banco de Dados';

COMMENT ON COLUMN LF.SATI_OBJECT_CONTROL.OBJECT_NAME IS 'Nome do Objeto';

COMMENT ON COLUMN LF.SATI_OBJECT_CONTROL.TIMESTAMP IS 'Data de Atualizacao do Objeto';

COMMENT ON COLUMN LF.SATI_OBJECT_CONTROL.HOUR IS 'Hora de Atualizacao do Objeto';

COMMENT ON COLUMN LF.SATI_OBJECT_CONTROL.OBJECT_TYPE IS 'Tipo de Objeto';

COMMENT ON COLUMN LF.SATI_OBJECT_CONTROL.OS_USER IS 'Usuario do Sistema Operacional';

COMMENT ON COLUMN LF.SATI_OBJECT_CONTROL.BD_USER IS 'Usuario do Banco de Dados';

COMMENT ON COLUMN LF.SATI_OBJECT_CONTROL.TERMINAL IS 'Terminal que iniciou a sessao e executou o processo';

COMMENT ON COLUMN LF.SATI_OBJECT_CONTROL.EVENT IS 'Evento realizado sobre o objeto';

COMMENT ON COLUMN LF.SATI_OBJECT_CONTROL.APPLICATION# IS 'Numero de Aplicacoes no Objeto';

--

-- PK_OBJECT_CONTROL (Index)

--

CREATE UNIQUE INDEX LF.PK_OBJECT_CONTROL ON LF.SATI_OBJECT_CONTROL
(OBJECT_NAME, OBJECT_TYPE, EVENT, TIMESTAMP, HOUR)

TABLESPACE TS_LF_IX

PCTFREE 10

INITRANS 2

MAXTRANS 255

STORAGE (

 INITIAL 3M

 NEXT 1M

 MINEXTENTS 1

 MAXEXTENTS UNLIMITED

 PCTINCREASE 0

 BUFFER_POOL DEFAULT

);

```
--  
-- Non Foreign Key Constraints for Table SATI_OBJECT_CONTROL  
--  
ALTER TABLE LF.SATI_OBJECT_CONTROL ADD (  
  CONSTRAINT PK_OBJECT_CONTROL  
  PRIMARY KEY  
  (OBJECT_NAME, OBJECT_TYPE, EVENT, TIMESTAMP, HOUR)  
  USING INDEX  
  TABLESPACE TS_LF_IX  
  PCTFREE 10  
  INITRANS 2  
  MAXTRANS 255  
  STORAGE (  
    INITIAL 3M  
    NEXT 1M  
    MINEXTENTS 1  
    MAXEXTENTS UNLIMITED  
    PCTINCREASE 0  
  ) DISABLE);
```

#####179 – Cria JOB para executar procedure de limpa PRIS

```
DECLARE N_JOB NUMBER;  
BEGIN  
  SYS.DBMS_JOB.SUBMIT  
  ( N_JOB,  
    'P_EXCLUI_LOG_PRIS;',  
    TO_DATE('27/09/2009','DD/MM/YYYY'),  
    'TRUNC (SYSDATE + 1) + 7/24'  
  );  
  COMMIT;  
  END;  
  /
```

#####180 alterar nome ou local de datafiles em um tablespace;

1- colocar o tablespace em modo off-line;

```
ALTER TABLESPACE TS_IDX_01 ONLINE;
```

2- mover o arquivo para o local correto e/ou com nome novo

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle


```
mv /u01/disk2/ts_idx_01_02.dbf /u01/disk2/datafile/ts_idx_01_2.dbf
```

3- alterar no banco de dados com o comando abaixo:

```
ALTER DATABASE RENAME FILE '/u01/disk2/ts_idx_01_02.dbf' TO  
'/u01/disk2/datafile/ts_idx_01_2.dbf'
```

4- colocar o tablespace online:

```
ALTER TABLESPACE TS_IDX_01 ONLINE;
```

#####181- Alterando o datafile de um tablespace para outro filesystem

1- shutdown immediate;

2- startup mount;

Após o banco montar copie os arquivos fisicos para a nova localidade/diretorio desejado com os comandos de sistema operacional no seu caso;

```
3- cp /diretorio_atual/*.dbf /novo_diretorio/*.dbf
```

Após finalizar a copia dos arquivos emita os comando no sql

4- alter database

```
rename file '/caminho_atual/*.dbf' to /novo_caminho/*.dbf;
```

5- alter database open;

Assim que a instancia subir consulte a visao v\$datafile, e verifique se os datafiles estao no novo diretorio

Caso estejam pode apagar o no /opt para liberar o espaço.

#####182- Criando index buffer em uma área de 32K de buffer

Create an index buffer

It's easy to create a separate index buffer in Oracle9i and we can perform the operation while the database is active. We start by moving all indexes to a separate tablespace, defined to a separate data cache and then set optimizer_index_caching to the correct value.

- Allocate a 32k cache buffer - Start by creating a region of RAM for a 32k data cache.
- alter system set db_32k_cache_size = 100m;
- Allocate a 32k tablespace - Here we use the blocksize argument to associate the tablespace with the data buffer. Note that with this syntax we are using Oracle Managed Files (OMF), so we do not need to specify the data file name:
 - create tablespace index_ts_32k blocksize 32k;
 - Move the indexes into the 32k tablespace - This command moves the indexes into the 32k tablespace with no interruption to existing index queries. It rebuilds the indexes as temporary segments, and makes sure that the new index is usable before dropping the old index.

- alter index cust_idx rebuild online tablespace index_ts_32k;

Now that the indexes are segregated into a separate tablespace and index buffer, we can run dictionary scripts to predict with relative accuracy, the amount of the indexes that we can expect to see in the RAM index buffer.

```
select
  value - blocks optimizer_index_caching
from
  v$parameter p,
  dba_segments s
where
  name = 'db_32k_cache_size'
and
  tablespace_name = 'INDEX_TS_32K';
```

This estimated value will provide a good average for the amount of an index that can be expected to reside in the index cache. This assume equal index access by the application, but you can query the v\$bh view to make sure that there is no skew in index access.

Conclusion

Oracle provides many tools for the Oracle professional to help the CBO always make the best decision about the way to access Oracle data. By working toward the optimal settings you can ensure that the majority of your SQL always executes quickly and efficiently.

#####183- Descobrir quais queries estão gerando mais parse

```
--1
SELECT PLAN_HASH_VALUE,COUNT(*)
FROM V$SQL
WHERE PARSING_SCHEMA_NAME NOT IN
('SYS','DBSNMP','SYSMAN')
GROUP BY PLAN_HASH_VALUE ORDER BY 2 desc;
```

```
--2
SELECT SQL_TEXT,EXECUTIONS
FROM V$SQLAREA
WHERE PLAN_HASH_VALUE = &hash_coletado_acima;
```

#####184- Verifica constraint FK que estão sem índices. (By T.Kyte)

```
select table_name, constraint_name,
       cname1 || nvl2(cname2,', ' || cname2,null) ||
       nvl2(cname3,', ' || cname3,null) || nvl2(cname4,', ' || cname4,null) ||
       nvl2(cname5,', ' || cname5,null) || nvl2(cname6,', ' || cname6,null) ||
       nvl2(cname7,', ' || cname7,null) || nvl2(cname8,', ' || cname8,null)
       columns
from ( select b.table_name,
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```

        b.constraint_name,
        max(decode( position, 1, column_name, null )) cname1,
        max(decode( position, 2, column_name, null )) cname2,
        max(decode( position, 3, column_name, null )) cname3,
        max(decode( position, 4, column_name, null )) cname4,
        max(decode( position, 5, column_name, null )) cname5,
        max(decode( position, 6, column_name, null )) cname6,
        max(decode( position, 7, column_name, null )) cname7,
        max(decode( position, 8, column_name, null )) cname8,
        count(*) col_cnt
    from (select substr(table_name,1,30) table_name,
        substr(constraint_name,1,30) constraint_name,
        substr(column_name,1,30) column_name,
        position
        from user_cons_columns ) a,
        user_constraints b
    where a.constraint_name = b.constraint_name
        and b.constraint_type = 'R'
    group by b.table_name, b.constraint_name
) cons
where col_cnt > ALL
( select count(*)
  from user_ind_columns i
  where i.table_name = cons.table_name
    and i.column_name in (cname1, cname2, cname3, cname4,
                          cname5, cname6, cname7, cname8 )
    and i.column_position <= cons.col_cnt
  group by i.index_name
)

```

#####185- Inclui índice para foreign key sem índice

-- OBS: melhora problemas de LOCK TM e até alguns casos de deadlock

```

select 'create index IX_' || columns || '_' || (substr(TABLE_NAME,1,7)) || ' on
' || TABLE_NAME || '(' || columns || ')' tablespace TS_IDX_02;' from( select table_name,
constraint_name,
cname1 || nvl2(cname2,', ' || cname2,null) ||
nvl2(cname3,', ' || cname3,null) || nvl2(cname4,', ' || cname4,null) ||
nvl2(cname5,', ' || cname5,null) || nvl2(cname6,', ' || cname6,null) ||
nvl2(cname7,', ' || cname7,null) || nvl2(cname8,', ' || cname8,null)
columns
from ( select b.table_name,
        b.constraint_name,
        max(decode( position, 1, column_name, null )) cname1,
        max(decode( position, 2, column_name, null )) cname2,
        max(decode( position, 3, column_name, null )) cname3,
        max(decode( position, 4, column_name, null )) cname4,

```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```

max(decode( position, 5, column_name, null )) cname5,
max(decode( position, 6, column_name, null )) cname6,
max(decode( position, 7, column_name, null )) cname7,
max(decode( position, 8, column_name, null )) cname8,
count(*) col_cnt
from (select substr(table_name,1,30) table_name,
      substr(constraint_name,1,30) constraint_name,
      substr(column_name,1,30) column_name,
      position
      from user_cons_columns ) a,
      user_constraints b
where a.constraint_name = b.constraint_name
      and b.constraint_type = 'R'
group by b.table_name, b.constraint_name
) cons
where col_cnt > ALL
( select count(*)
  from user_ind_columns i
  where i.table_name = cons.table_name
        and i.column_name in (cname1, cname2, cname3, cname4,
                               cname5, cname6, cname7, cname8 )
        and i.column_position <= cons.col_cnt
  group by i.index_name
));

```

186- Criando scheduler para executar determinada procedure

```

BEGIN
sys.dbms_scheduler.create_job(
job_name => '"LF"."EXECUTA_P_INTERFACE_SAP"',
job_type => 'PLSQL_BLOCK',
job_action => 'begin
  P_INTERFACE_SAP;
end;',
repeat_interval => 'FREQ=DAILY;BYHOUR=23;BYMINUTE=0;BYSECOND=0',
start_date => to_timestamp_tz('2012-06-27 -3:00', 'YYYY-MM-DD TZH:TZM'),
job_class => '"DEFAULT_JOB_CLASS"',
comments => 'executa p_interface_SAP',
auto_drop => FALSE,
enabled => FALSE);
sys.dbms_scheduler.set_attribute( name => '"LF"."EXECUTA_P_INTERFACE_SAP"', attribute =>
'restartable', value => TRUE);
sys.dbms_scheduler.enable( '"LF"."EXECUTA_P_INTERFACE_SAP"' );

```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

END;

187- Histórico das conexões

```

SELECT  ROUND(BITAND(s.ownerid, 65535) ) parent_session_sid,
        ROUND(BITAND(s.ownerid, 16711680) / 65536) parent_session_instid, s.saddr, s.SID,
s.serial#, s.audsid, s.paddr,
        s.user#, s.username, s.command, s.ownerid, s.taddr, s.lockwait, s.status, s.server,
s.schema#, s.schemaname,
        s.osuser, s.process, s.machine, s.port, s.terminal, UPPER(s.program) program, s.TYPE,
s.sql_address,
        s.sql_hash_value, s.sql_id, s.sql_child_number, s.prev_sql_addr, s.prev_hash_value,
s.prev_sql_id,
        s.prev_child_number, s.plsql_entry_object_id, s.plsql_entry_subprogram_id,
s.plsql_object_id,
        s.plsql_subprogram_id, s.module, s.module_hash, s.action, s.action_hash, s.client_info,
s.fixed_table_sequence,
        s.row_wait_obj#, s.row_wait_file#, s.row_wait_block#, s.row_wait_row#, s.logon_time,
s.last_call_et,
        s.pdml_enabled, s.failover_type, s.failover_method, s.failed_over,
s.resource_consumer_group, s.pdml_status,
        s.pddl_status, s.pq_status, s.current_queue_duration, s.client_identifier,
s.blocking_session_status,
        s.blocking_instance, s.blocking_session, s.seq#, s.event#, s.event, s.p1text, s.p1, s.p1raw,
s.p2text, s.p2,
        s.p2raw, s.p3text, s.p3, s.p3raw, s.wait_class_id, s.wait_class#, s.wait_class, s.wait_time,
s.seconds_in_wait,
        s.state, s.service_name, s.sql_trace, s.sql_trace_waits, s.sql_trace_binds, s.ecid,
stat.cpu - stat.cpu_this_call_start cpu_this_call, stat.cpu, stat.uga_memory,
stat.pga_memory, stat.commits,
        stat.rollsbacks, si.block_gets, si.consistent_gets, si.physical_reads, si.block_changes,
si.consistent_changes,
        p.program "OSProgram", p.spid, p.pid
FROM v$session s,
     v$ses_io si,
     v$process p,
     (SELECT  ss.SID stat_sid, SUM(Decode(sn.NAME, 'CPU used when call started', ss.VALUE,
0) )
                                cpu_this_call_start,
            SUM(Decode(sn.NAME, 'CPU used by this session', ss.VALUE, 0) ) cpu,
            SUM(Decode(sn.NAME, 'session uga memory', ss.VALUE, 0) ) uga_memory,
            SUM(Decode(sn.NAME, 'session pga memory', ss.VALUE, 0) ) pga_memory,
            SUM(Decode(sn.NAME, 'user commits', ss.VALUE, 0) ) commits,
            SUM(Decode(sn.NAME, 'user rollsbacks', ss.VALUE, 0) ) rollsbacks
        FROM v$sesstat ss, v$statname sn
        WHERE ss.statistic# = sn.statistic#

```

```

AND(sn.NAME = 'CPU used when call started' OR sn.NAME = 'CPU used by this
session'
OR sn.NAME = 'session uga memory' OR sn.NAME = 'session pga memory' OR
sn.NAME = 'user commits'
OR sn.NAME = 'user rollbacks')
GROUP BY ss.SID) stat
WHERE (s.status = 'ACTIVE') AND(si.SID(+) = s.SID) AND(p.addr(+) = s.paddr) AND(stat.stat_sid
= s.SID)
ORDER BY "USERNAME", ownerid;
#####188- Fazendo select na tabela para saber como ela era em determinada data
select * from lf.lf_nf_saida_imposto
AS OF TIMESTAMP TO_TIMESTAMP
('2012-11-05 10:30:00', 'yyyy-mm-dd hh24:mi:ss')
where cod_matriz = '2500'
and cod_filial = '0168'
and num_nf = '002402'
and dt_emissao = '31/dec/2010'
and cod_imposto in ('09', '10')

```

#####189- Inserir as select colocando log em outra tabela e não fazendo rollback após erro.

Digamos que eu queria fazer um insert as select pra uma tabela, e tenham alguns erros e validação e tal, eu posso mandar o Oracle ignorar essas linhas e importar o que deu certo:

1- Crio LOG:

```

BEGIN
  DBMS_ERRLOG.create_error_log (dml_table_name => 'dest');
END;
/
2- Faço insert
INSERT INTO dest
SELECT *
FROM source
LOG ERRORS INTO err$_dest ('INSERT') REJECT LIMIT UNLIMITED;

```

3- verificar o log de erros:

```

COLUMN ora_err_mesg$ FORMAT A70
SELECT ora_err_number$, ora_err_mesg$
FROM err$_dest
WHERE ora_err_tag$ = 'INSERT';

```

Fonte: <http://www.oracle-base.com/articles/10g/dml-error-logging-10gr2.php>

190- Verificando propriedades do Banco de dados.

```
select * from database_properties
```

191- Eliminar job

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

Para colocar em BROKEN
EXEC DBMS_JOB.BROKEN(NUM_JOB,TRUE);
NUM_JOB pego na DBA_JOBS

Para matar
EXEC DBMS_JOB.REMOVE (NUM_JOB);

Criar instrução para matar em série
select 'EXEC DBMS_JOB.REMOVE(' || JOB || ');' from dba_jobs where what like 'begin
PCK_PRIS.%'

192- Criando Mview para ser atualizada a cada dia 02 de cada mês

```
CREATE MATERIALIZED VIEW ADMRMS.CEA_EMISS_NF_SYNCHRO  
NOCACHE  
LOGGING  
BUILD IMMEDIATE  
REFRESH COMPLETE ON DEMAND  
START WITH TO_DATE('02-jul-2013','dd-mon-yyyy')  
NEXT trunc(ADD_MONTHS(sysdate,1))  
AS  
SELECT ...
```

193- Usando Oracle Replay (RAT – Real Application Testing)

Verificar:

- Using Workload Capture and Replay in 11G (Doc ID 445116.1)
- Master Note for Real Application Testing Option (Doc ID 1464274.1)

```
BEGIN DBMS_WORKLOAD_REPLAY.PROCESS_CAPTURE (capture_dir => 'WORKLOAD_DIR');  
END;
```

```
wrc replaydir=/dbfs/FS1/poc/globosat/rat mode=calibrate
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
[oracle@ed03db05 rat]$ wrc replaydir=/dbfs/FS1/poc/globosat/rat mode=calibrate
```

Workload Replay Client: Release 11.2.0.3.0 - Production on Wed Sep 4 12:21:37 2013

Copyright (coffee) 1982, 2011, Oracle and/or its affiliates. All rights reserved.

Report for Workload in: /dbfs/FS1/poc/globosat/rat

Recommendation:

Consider using at least 5 clients divided among 2 CPU(s)

You will need at least 146 MB of memory per client process.

If your machine(s) cannot match that number, consider using more clients.

Workload Characteristics:

- max concurrency: 192 sessions

- total number of sessions: 1295

Assumptions:

- 1 client process per 50 concurrent sessions

- 4 client process per CPU

- 256 KB of memory cache per concurrent session

- think time scale = 100

- connect time scale = 100

- synchronization = TRUE

BEGIN

DBMS_WORKLOAD_REPLAY.INITIALIZE_REPLAY (replay_name => 'IBMSBETO',
replay_dir => 'WORKLOAD_DIR');

END;

BEGIN

DBMS_WORKLOAD_REPLAY.PREPARE_REPLAY (synchronization => TRUE,
sts_cap_interval => 300);

END;

wrc replaydir=/dbfs/FS1/poc/globosat/rat &
wrc system/gsat2013 replaydir=/dbfs/FS1/poc/globosat/rat &
wrc system/gsat2013 replaydir=/dbfs/FS1/poc/globosat/rat &
wrc system/gsat2013 replaydir=/dbfs/FS1/poc/globosat/rat &
wrc system/gsat2013 replaydir=/dbfs/FS1/poc/globosat/rat &
wrc system/gsat2013 replaydir=/dbfs/FS1/poc/globosat/rat &
wrc system/gsat2013 replaydir=/dbfs/FS1/poc/globosat/rat &
wrc system/gsat2013 replaydir=/dbfs/FS1/poc/globosat/rat &
wrc system/gsat2013 replaydir=/dbfs/FS1/poc/globosat/rat &
wrc system/gsat2013 replaydir=/dbfs/FS1/poc/globosat/rat &

BEGIN

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle


```
DBMS_WORKLOAD_REPLAY.START_REPLAY ();  
END;
```

```
BEGIN  
DBMS_WORKLOAD_REPLAY.RESUME_REPLAY();  
END;
```

GERAR RELATÓRIO REPLAY:

```
DECLARE  
cap_id    NUMBER;  
rep_id    NUMBER;  
rep_rpt    CLOB;  
t_out_file UTL_FILE.file_type;  
t_buffer   VARCHAR2(32767);  
t_amount   BINARY_INTEGER := 1000;  
t_pos      INTEGER := 1;  
t_clob_len INTEGER;  
P_DIR VARCHAR2(100) := 'WORKLOAD_DIR';  
P_FILE VARCHAR2(100) := 'replay.html';  
  
BEGIN  
cap_id := DBMS_WORKLOAD_REPLAY.GET_REPLAY_INFO(dir => 'WORKLOAD_DIR');  
/* Get the latest replay for that capture */  
SELECT max(id)  
INTO rep_id  
FROM dba_workload_replays  
WHERE capture_id = cap_id;  
  
rep_rpt := DBMS_WORKLOAD_REPLAY.REPORT(replay_id => rep_id,  
format => DBMS_WORKLOAD_REPLAY.TYPE_HTML);  
  
t_clob_len := DBMS_LOB.GetLength(rep_rpt);  
t_out_file := UTL_FILE.fOpen(p_dir,p_file, 'W', 32767);  
WHILE t_pos < t_clob_len LOOP  
  
    dbms_output.put_line(' t_pos before : ' || t_pos);  
    DBMS_LOB.Read(rep_rpt, t_amount, t_pos, t_buffer);  
    UTL_FILE.Put(t_out_file, t_buffer);  
    UTL_FILE.fflush(t_out_file);  
    dbms_output.put_line(' t_pos after : ' || t_pos);  
    t_pos := t_pos + t_amount;  
  
END LOOP;  
dbms_output.put_line('completed writing');  
UTL_FILE.fClose(t_out_file);  
EXCEPTION  
WHEN OTHERS THEN  
    IF(UTL_FILE.Is_Open(t_out_file))THEN  
        UTL_FILE.fClose(t_out_file);
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
END IF;  
RAISE;  
  
END;  
/
```

COMPARANDO COM O CAPTURE:

```
DECLARE  
cap_id    NUMBER;  
rep_id    NUMBER;  
rep_rpt   CLOB;  
t_out_file UTL_FILE.file_type;  
t_buffer  VARCHAR2(32767);  
t_amount  BINARY_INTEGER := 1000;  
t_pos     INTEGER := 1;  
t_clob_len INTEGER;  
P_DIR VARCHAR2(100) := 'WORKLOAD_DIR';  
P_FILE VARCHAR2(100) := 'replay_compare.html';  
  
BEGIN  
cap_id := DBMS_WORKLOAD_REPLAY.GET_REPLAY_INFO(dir => 'WORKLOAD_DIR');  
/* Get the latest replay for that capture */  
SELECT max(id)  
INTO rep_id  
FROM dba_workload_replays  
WHERE capture_id = cap_id;  
  
DBMS_WORKLOAD_REPLAY.COMPARE_PERIOD_REPORT (  
    replay_id1 => 1,  
    replay_id2 => null,  
    format => DBMS_WORKLOAD_CAPTURE.TYPE_HTML,  
    result => rep_rpt);  
  
t_clob_len := DBMS_LOB.GetLength(rep_rpt);  
t_out_file := UTL_FILE.fOpen(p_dir,p_file, 'W', 32767);  
WHILE t_pos < t_clob_len LOOP  
  
    dbms_output.put_line(' t_pos before : ' || t_pos);  
    DBMS_LOB.Read(rep_rpt, t_amount, t_pos, t_buffer);  
    UTL_FILE.Put(t_out_file, t_buffer);  
    UTL_FILE.fflush(t_out_file);  
    dbms_output.put_line(' t_pos after : ' || t_pos);  
    t_pos := t_pos + t_amount;  
  
END LOOP;  
dbms_output.put_line('completed writing');  
UTL_FILE.fClose(t_out_file);
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

EXCEPTION

WHEN OTHERS THEN

IF(UTL_FILE.Is_Open(t_out_file))THEN

UTL_FILE.fClose(t_out_file);

END IF;

RAISE;

END;

/

194- Automatic SQL Tuning in Oracle Database 10g

Automatic SQL Tuning in Oracle Database 10g

This article the discusses the new features which automate the tuning of SQL statements in Oracle 10g:

- Overview
- SQL Tuning Advisor
- Managing SQL Profiles
- SQL Tuning Sets
- Useful Views

Overview

In its normal mode the query optimizer needs to make decisions about execution plans in a very short time. As a result it may not always be able to obtain enough information to make the best decision. Oracle 10g allows the optimizer to run in tuning mode where it can gather additional information and make recommendations about how specific statements can be tuned further. This process may take several minutes for a single statement so it is intended to be used on high-load resource-intensive statements.

In tuning mode the optimizer performs the following analysis:

- Statistics Analysis - The optimizer recommends the gathering of statistics on objects with missing or stale statistics. Additional statistics for these objects are stored in an SQL profile.
- SQL Profiling - The optimizer may be able to improve performance by gathering additional statistics and altering session specific parameters such as the OPTIMIZER_MODE. If such improvements are possible the information is stored in an SQL profile. If accepted this information can then used by the optimizer when running in normal mode. Unlike a stored outline which fixes the execution plan, an SQL profile may still be of benefit when the contents of the table alter drastically. Even so, it's sensible to update profiles periodically. The SQL profiling is not performed when the tuining optimizer is run in limited mode.
- Access Path Analysis - The optimizer investigates the effect of new or modified indexes on the access path. It's index recommendations relate to a specific statement so where necessary it will also suggest the use of the SQL Access Advisor to check the impact of these indexes on a representative SQL workload.
- SQL Structure Analysis - The optimizer suggests alternatives for SQL statements that contain structures that may impact on performance. The implementation of these suggestions requires human intervention to check their validity.

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

The automatic SQL tuning features are accessible from Enterprise Manager on the "Advisor Central" page these or from PL/SQL using the DBMS_SQLTUNE package. This article will focus on the PL/SQL API as the Enterprise Manager interface is reasonably intuitive.

SQL Tuning Advisor

In order to access the SQL tuning advisor API a user must be granted the ADVISOR privilege.

```
CONN sys/password AS SYSDBA
```

```
GRANT ADVISOR TO scott;
```

```
CONN scott/tiger
```

The first step when using the SQL tuning advisor is to create a new tuning task using the CREATE_TUNING_TASK function. The statements to be analyzed can be retrieved from the Automatic Workload Repository (AWR), the cursor cache, an SQL tuning set or specified manually.

```
SET SERVEROUTPUT ON
```

```
-- Tuning task created for specific a statement from the AWR.
```

```
DECLARE
```

```
l_sql_tune_task_id VARCHAR2(100);
```

```
BEGIN
```

```
l_sql_tune_task_id := DBMS_SQLTUNE.create_tuning_task (
    begin_snap => 764,
    end_snap   => 938,
    sql_id      => '19v5guvsgcd1v',
    scope       => DBMS_SQLTUNE.scope_comprehensive,
    time_limit  => 60,
    task_name   => '19v5guvsgcd1v_AWR_tuning_task',
    description => 'Tuning task for statement 19v5guvsgcd1v in AWR.');
```

```
DBMS_OUTPUT.put_line('l_sql_tune_task_id: ' || l_sql_tune_task_id);
```

```
END;
```

```
/
```

```
-- Tuning task created for specific a statement from the cursor cache.
```

```
DECLARE
```

```
l_sql_tune_task_id VARCHAR2(100);
```

```
BEGIN
```

```
l_sql_tune_task_id := DBMS_SQLTUNE.create_tuning_task (
    sql_id      => '19v5guvsgcd1v',
    scope       => DBMS_SQLTUNE.scope_comprehensive,
    time_limit  => 60,
    task_name   => '19v5guvsgcd1v_tuning_task',
    description => 'Tuning task for statement 19v5guvsgcd1v.');
```

```
DBMS_OUTPUT.put_line('l_sql_tune_task_id: ' || l_sql_tune_task_id);
```

```
END;
```

```
/
```

```
-- Tuning task created from an SQL tuning set.
```

```
DECLARE
```

```
l_sql_tune_task_id VARCHAR2(100);
```

```
BEGIN
```

```
l_sql_tune_task_id := DBMS_SQLTUNE.create_tuning_task (
    sqlset_name => 'test_sql_tuning_set',
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```

        scope    => DBMS_SQLTUNE.scope_comprehensive,
        time_limit => 60,
        task_name => 'sqlset_tuning_task',
        description => 'Tuning task for an SQL tuning set. ');
DBMS_OUTPUT.put_line('l_sql_tune_task_id: ' || l_sql_tune_task_id);
END;
/

-- Tuning task created for a manually specified statement.
DECLARE
    l_sql          VARCHAR2(500);
    l_sql_tune_task_id VARCHAR2(100);
BEGIN
    l_sql := 'SELECT e.*, d.* ' ||
        'FROM emp e JOIN dept d ON e.deptno = d.deptno ' ||
        'WHERE NVL(empno, "0") = :empno';

    l_sql_tune_task_id := DBMS_SQLTUNE.create_tuning_task (
        sql_text    => l_sql,
        bind_list   => sql_binds(anydata.ConvertNumber(100)),
        user_name   => 'scott',
        scope       => DBMS_SQLTUNE.scope_comprehensive,
        time_limit  => 60,
        task_name   => 'emp_dept_tuning_task',
        description => 'Tuning task for an EMP to DEPT join query. ');
    DBMS_OUTPUT.put_line('l_sql_tune_task_id: ' || l_sql_tune_task_id);
END;
/

```

If the TASK_NAME parameter is specified it's value is returned as the SQL tune task identifier. If omitted a system generated name like "TASK_1478" is returned. If the SCOPE parameter is set to scope_limited the SQL profiling analysis is omitted. The TIME_LIMIT parameter simply restricts the time the optimizer can spend compiling the recommendations.

The following examples will reference the last tuning set as it has no external dependencies other than the SCOTT schema. The NVL in the SQL statement was put in to provoke a reaction from the optimizer. In addition we can delete the statistics from one of the tables to provoke it even more.

```
EXEC DBMS_STATS.delete_table_stats('SCOTT','EMP');
```

With the tuning task defined the next step is to execute it using the EXECUTE_TUNING_TASK procedure.

```
EXEC DBMS_SQLTUNE.execute_tuning_task(task_name => 'emp_dept_tuning_task');
```

During the execution phase you may wish to pause and restart the task, cancel it or reset the task to allow it to be re-executed.

-- Interrupt and resume a tuning task.

```
EXEC DBMS_SQLTUNE.interrupt_tuning_task (task_name => 'emp_dept_tuning_task');
```

```
EXEC DBMS_SQLTUNE.resume_tuning_task (task_name => 'emp_dept_tuning_task');
```

-- Cancel a tuning task.

```
EXEC DBMS_SQLTUNE.cancel_tuning_task (task_name => 'emp_dept_tuning_task');
```

-- Reset a tuning task allowing it to be re-executed.

```
EXEC DBMS_SQLTUNE.reset_tuning_task (task_name => 'emp_dept_tuning_task');
The status of the tuning task can be monitored using the DBA_ADVISOR_LOG view.
SELECT task_name, status FROM dba_advisor_log WHERE owner = 'SCOTT';
```

TASK_NAME	STATUS
emp_dept_tuning_task	COMPLETED

1 row selected.

Once the tuning task has executed successfully the recommendations can be displayed using the REPORT_TUNING_TASK function.

```
SET LONG 10000;
SET PAGESIZE 1000
SET LINESIZE 200
SELECT DBMS_SQLTUNE.report_tuning_task('emp_dept_tuning_task') AS recommendations
FROM dual;
SET PAGESIZE 24
```

In this case the output looks like this.

RECOMMENDATIONS

GENERAL INFORMATION SECTION

```
Tuning Task Name : emp_dept_tuning_task
Scope           : COMPREHENSIVE
Time Limit(seconds): 60
Completion Status : COMPLETED
Started at      : 05/06/2004 09:29:13
Completed at    : 05/06/2004 09:29:15
```

SQL ID : 0wrmfv2yvswx1

SQL Text: SELECT e.*, d.* FROM emp e JOIN dept d ON e.deptno = d.deptno
WHERE NVL(empno, '0') = :empno

FINDINGS SECTION (2 findings)

1- Statistics Finding

Table "SCOTT"."EMP" and its indices were not analyzed.

Recommendation

Consider collecting optimizer statistics for this table and its indices.
execute dbms_stats.gather_table_stats(ownname => 'SCOTT', tabname =>
'EMP', estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE,
method_opt => 'FOR ALL COLUMNS SIZE AUTO', cascade => TRUE)

Rationale

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

The optimizer requires up-to-date statistics for the table and its indices in order to select a good execution plan.

2- Restructure SQL finding (see plan 1 in explain plans section)

The predicate `NVL("E"."EMPNO",0)=:B1` used at line ID 2 of the execution plan contains an expression on indexed column "EMPNO". This expression prevents the optimizer from selecting indices on table "SCOTT"."EMP".

Recommendation

Rewrite the predicate into an equivalent form to take advantage of indices. Alternatively, create a function-based index on the expression.

Rationale

The optimizer is unable to use an index if the predicate is an inequality condition or if there is an expression or an implicit data type conversion on the indexed column.

EXPLAIN PLANS SECTION

1- Original

Plan hash value: 1863486531

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		1	107	4 (0)	00:00:01
1	NESTED LOOPS		1	107	4 (0)	00:00:01
2	TABLE ACCESS FULL	EMP	1	87	3 (0)	00:00:01
3	TABLE ACCESS BY INDEX ROWID	DEPT	1	20	1 (0)	00:00:01
4	INDEX UNIQUE SCAN	PK_DEPT	1		0 (0)	00:00:01

Note

- dynamic sampling used for this statement

1 row selected.

Once the tuning session is over the tuning task can be dropped using the `DROP_TUNING_TASK` procedure.

BEGIN

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
DBMS_SQLTUNE.drop_tuning_task (task_name => '19v5guvsgcd1v_AWR_tuning_task');
DBMS_SQLTUNE.drop_tuning_task (task_name => '19v5guvsgcd1v_tuning_task');
DBMS_SQLTUNE.drop_tuning_task (task_name => 'sqlset_tuning_task');
DBMS_SQLTUNE.drop_tuning_task (task_name => 'emp_dept_tuning_task');
END;
/
```

Managing SQL Profiles

To manage SQL profiles a user needs the following privileges.

```
CONN sys/password AS SYSDBA
GRANT CREATE ANY SQL PROFILE TO scott;
GRANT DROP ANY SQL PROFILE TO scott;
GRANT ALTER ANY SQL PROFILE TO scott;
CONN scott/tiger
```

If the recommendations of the SQL tuning advisor include a suggested profile you can choose to accept it using the ACCEPT_SQL_PROFILE procedure.

```
SET SERVEROUTPUT ON
```

```
DECLARE
  l_sql_tune_task_id VARCHAR2(20);
BEGIN
  l_sql_tune_task_id := DBMS_SQLTUNE.accept_sql_profile (
    task_name => 'emp_dept_tuning_task',
    name      => 'emp_dept_profile');
  DBMS_OUTPUT.put_line('l_sql_tune_task_id: ' || l_sql_tune_task_id);
END;
/
```

The NAME parameter is used to specify a name for the profile. If it is not specified a system generated name will be used.

The STATUS, NAME, DESCRIPTION, and CATEGORY attributes of an SQL profile can be altered using the ALTER_SQL_PROFILE procedure.

```
BEGIN
  DBMS_SQLTUNE.alter_sql_profile (
    name      => 'emp_dept_profile',
    attribute_name => 'STATUS',
    value      => 'DISABLED');
END;
/
```

Existing SQL profiles can be dropped using the DROP_SQL_PROFILE procedure.

```
BEGIN
  DBMS_SQLTUNE.drop_sql_profile (
    name      => 'emp_dept_profile',
    ignore => TRUE);
END;
/
```

The IGNORE parameter prevents errors being reported if the specified profile does not exist.

SQL Tuning Sets

An SQL tuning set is a group of statements along with their execution context. These can be created automatically via Enterprise Manager or manually provided you have the necessary privileges.

```
CONN sys/password AS SYSDBA
GRANT ADMINISTER ANY SQL TUNING SET TO scott;
```


CONN scott/tiger

An SQL tuning set is created using the CREATE_SQLSET procedure.

```
BEGIN
```

```
  DBMS_SQLTUNE.create_sqlset (  
    sqlset_name => 'test_sql_tuning_set',  
    description => 'A test SQL tuning set.');
```

```
END;
```

```
/
```

Statements are added to the set using the LOAD_SQLSET procedure which accepts a REF CURSOR of statements retrieved using one of the following pipelined functions:

- SELECT_WORKLOAD_REPOSITORY - Retrieves statements from the Automatic Workload Repository (AWR).
- SELECT_CURSOR_CACHE - Retrieves statements from the cursor cache.
- SELECT_SQLSET - Retrieves statements from another SQL tuning set.

The following are examples of their usage.

-- Load the SQL set from the Automatic Workload Repository (AWR).

```
DECLARE
```

```
  l_cursor DBMS_SQLTUNE.sqlset_cursor;
```

```
BEGIN
```

```
  OPEN l_cursor FOR
```

```
    SELECT VALUE(p)
```

```
  FROM  TABLE (DBMS_SQLTUNE.select_workload_repository (  
    765, -- begin_snap
```

```
    766, -- end_snap
```

```
    NULL, -- basic_filter
```

```
    NULL, -- object_filter
```

```
    NULL, -- ranking_measure1
```

```
    NULL, -- ranking_measure2
```

```
    NULL, -- ranking_measure3
```

```
    NULL, -- result_percentage
```

```
    10) -- result_limit
```

```
  ) p;
```

```
  DBMS_SQLTUNE.load_sqlset (  
    sqlset_name  => 'test_sql_tuning_set',  
    populate_cursor => l_cursor);
```

```
END;
```

```
/
```

-- Load the SQL set from the cursor cache.

```
DECLARE
```

```
  l_cursor DBMS_SQLTUNE.sqlset_cursor;
```

```
BEGIN
```

```
  OPEN l_cursor FOR
```

```
    SELECT VALUE(p)
```

```
  FROM  TABLE (DBMS_SQLTUNE.select_cursor_cache (  
    NULL, -- basic_filter
```

```
    NULL, -- object_filter
```

```
    NULL, -- ranking_measure1
```

```
    NULL, -- ranking_measure2
```

```

        NULL, -- ranking_measure3
        NULL, -- result_percentage
        1) -- result_limit
    ) p;

DBMS_SQLTUNE.load_sqlset (
    sqlset_name => 'test_sql_tuning_set',
    populate_cursor => l_cursor);
END;
/

-- Create a new set and load it from the existing one.
DECLARE
    l_cursor DBMS_SQLTUNE.sqlset_cursor;
BEGIN
    DBMS_SQLTUNE.create_sqlset(
        sqlset_name => 'test_sql_tuning_set_2',
        description => 'Another test SQL tuning set.');
```

OPEN l_cursor FOR

```

    SELECT VALUE(p)
    FROM   TABLE (DBMS_SQLTUNE.select_sqlset (
        'test_sql_tuning_set', -- sqlset_name
        NULL,                -- basic_filter
        NULL,                -- object_filter
        NULL,                -- ranking_measure1
        NULL,                -- ranking_measure2
        NULL,                -- ranking_measure3
        NULL,                -- result_percentage
        NULL)                -- result_limit
    ) p;

DBMS_SQLTUNE.load_sqlset (
    sqlset_name => 'test_sql_tuning_set_2',
    populate_cursor => l_cursor);
END;
/
```

The contents of an SQL tuning set can be displayed using the SELECT_SQLSET function.

```

SELECT *
FROM   TABLE(DBMS_SQLTUNE.select_sqlset ('test_sql_tuning_set'));
```

References can be added to a set to indicate its usage by a client using the ADD_SQLSET_REFERENCE function. The resulting reference ID can be used to remove it using the REMOVE_SQLSET_REFERENCE procedure.

```

DECLARE
    l_ref_id NUMBER;
BEGIN
    -- Add a reference to a set.
    l_ref_id := DBMS_SQLTUNE.add_sqlset_reference (
        sqlset_name => 'test_sql_tuning_set',
        reference    => 'Used for manual tuning by SQL*Plus.');
```

-- Delete the reference.

```
DBMS_SQLTUNE.remove_sqlset_reference (  
  sqlset_name => 'test_sql_tuning_set',  
  reference_id => l_ref_id);
```

END;

/

The UPDATE_SQLSET procedure is used to update specific string (MODULE and ACTION) and number (PRIORITY and PARSING_SCHEMA_ID) attributes of specific statements within a set.

BEGIN

```
DBMS_SQLTUNE.update_sqlset (  
  sqlset_name   => 'test_sql_tuning_set',  
  sql_id        => '19v5guvsgcd1v',  
  attribute_name => 'ACTION',  
  attribute_value => 'INSERT');
```

END;

/

The contents of a set can be trimmed down or deleted completely using the DELETE_SQLSET procedure.

BEGIN

-- Delete statements with less than 50 executions.

```
DBMS_SQLTUNE.delete_sqlset (  
  sqlset_name => 'test_sql_tuning_set',  
  basic_filter => 'executions < 50');
```

-- Delete all statements.

```
DBMS_SQLTUNE.delete_sqlset (  
  sqlset_name => 'test_sql_tuning_set');
```

END;

/

Tuning sets can be dropped using the DROP_SQLSET procedure.

BEGIN

```
DBMS_SQLTUNE.drop_sqlset (sqlset_name => 'test_sql_tuning_set');  
DBMS_SQLTUNE.drop_sqlset (sqlset_name => 'test_sql_tuning_set_2');
```

END;

/

Useful Views

Useful views related to automatic SQL tuning include:

- DBA_ADVISOR_TASKS
- DBA_ADVISOR_FINDINGS
- DBA_ADVISOR_RECOMMENDATIONS
- DBA_ADVISOR_RATIONALE
- DBA_SQLTUNE_STATISTICS
- DBA_SQLTUNE_BINDS
- DBA_SQLTUNE_PLANS
- DBA_SQLSET
- DBA_SQLSET_BINDS
- DBA_SQLSET_STATEMENTS
- DBA_SQLSET_REFERENCES
- DBA_SQL_PROFILES

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

- V\$SQL
- V\$SQLAREA
- V\$ACTIVE_SESSION_HISTORY

#####195- Trabalhando com Flashback Query, database e version.

– query table data from specific point in time

```
SELECT * FROM matthiash
AS OF TIMESTAMP
TO_TIMESTAMP('2013-05-03 14:00:00', 'YYYY-MM-DD HH24:MI:SS');
```

– query table data from one hour ago

```
SELECT * FROM matthiash
AS OF TIMESTAMP
(SYSTIMESTAMP - INTERVAL '60' MINUTE);
```

– query all versions of one or more records

```
SELECT versions_startscn, versions_starttime,
versions_endscn, versions_endtime,
versions_xid, versions_operation, m_id
FROM matthiash
VERSIONS BETWEEN TIMESTAMP
TO_TIMESTAMP('2013-05-03 13:00:00', 'YYYY-MM-DD HH24:MI:SS')
AND TO_TIMESTAMP('2013-05-03 14:45:00', 'YYYY-MM-DD HH24:MI:SS')
```

– get the user name of the user who modified records

```
GRANT SELECT ANY TRANSACTION TO matthiash;
```

```
SELECT xid, logon_user
FROM flashback_transaction_query
WHERE xid IN (
SELECT versions_xid FROM matthiash VERSIONS BETWEEN TIMESTAMP
TO_TIMESTAMP('2013-05-03 13:00:00', 'YYYY-MM-DD HH24:MI:SS')
AND TO_TIMESTAMP('2013-05-03 14:45:00', 'YYYY-MM-DD HH24:MI:SS')
)
AND xid = HEXTORAW('08000800C6070000');
```

– flashback entire table to specific point in time

```
ALTER TABLE matthiash ENABLE ROW MOVEMENT
```

```
FLASHBACK TABLE matthiash
TO TIMESTAMP TO_TIMESTAMP('2013-05-03 13:00:00', 'YYYY-MM-DD HH24:MI:SS');
```

– creating restore point and flashback table to restore point

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
CREATE RESTORE POINT before_table_update;
```

```
SELECT NAME, SCN, TIME  
FROM V$RESTORE_POINT;
```

```
UPDATE matthiash SET m_id=0;
```

```
FLASHBACK TABLE matthiash.matthiash  
TO RESTORE POINT before_table_update;
```

– recover dropped table from recycle bin (note: does not work for all types of tables)

```
DROP TABLE matthiash;
```

```
SELECT * FROM recyclebin;
```

```
FLASHBACK TABLE matthiash TO BEFORE DROP;
```

```
FLASHBACK TABLE "BIN$29FWNr5ICjbgQ68BEqzYLw==$0" TO BEFORE DROP RENAME TO  
matthiash;
```

– recover entire database to restore point, SCN or point in time

```
CREATE RESTORE POINT before_table_drop GUARANTEE FLASHBACK DATABASE;
```

```
DROP TABLE matthiash PURGE;
```

```
SHUTDOWN IMMEDIATE;  
STARTUP MOUNT;
```

```
FLASHBACK DATABASE TO RESTORE POINT BEFORE_TABLE_DROP;
```

```
FLASHBACK DATABASE TO SCN 46963;
```

```
FLASHBACK DATABASE TO TIME "TO_TIMESTAMP('2013-05-03 15:00:00', 'YYYY-MM-DD  
HH24:MI:SS')";
```

```
ALTER DATABASE OPEN RESETLOGS;
```

#####196- Se apagar datafile com banco aberto.

No Linux com root vc faz:

1- lsof | grep oradata

Acha o processo que está usando o datafile que foi apagado;

2- cd /proc/root/numero_processo

3- pega o cara e dá um CP para local correto

#####197- Criando JOB para DML

```
-- criando JOB
begin
dbms_scheduler.create_job(
job_name => 'test_job',
job_type => 'PLSQL_BLOCK',
job_action => 'update emp set esal=esal*10;',
start_date => SYSDATE,
repeat_interval => 'FREQ=DAILY; INTERVAL=10',
comments => 'Update tabela EMP');
end;
--colocando pra rodar
begin
dbms_scheduler.run_job('TEST_job',TRUE);
end;
```

#####198- Verificando datas da coleta de estatísticas

```
SELECT TRUNC(last_analyzed) DATA, COUNT(1) qtd
FROM dba_tables
WHERE owner = 'SYS'
GROUP BY TRUNC(last_analyzed)
ORDER BY 1
```

--Variação

```
select table_name, last_analyzed from all_tables where table_name like 'BSC_%' order by 2
desc
```

#####199- Trabalhando com ADVISOR TUNING

Primeiramente com o item 66.2 vemos através do relatório do ADDM ou até mesmo pode ser por outra query o SQL_ID de alguma query problemática

--Criando a TASK

```
SET SERVEROUTPUT ON
DECLARE
l_sql_tune_task_id VARCHAR2(100);
BEGIN
l_sql_tune_task_id := DBMS_SQLTUNE.create_tuning_task (
sql_id => '1pm6aq84srup6',
scope => DBMS_SQLTUNE.scope_comprehensive,
time_limit => 4800,
task_name => '1pm6aq84srup6_tuning_task',
description => 'Tuning task for statement 1pm6aq84srup6.');
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
DBMS_OUTPUT.put_line('l_sql_tune_task_id: ' || l_sql_tune_task_id);
END;
/
```

```
--Executando a TASK
EXEC DBMS_SQLTUNE.EXECUTE_TUNING_TASK(task_name => '1pm6aq84srup6_tuning_task');
```

```
--Ver relatório
SET LONG 10000;
SET PAGESIZE 1000
SET LINESIZE 200
SELECT DBMS_SQLTUNE.report_tuning_task('1pm6aq84srup6_tuning_task') AS
recommendations FROM dual;
```

```
--Dropar task
DBMS_SQLTUNE.DROP_TUNING_TASK(task_name => '1pm6aq84srup6_tuning_task');
```

#####200 Extraindo DDL de criação de objetos diversas formas

Usando o pacote DBMS_METADATA no SQL*Plus

-- Verificando algumas das funções que utilizarei para geração dos comandos DDL's
SCOTT> desc dbms_metadata;

FUNCTION GET_DDL RETURNS CLOB

Nome do Argumento	Tipo	In/Out Default?
-------------------	------	-----------------

OBJECT_TYPE	VARCHAR2	IN
NAME	VARCHAR2	IN
SCHEMA	VARCHAR2	IN DEFAULT
VERSION	VARCHAR2	IN DEFAULT
MODEL	VARCHAR2	IN DEFAULT
TRANSFORM	VARCHAR2	IN DEFAULT

FUNCTION GET_DEPENDENT_DDL RETURNS CLOB

Nome do Argumento	Tipo	In/Out Default?
-------------------	------	-----------------

OBJECT_TYPE	VARCHAR2	IN
BASE_OBJECT_NAME	VARCHAR2	IN
BASE_OBJECT_SCHEMA	VARCHAR2	IN DEFAULT
VERSION	VARCHAR2	IN DEFAULT
MODEL	VARCHAR2	IN DEFAULT

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

TRANSFORM	VARCHAR2	IN	DEFAULT
OBJECT_COUNT	NUMBER	IN	DEFAULT

FUNCTION GET_GRANTED_DDL RETURNS CLOB

Nome do Argumento	Tipo	In/Out	Default?
-------------------	------	--------	----------

OBJECT_TYPE	VARCHAR2	IN	
GRANTEE	VARCHAR2	IN	DEFAULT
VERSION	VARCHAR2	IN	DEFAULT
MODEL	VARCHAR2	IN	DEFAULT
TRANSFORM	VARCHAR2	IN	DEFAULT
OBJECT_COUNT	NUMBER	IN	DEFAULT

-- Configurando o ambiente

SCOTT> set linesize 1000

SCOTT> set pagesize 1000

SCOTT> set long 9999999

-- Adicionando um terminador SQL (; ou /) para cada sentença DDL gerada

SCOTT> exec dbms_metadata.set_transform_param(
dbms_metadata.session_transform,'SQLTERMINATOR',true);

Procedimento PL/SQL concluído com sucesso.

-- Suprimindo qualquer informação de atributos de armazenamento de segmentos

SCOTT> exec dbms_metadata.set_transform_param(
dbms_metadata.session_transform,'SEGMENT_ATTRIBUTES',false);

Procedimento PL/SQL concluído com sucesso.

-- Gerando DDL para a tabela T1

SCOTT> select dbms_metadata.get_ddl('TABLE','T1') "DDL TABLE" from dual;

DDL TABLE

```
CREATE TABLE "SCOTT"."T1"
( "ID" NUMBER,
"NAME" VARCHAR2(100),
CONSTRAINT "PK_T1" PRIMARY KEY ("ID") ENABLE
);
```

-- Gerando DDL para a view VIEW_T1_NOME

SCOTT> select dbms_metadata.get_ddl('VIEW','VIEW_T1_NOME') "DDL VIEW" from dual;

DDL VIEW

```
CREATE OR REPLACE FORCE VIEW "SCOTT"."VIEW_T1_NOME" ("NOME") AS
SELECT NOME FROM T1;
```

-- Gerando DDL apenas para a chave estrangeira definida na tabela T2

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle


```
SCOTT> select dbms_metadata.get_dependent_ddl('REF_CONSTRAINT','T2') "DDL FK" from dual;
```

DDL FK

```
ALTER TABLE "SCOTT"."T2" ADD CONSTRAINT "FK_T2_T1" FOREIGN KEY ("ID")
REFERENCES "SCOTT"."T1" ("ID") ENABLE;
```

-- Gerando DDL para restrições do tipo (PK/UK/CHK) existentes definidas em T1

```
SCOTT> select dbms_metadata.get_dependent_ddl('CONSTRAINT','T1') "DDL PK/UK/CHK"
from dual;
```

DDL PK/UK/CHK

```
ALTER TABLE "SCOTT"."T1" ADD CONSTRAINT "PK_T1" PRIMARY KEY ("ID") ENABLE;
```

-- Gerando o comando DDL para qualquer gatilho existente para a tabela T1

```
SCOTT> select dbms_metadata.get_dependent_ddl('TRIGGER','T1') "DDL TRIGGER" from dual;
```

DDL TRIGGER

```
CREATE OR REPLACE TRIGGER "SCOTT"."TRG_BI_T1"
BEFORE INSERT ON T1
FOR EACH ROW
BEGIN
:NEW.ID := DBMS_RANDOM.RANDOM;
END;
/
ALTER TRIGGER "SCOTT"."TRG_BI_T1" ENABLE;
```

-- Gerando comando DDL para qualquer índice existente para a tabela T1

```
SCOTT> select dbms_metadata.get_dependent_ddl('INDEX','T1') "DDL ÍNDICE" from dual;
```

DDL ÍNDICE

```
CREATE INDEX "SCOTT"."I_T1_NOME" ON "SCOTT"."T1" ("NOME");
CREATE UNIQUE INDEX "SCOTT"."PK_T1" ON "SCOTT"."T1" ("ID");
```

-- Gerando DDL para a view materializada

```
SCOTT> select dbms_metadata.get_ddl('MATERIALIZED_VIEW','MVIEW_T1') "DDL MVIEW"
from dual;
```

DDL MVIEW

```
CREATE MATERIALIZED VIEW "SCOTT"."MVIEW_T1"
ORGANIZATION HEAP PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS
LOGGING
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
TABLESPACE "USERS"
```

```
BUILD IMMEDIATE
USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
TABLESPACE "USERS"
REFRESH FORCE ON DEMAND START WITH sysdate+0 NEXT sysdate + 1/1440
WITH PRIMARY KEY USING DEFAULT LOCAL ROLLBACK SEGMENT
DISABLE QUERY REWRITE
AS SELECT "T1"."ID" "ID", "T1"."NOME" "NOME" FROM "T1" "T1";
```

-- Exemplo para geração de comandos DDL's para todas as tabelas existentes

```
SCOTT> select dbms_metadata.get_ddl(object_type, object_name) ddl
2 from user_objects
3 where object_type = 'TABLE';
```

DDL

```
-----
CREATE TABLE "SCOTT"."MVIEW_T1"
( "ID" NUMBER,
"NOME" VARCHAR2(100),
CONSTRAINT "PK_T11" PRIMARY KEY ("ID") ENABLE
);
```

```
CREATE TABLE "SCOTT"."T1"
( "ID" NUMBER,
"NOME" VARCHAR2(100),
CONSTRAINT "PK_T1" PRIMARY KEY ("ID") ENABLE
);
```

```
CREATE TABLE "SCOTT"."T2"
( "ID" NUMBER,
CONSTRAINT "FK_T2_T1" FOREIGN KEY ("ID")
REFERENCES "SCOTT"."T1" ("ID") ENABLE
);
```

Apenas como demonstração, irei gerar abaixo os comandos DDL's necessários para concessão de privilégios, criação do usuário, role e tablespace:

```
SCOTT> connect / as sysdba
Conectado.
```

```
SYS> set linesize 1000
SYS> set pagesize 1000
SYS> set long 9999999
```

```
SYS> exec dbms_metadata.set_transform_param(
dbms_metadata.session_transform, 'SQLTERMINATOR', true);
```

Procedimento PL/SQL concluído com sucesso.

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
SYS> exec dbms_metadata.set_transform_param(
    dbms_metadata.session_transform,'SEGMENT_ATTRIBUTES',false);
```

Procedimento PL/SQL concluído com sucesso.

-- Gerando DDL para criação da tablespace USERS

```
SYS> select dbms_metadata.get_ddl('TABLESPACE','USERS') DDL from dual;
```

DDL

```
-----
CREATE TABLESPACE "USERS" DATAFILE
'C:\ORACLEXE\ORADATA\XE\USERS.DBF' SIZE 104857600
AUTOEXTEND ON NEXT 10485760 MAXSIZE 5120M
LOGGING ONLINE PERMANENT BLOCKSIZE 8192
EXTENT MANAGEMENT LOCAL AUTOALLOCATE SEGMENT SPACE MANAGEMENT AUTO;
ALTER DATABASE DATAFILE
'C:\ORACLEXE\ORADATA\XE\USERS.DBF' RESIZE 3145728000;
```

-- Gerando DDL para criação do usuário SCOTT

```
SYS> select dbms_metadata.get_ddl('USER','SCOTT') DDL from dual;
```

DDL

```
-----
CREATE USER "SCOTT" IDENTIFIED BY VALUES 'F894844C34402B67'
DEFAULT TABLESPACE "USERS"
TEMPORARY TABLESPACE "TEMP";
```

-- Gerando DDL de privilégios de sistema concedidos ao usuário SCOTT

```
SYS> select dbms_metadata.get_granted_ddl('SYSTEM_GRANT','SCOTT') DDL from dual;
```

DDL

```
-----
GRANT CREATE MATERIALIZED VIEW TO "SCOTT";
GRANT CREATE VIEW TO "SCOTT";
GRANT UNLIMITED TABLESPACE TO "SCOTT";
```

-- Gerando DDL de roles concedidas ao usuário SCOTT

```
SYS> select dbms_metadata.get_granted_ddl('ROLE_GRANT','SCOTT') DDL from dual;
```

DDL

```
-----
GRANT "CONNECT" TO "SCOTT";
GRANT "RESOURCE" TO "SCOTT";
```

-- Gerando DDL de privilégios de objetos que foram concedidos pelo usuário

-- SCOTT ao usuário ADAM

```
SYS> select dbms_metadata.get_granted_ddl('OBJECT_GRANT','ADAM') DDL from dual;
```

DDL

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
-----  
GRANT SELECT ON "SCOTT"."T1" TO "ADAM";  
GRANT UPDATE ON "SCOTT"."T1" TO "ADAM";
```

```
-- Gerando DDL de criação da role CONNECT  
SYS> select dbms_metadata.get_ddl('ROLE','CONNECT') DDL from dual;
```

DDL

```
-----  
CREATE ROLE "CONNECT";
```

Usando os utilitários exp/imp

```
C:\exp system/senha@SATI file=c:\extrai_ddl.dmp grants=n statistics=none rows=n full=y  
C:\>imp system/senha@SATI file=c:\extrai_ddl.dmp indexfile=c:\ddl.sql full=y  
-- exportando as tabelas do schema SCOTT  
C:\exp scott/tiger file=c:\scott grants=n statistics=none rows=n
```

Export: Release 10.2.0.1.0 - Production on Sex Jul 18 12:58:03 2008

Copyright (c) 1982, 2005, Oracle. All rights reserved.

Conectado a: Oracle Database 10g Express Edition Release 10.2.0.1.0 - Production
Exportação executada no conjunto de caracteres de WE8PC850 e no conjunto de caracteres de AL16UTF16 NCHAR o servidor usa WE8MSWIN1252 conjunto de caracteres (conversão de conjunto de caracteres possível)

OBS: dados (linhas) da tabela não serão exportados

OBS: concessões em tabelas/views/seqüências/funções não serão exportadas

- . exportando objetos e ações procedurais anteriores ao esquema
- . exportando os nomes da biblioteca de função externa para usuário SCOTT
- . exportando sinônimos do tipo PÚBLICO
- . exportando sinônimos do tipo privado
- . exportando definições de tipos de objeto para usuário SCOTT

Sobre exportar objetos de SCOTT ...

- . exportando vínculos de banco de dados
- . exportando números de seqüência
- . exportando definições de cluster
- . sobre exportar tabelas de SCOTT ... via Caminho Convencional ...
 - . . exportando tabela MVIEW_T1
 - . . exportando tabela T1
 - . . exportando tabela T2
- . exportando sinônimos
- . exportando views
- . exportando procedimentos armazenados
- . exportando operadores
- . exportando restrições referenciais de integridade
- . exportando gatilhos
- . exportando tipos de índices
- . exportando índices funcionais, extensíveis e de bitmap

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

- . exportando ações contabilizáveis
- . exportando views materializadas
- . exportando logs de snapshot
- . exportando filas de serviço
- . exportando filhos e grupos de renovação
- . exportando dimensões
- . exportando objetos e ações procedurais posteriores ao esquema
- . exportando estatística

Exportação encerrada com sucesso, sem advertências.

-- Gerando os comandos DDL's para arquivo texto
C:\>imp scott/tiger file=c:\scott indexfile=c:\ddl.sql

Import: Release 10.2.0.1.0 - Production on Sex Jul 18 12:59:25 2008

Copyright (c) 1982, 2005, Oracle. All rights reserved.

Conectado a: Oracle Database 10g Express Edition Release 10.2.0.1.0 - Production

Arquivo de exportação criado por EXPORT:V10.02.01 via caminho convencional
importação realizada nos conjuntos de caracteres WE8PC850 e NCHAR AL16UTF16
o servidor de importação usa o conjunto de caracteres WE8MSWIN1252 (conversão
de charset possível)

. . saltando a tabela "MVIEW_T1"
. . saltando a tabela "T1"
. . saltando a tabela "T2"

Importação encerrada com sucesso, sem advertências.

-- Verificando os comandos DDL's gerados
C:\>type C:\ddl.sql

```
REM CREATE TABLE "SCOTT"."MVIEW_T1" ("ID" NUMBER, "NOME" VARCHAR2(100))  
REM PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 STORAGE(INITIAL 65536  
REM FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT) TABLESPACE "USERS"  
REM LOGGING NOCOMPRESS ;
```

CONNECT SCOTT;

```
CREATE UNIQUE INDEX "SCOTT"."PK_T11" ON "MVIEW_T1" ("ID" ) PCTFREE 10  
INITRANS 2 MAXTRANS 255 STORAGE(INITIAL 65536 FREELISTS 1 FREELIST GROUPS  
1 BUFFER_POOL DEFAULT) TABLESPACE "USERS" LOGGING;
```

```
REM ALTER TABLE "SCOTT"."MVIEW_T1" ADD CONSTRAINT "PK_T11" PRIMARY KEY  
REM ("ID") USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 STORAGE(INITIAL  
REM 65536 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT) TABLESPACE  
REM "USERS" LOGGING ENABLE;
```

```
REM CREATE TABLE "SCOTT"."T1" ("ID" NUMBER, "NOME" VARCHAR2(100)) PCTFREE  
REM 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 STORAGE(INITIAL 65536 FREELISTS
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
REM 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT) TABLESPACE "USERS" LOGGING
REM NOCOMPRESS;

CREATE UNIQUE INDEX "SCOTT"."PK_T1" ON "T1" ("ID" ) PCTFREE 10 INITRANS 2
MAXTRANS 255 STORAGE(INITIAL 65536 FREELISTS 1 FREELIST GROUPS 1
BUFFER_POOL DEFAULT) TABLESPACE "USERS" LOGGING;

CREATE INDEX "SCOTT"."I_T1_NOME" ON "T1" ("NOME" ) PCTFREE 10 INITRANS 2
MAXTRANS 255 STORAGE(INITIAL 65536 FREELISTS 1 FREELIST GROUPS 1
BUFFER_POOL DEFAULT) TABLESPACE "USERS" LOGGING;

REM ALTER TABLE "SCOTT"."T1" ADD CONSTRAINT "PK_T1" PRIMARY KEY ("ID")
REM USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 STORAGE(INITIAL 65536
REM FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT) TABLESPACE "USERS"
REM LOGGING ENABLE;

REM CREATE TABLE "SCOTT"."T2" ("ID" NUMBER) PCTFREE 10 PCTUSED 40
REM INITRANS 1 MAXTRANS 255 STORAGE(INITIAL 65536 FREELISTS 1 FREELIST
REM GROUPS 1 BUFFER_POOL DEFAULT) TABLESPACE "USERS" LOGGING NOCOMPRESS;

REM ALTER TABLE "SCOTT"."T2" ADD CONSTRAINT "FK_T2_T1" FOREIGN KEY ("ID")
REM REFERENCES "T1" ("ID") ENABLE NOVALIDATE ;

REM ALTER TABLE "SCOTT"."T2" ENABLE CONSTRAINT "FK_T2_T1";
```

Usando os utilitários expdp/impdp (Datapump 10g)

-- Realizando a exportação apenas dos metadados

C:\>expdp scott/tiger directory=data_pump_dir dumpfile=scott content=metadata_only

Export: Release 10.2.0.1.0 - Production on Sexta-Feira, 18 Julho, 2008 13:32:50

Copyright (c) 2003, 2005, Oracle. All rights reserved.

Conectado a: Oracle Database 10g Express Edition Release 10.2.0.1.0 - Production

Iniciando "SCOTT"."SYS_EXPORT_SCHEMA_01":

scott/***** directory=data_pump_dir dumpfile=scott content=metadata_only

Processando o tipo de objeto SCHEMA_EXPORT/PRE_SCHEMA/PROACT_SCHEMA

Processando o tipo de objeto SCHEMA_EXPORT/TABLE/TABLE

Processando o tipo de objeto

SCHEMA_EXPORT/TABLE/GRANT/OWNER_GRANT/OBJECT_GRANT

Processando o tipo de objeto SCHEMA_EXPORT/TABLE/INDEX/INDEX

Processando o tipo de objeto SCHEMA_EXPORT/TABLE/CONSTRAINT/CONSTRAINT

Processando o tipo de objeto SCHEMA_EXPORT/TABLE/INDEX/STATISTICS/INDEX_STATISTICS

Processando o tipo de objeto SCHEMA_EXPORT/TABLE/COMMENT

Processando o tipo de objeto SCHEMA_EXPORT/VIEW/VIEW

Processando o tipo de objeto SCHEMA_EXPORT/TABLE/CONSTRAINT/REF_CONSTRAINT

Processando o tipo de objeto SCHEMA_EXPORT/TABLE/TRIGGER

Processando o tipo de objeto SCHEMA_EXPORT/MATERIALIZED_VIEW

Processando o tipo de objeto SCHEMA_EXPORT/JOB

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

Tabela-mestre "SCOTT"."SYS_EXPORT_SCHEMA_01" carregada/descarregada com sucesso

*

Conjunto de arquivos de dump para SCOTT.SYS_EXPORT_SCHEMA_01 é:
C:\ORACLE\APP\ORACLE\ADMIN\XE\DPDUMP\SCOTT.DMP
O job "SCOTT"."SYS_EXPORT_SCHEMA_01" foi concluído com sucesso em 13:33:06

-- Gerando os comandos DDL's para arquivo texto
C:\>impdp scott/tiger directory=data_pump_dir dumpfile=scott sqlfile=ddl.sql

Import: Release 10.2.0.1.0 - Production on Sexta-Feira, 18 Julho, 2008 13:34:24

Copyright (c) 2003, 2005, Oracle. All rights reserved.

Conectado a: Oracle Database 10g Express Edition Release 10.2.0.1.0 - Production
ORA-39154: Objetos de esquemas estrangeiros foram removidos da importação
Tabela-mestre "SCOTT"."SYS_SQL_FILE_FULL_01" carregada/descarregada com sucesso
Iniciando "SCOTT"."SYS_SQL_FILE_FULL_01":
scott/***** directory=data_pump_dir dumpfile=scott sqlfile=ddl.sql
Processando o tipo de objeto SCHEMA_EXPORT/PRE_SCHEMA/PROCT_SCHEMA
Processando o tipo de objeto SCHEMA_EXPORT/TABLE/TABLE
Processando o tipo de objeto
SCHEMA_EXPORT/TABLE/GRANT/OWNER_GRANT/OBJECT_GRANT
Processando o tipo de objeto SCHEMA_EXPORT/TABLE/INDEX/INDEX
Processando o tipo de objeto SCHEMA_EXPORT/TABLE/CONSTRAINT/CONSTRAINT
Processando o tipo de objeto SCHEMA_EXPORT/TABLE/COMMENT
Processando o tipo de objeto SCHEMA_EXPORT/VIEW/VIEW
Processando o tipo de objeto SCHEMA_EXPORT/TABLE/CONSTRAINT/REF_CONSTRAINT
Processando o tipo de objeto SCHEMA_EXPORT/TABLE/TRIGGER
Processando o tipo de objeto SCHEMA_EXPORT/MATERIALIZED_VIEW
O job "SCOTT"."SYS_SQL_FILE_FULL_01" foi concluído com sucesso em 13:34:28

-- Verificando os comandos DDL's gerados
C:\>type ddl.sql
-- CONNECT SCOTT
-- SCHEMA_EXPORT/PRE_SCHEMA/PROCT_SCHEMA
BEGIN
sys.dbms_logrep_imp.instantiate_schema(schema_name=>SYS_CONTEXT
('USERENV','CURRENT_SCHEMA'),
export_db_name=>'XE', inst_scn=>'372450');
COMMIT;
END;
/

-- SCHEMA_EXPORT/TABLE/TABLE
CREATE TABLE "SCOTT"."T1"
("ID" NUMBER,
"NOME" VARCHAR2(100)
) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade - RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
TABLESPACE "USERS";

CREATE TABLE "SCOTT"."T2"
( "ID" NUMBER
) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
TABLESPACE "USERS";

CREATE TABLE "SCOTT"."MVIEW_T1"
( "ID" NUMBER,
"NOME" VARCHAR2(100)
) PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255 NOCOMPRESS LOGGING
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
TABLESPACE "USERS";

-- SCHEMA_EXPORT/TABLE/GRANT/OWNER_GRANT/OBJECT_GRANT
GRANT SELECT ON "SCOTT"."T1" TO "ADAM";

GRANT UPDATE ON "SCOTT"."T1" TO "ADAM";

-- SCHEMA_EXPORT/TABLE/INDEX/INDEX
CREATE UNIQUE INDEX "SCOTT"."PK_T1" ON "SCOTT"."T1" ("ID")
PCTFREE 10 INITRANS 2 MAXTRANS 255
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
TABLESPACE "USERS" PARALLEL 1;

ALTER INDEX "SCOTT"."PK_T1" NOPARALLEL;

CREATE INDEX "SCOTT"."I_T1_NOME" ON "SCOTT"."T1" ("NOME")
PCTFREE 10 INITRANS 2 MAXTRANS 255
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
TABLESPACE "USERS" PARALLEL 1;

ALTER INDEX "SCOTT"."I_T1_NOME" NOPARALLEL;

CREATE UNIQUE INDEX "SCOTT"."PK_T11" ON "SCOTT"."MVIEW_T1" ("ID")
PCTFREE 10 INITRANS 2 MAXTRANS 255
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
TABLESPACE "USERS" PARALLEL 1;

ALTER INDEX "SCOTT"."PK_T11" NOPARALLEL;

-- SCHEMA_EXPORT/TABLE/CONSTRAINT/CONSTRAINT
ALTER TABLE "SCOTT"."T1" ADD CONSTRAINT "PK_T1" PRIMARY KEY ("ID")
```



```
USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
TABLESPACE "USERS" ENABLE;

ALTER TABLE "SCOTT"."MVIEW_T1" ADD CONSTRAINT "PK_T11" PRIMARY KEY ("ID")
USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
TABLESPACE "USERS" ENABLE;

-- SCHEMA_EXPORT/TABLE/COMMENT
COMMENT ON MATERIALIZED VIEW "SCOTT"."MVIEW_T1" IS
'snapshot table for snapshot SCOTT.MVIEW_T1';

-- SCHEMA_EXPORT/VIEW/VIEW
CREATE FORCE VIEW "SCOTT"."VIEW_T1_NOME" ("NOME") AS
SELECT NOME FROM T1;

-- SCHEMA_EXPORT/TABLE/CONSTRAINT/REF_CONSTRAINT
ALTER TABLE "SCOTT"."T2" ADD CONSTRAINT "FK_T2_T1" FOREIGN KEY ("ID")
REFERENCES "SCOTT"."T1" ("ID") ENABLE;

-- SCHEMA_EXPORT/TABLE/TRIGGER
CREATE TRIGGER "SCOTT"."TRG_BI_T1"
BEFORE INSERT ON T1
FOR EACH ROW
BEGIN
:NEW.ID := DBMS_RANDOM.RANDOM;
END;
/

ALTER TRIGGER "SCOTT"."TRG_BI_T1" ENABLE;

ALTER TRIGGER "SCOTT"."TRG_BI_T1"
COMPILE
PLSQL_OPTIMIZE_LEVEL = 2
PLSQL_CODE_TYPE= INTERPRETED;

-- SCHEMA_EXPORT/MATERIALIZED_VIEW
CREATE MATERIALIZED VIEW "SCOTT"."MVIEW_T1" USING
("MVIEW_T1", (8, 'XE', 1, 0, 0, "SCOTT", "T1", '2008-07-18 13:32:43', 0, 14796,
'1950-01-01 12:00:00', "", 0, 372263, 0, NULL, (1, "ID", "ID", 0, 321, 0)),
2097249, 8, ('1950-01-01 12:00:00', 4, 0, 0, 372263, 0, 0, 2, NULL, NULL))
REFRESH FORCE WITH PRIMARY KEY AS
SELECT "T1"."ID" "ID", "T1"."NOME" "NOME" FROM "T1" "T1";

ALTER MATERIALIZED VIEW "SCOTT"."MVIEW_T1" COMPILE;
```

#####201- Setando o log do listener

Onde está o log do listener?

\$ORACLE_HOME/network/log/listener.log

Para limpá-lo, primeiro desabilitamos a "alimentação" do log:

- Entra no prompt do listener:

lsnrctl

- Seta o status do log como OFF:

LSNRCTL> set log_status off

Então podemos apagar o arquivo ou se necessário copiá-lo para outro disco/partição e posteriormente habilitamos novamente o log que, caso tenha sido excluído o arquivo, um novo será gerado:

- Entra no prompt do listener:

lsnrctl

- Seta o status do log como ON:

LSNRCTL> set log_status on

#####202- Criando tablespace e trabalhando com ele

--Criando

CREATE TABLESPACE nome_tablespace LOGGING DATAFILE

'/u01/app/oracle/oradata/TSH1/nome_tablespace.dbf'

SIZE 100m

AUTOEXTEND ON NEXT 100m

EXTENT MANAGEMENT LOCAL;

-- Adicionando datafile

alter tablespace nome_tablespace add datafile

'/u01/app/oracle/oradata/TSH1/nome_tablespace_2.dbf' size 2G;

-- Alterando o datafile para autoextend on (ou off)

alter database datafile

'/u01/app/oracle/oradata/TSH1/nome_tablespace_2.dbf'

autoextend on

maxsize unlimited;

#####203- Tornar usuário como usuário do Enterprise Manager

execute MGMT_USER.MAKE_EM_USER('USER_NAME');

#####204- Comandos úteis de cluster

Comandos para execução manual

ver status (logado como root ou grid)
/u01/app/11.2.0/grid/bin/crsctl stat res -t

Subir tudo (logado como root)
/u01/app/11.2.0/grid/bin/crsctl start cluster -all

Subir específico (logado como root)
/u01/app/11.2.0/grid/bin/crsctl start cluster -n roma0X

Comando para baixar manual (logado como root)
/u01/app/11.2.0/grid/bin/crsctl stop cluster -all

Baixar específico (logado como root)
/u01/app/11.2.0/grid/bin/crsctl stop cluster -n roma0X

Startar crs (logado como root) starta no node de onde é executado
/u01/app/11.2.0/grid/bin/crsctl start crs

Dica:
Caso queira verificar os logs do crs eles estarão em
/u01/app/11.2.0/grid/log/roma0X

#####205- Verificar versão do SO Linux
cat /etc/os-release

#####206- acertar backspace no sqlplus

Solution:
Run this command from your sqlplus "!stty erase [BACKSPACE]"

Note: Don't type "BACKSPACE" but press backspace key instead.

#####207- Estimando o tamanho de uma Materialized View

```
SQL> set serveroutput on;
SQL>
SQL> DECLARE
2 no_of_rows NUMBER;
3 mv_size NUMBER;
4 BEGIN
5 DBMS_MVIEW.estimate_mview_size('MV1',
6 'SELECT c.cust_id, SUM(amount_sold) AS dollar_sales,
7 COUNT(amount_sold) AS cnt_dollars, COUNT(*) AS cnt
8 FROM sales s, customers c
9 WHERE s.cust_id= c.cust_id
10 GROUP BY c.cust_id' , no_of_rows, mv_size);
11 DBMS_OUTPUT.put_line ( '');
12 DBMS_OUTPUT.put_line ( 'Complete MV');
```

Proibida distribuição sem citar a fonte

Raul Francisco C F Andrade – RAULdba

Consultor Oracle nas versões 9i, 10 e 11g e 12c.

Instrutor Oracle

```
13 DBMS_OUTPUT.put_line ( 'No of Rows: ' || no_of_rows );
14 DBMS_OUTPUT.put_line ( 'MV Size: ' || mv_size);
15 DBMS_OUTPUT.put_line ( '');
16 END;
17 /
```

Complete MV
No of Rows: 7059
MV Size: 621192
PL/SQL procedure successfully completed.