


I'm not robot  reCAPTCHA

[Continue](#)

## How to calculate db\_cache\_size 12c

All operations in the Oracle database buffer cache (select DML, DDL) are not performed directly from the physical display, but in a cache area called buffer cache. If the line is changed, the change in this area will first be made. If you want to access the line, the first Oracle will look at whether the information is here. If the information is here, the data is sent to the user or changed the user request function. In this case, the Oracle calls this event HIT. If the information is not in this area (buffer cache) the information is read from the disk and placed in the memory zone (buffer cache). In this case, the Oracle calls this event MISS. Once changed and written to disk or sent to the user, the data will be stored in this area for further use. If new data is required to be retrieved from the buffer cache and there is no room, the oldest data will be replaced by the latest. A well-optimized database must not have too many reloads of data in the buffer cache. To reload the data, reload the data to calculate the buffer cache result ratio. This ratio can be calculated for the entire instance (since the database started):  
SQL> SELECT (P1.VALUE +P2.VALUE) / (P1.VALUE +P2.VALUE)\*100 BUFFER HITS RATIO v\$sysstat P1, v\$sysstat P2, v\$sysstat P3 2 WHERE P1.NAME=LOWER (DB BLOCK GETS) 3 and P2.NAME=LOWER (CONSISTENT GETS) 4 and P3.NAME=LOWER (PHYSICAL READS);  
BUFFER REFERENCE ----- 96.5158211The ratio should be over 90%, if not an increase in the shared\_pool\_size cache-hit ratio going down to 90% then.\* Oracle 8 and earlier: increase reset parameter DB\_BLOCK\_BUFFERS.\* Oracle 9 from: increase initialization parameter DB\_CACHE\_SIZE. If the buffer cache is well set up and no complete table checks are running in the database, this ratio must be up to 90%.  
SQL> change the system set db\_cache\_size = on ; The system has changed. SQL> select NAME,SIZE\_FOR\_ESTIMATE BC size(M), SIZE\_FACTOR,ESTD\_PHYSICAL\_READ\_FACTOR es\_phy\_r\_f,ESTD\_PHYSICAL\_READS es\_phy\_r v\$db\_cache\_size; NAME BC SIZE(M) SIZE\_FACTOR ES\_PHY\_R\_F ES\_PHY\_R -----DEFAULT 28 .0909 2.1867 27531DEFAULT 56 .1818 1.3393 16862 DEFAULT 84 .2727 1.1498 14476DEFAULT 112 .3636 1.0095 12709DEFAULT 140 .4545 1.12590DEFAULT 168 .5455 1.12590DEFAULT 196 .6364 1.12590DEFAULT 224 .7273 1.12590DEFAULT 252 .8182 1.12590DEFAULT 280 .9091 1.1259 0 DEFAULT 308 1 1 12590: DEFAULT 336 1.0909 1 12590DEFAULT 364 1.1818 1 12590default 392 1.2727 1 12590DEFAULT 420 1.3636 1 12590DEFAULT 448 1.4545 1 12590DEFAULT 476 1.5455 1 12590DEFAULT 504 1.6364 1 12590DEFAULT 532 1.7273 1 12590DEFAULT 560 1.8182 1 1259020 lines selected. Acc above.-CASE 1: IF BUFFER CACHE is set to 84 MB, then the physical reads are 14476.CASE 2: IF THE BUFFER cache is set to 112 MB, then the physical reads are 12709.CASE 3: IF the buffer cache is set to 140 MB, then the physical reads are 12590.If BC is greater than 140 Mb does not benefit database performance, so having a 140 Mb buffer cache could be a good solution for this system. The buffer cache size is managed DB\_CACHE\_SIZE the initialization parameter. Resize buffer cache: Now check the maximum size of that SGA that may grow. Compare the size of the SGA parameter with the shared pool, buffer cache, and re-log sizes together with the java pool. SQL> show parameter sga\_max\_size;select amount(value) if SGA size from V\$parameter; name is ('Shared\_pool\_size', 'db\_cache\_size', 'log\_buffer', 'java\_pool\_size'); If the size is very close, reduce the size of the other to one size. Check shared\_pool and buffer\_cache size separately. SQL> alter system set db\_cache\_size = 100M; SQL> alter system set shared\_pool\_size = 200M; Example - In my database, bc can be set at 140 MB to ensure satisfactory performance. CHANGE SYSTEM SET db\_cache\_size=OFF;-- Then reduce the buffer cache size to 140 Megabytes.ALTER system set db\_cache\_size = 140M-- Increase the shared party size equal to the amount u reduce buffer cache size i.e. equal buffer cache size - 140M E assume 300MALTER SYSTEM SET THE 300M db\_cache\_size (formerly db\_block\_buffers) adjusts the size of the RAM data buffer cache area. Oracle has a v\$db\_cache\_size utility to determine marginal benefits (in terms of reducing disk counts) by adding more data to the cache. In the gruesome Oracle9i we also have a separate database cache size parameters for several block sizes, db\_nk\_cache\_size. Some experts say that Oracle indices should be removed from db\_cache\_size and moved db\_32k\_cache\_size to the region to build cleaner tree structures and reduce the I/O index range of scans. It's important to remember that there are downsides that have super-large db\_cache\_size. Although direct access to the data is made hashing, there are times when the database must examine all blocks of the RAM cache. Systems with high repeat recognizes - If the program issues a truncated table, uses temporary tables or runs a large data cleanup, Oracle must sweep all db\_cache\_size remove dirty blocks. This can cause excessive overhead system db\_cache\_size greater than 10 gigabytes. High Update Systems - The database compiler (DBWR) process must sweep all blocks db\_cache\_size asynchronous write. Having a huge db\_cache\_size can cause excessive work by a database writer. RAC systems - Systems that use the RAC of Oracle9 can experience a big cross-call when you use a large db\_cache\_size in multiple copies of the RAC. This inter-instance ping can cause excessive overheads, and therefore the RAC DBA tries to allocate RAC cases to access certain areas of the database. Ion tool plots historical use of your db\_cache\_size: Also see my notes db\_cache\_size: Buffer States: Confirmed: Currently Used. Clean: This means that the buffer is now unapproved and the candidate for immediate expiry when the current (data block) have changed and must be emptied to disk DBWn before it can get out of age. Three buffer formations: database buffer cache parameter: DB\_CACHE\_SIZE (block size is the size of declare DB\_BLOCK\_SIZE) DB\_KEEP\_CACHE\_SIZE DB\_2K\_CACHE\_SIZE DB\_4K\_CACHE\_SIZE DB\_8K\_CACHE\_SIZE DB\_16K\_CACHE\_SIZE DB\_32K\_CACHE\_SIZE DB\_RACycle\_CACHE\_SIZE DB\_BLOCK\_SIZE (Default block size that cannot be changed later anyway) SQL> SHOW PARAMETER DB\_BLOCK\_SIZE; NAME TYPE ----- DB\_BLOCK\_SIZE 8192 SQL> CREATE A TABLESPACE TEST DATA FILE D:\oracle\oradata\dba50\test01.dbf SIZE 200M BLOCKSIZE 2K; CREATE TABLESPACE TEST \* ERROR on line 1: ORA-29339: Tablespace block size 2048 does not match configured block sizes in SQL> change system set db\_2k\_cache\_size = 2M; Change system set db\_2k\_cache\_size = 2M \* ERROR line 1: ORA-02097: The parameter cannot be changed because the specified value does not match the size of the ORA-00384: Insufficient memory to grow cache NOTE: Because the SGA MAX size already does not be filled in so it is necessary to increase the SGA\_MAX size or reduce the cache size/redolog buffer cache size/cache size of the database. SQL> display parameter db\_cache\_size; NAME TYPE VALUE ----- db\_cache\_size a large integer of 25165824 in SQL> Change system set db\_cache\_size = 20M; The system has changed. SQL> change system set db\_2k\_cache\_size = 2M; The system has changed. SQL> CREATE A TABLESPACE TEST DATA FILE D:\oracle\oradata\dba50\test01.dbf SIZE 200M BLOCKSIZE 2K; Teblespace created. Show cache result: SQL> select (1-(physical.value-direct.value-lob.value)/logical.value)\*100 Buffer hit v\$sysstat self, v\$sysstat direct, v\$sysstat lob, v\$sysstat logical, where physical.name='physical reads' and direct.name='physical reads direct' and lob.name='physical reads lob' and logical.name='logical session reads'; Buffer hit ----- 97.0100575 buffer cache performance indicators: SQL> select name, v\$sysstat where name = free buffer checked, NAME VALUE ----- free buffer checked 0 This statistic is the number of buffers between the number of buffers found in the free buffer. Buffers are skipped because they are dirty or pinned. SQL> select the event, total\_waits v\$ system\_event where the event ('buffer busy waiting'); EVENT TOTAL\_WAITS ----- buffer busy 24 Buffer Busy Waits indicates that the buffer cache that multiple processes are trying to access simultaneously in the buffer cache. Tuning Steps: Make it larger Use multiple buffer farm cache table memory to bypass the buffer cache to use the registry to correctly repair free buffer waits 1. Make it bigger: SQL> show parameter db\_cache\_size; NAME TYPE VALUE ----- db\_cache\_size a large integer 20971520 SQL> display parameter sga\_max\_size; NAME TYPE VALUE ----- sga\_max\_size a large integer of 135338868 in SQL> Change system sga\_max\_size = 160M scope = spfile; The system has changed. SQL> Startup Force; Oracle instance started. Total system global area 168893060 bytes fixed size 453252 bytes variable size 134217728 bytes database buffers of 335554432 bytes re-buffers 667648 bytes of the database connected. Database open. SQL> change system set db\_cache\_size = 40M; The system has changed. SQL> display parameter db\_cache\_size; The name type ----- db\_cache\_size a large integer 41943040 You can take advice before db\_cache\_size increase. Db\_cache\_size = On; The system has changed. SQL> select the name,SIZE\_FOR\_ESTIMATE,ESTD\_PHYSICAL\_READS v\$db\_cache\_size; NAME TYPE VALUE ----- db\_cache\_size string OFF SQL> change system set db\_cache\_size = ready; The system has changed. SQL> change system set db\_2k\_cache\_size = 2M; The system has changed. SQL> change system set db\_recycle\_cache\_size = 2M; The system has changed. SQL> change the table scott.dept storage(buffer\_pool) The table has been changed. SQL> Create table d(id-number) storage (buffer\_pool stored); Table created. SQL> create table dd(id number) storage (buffer\_pool recycle); Table created. SQL> select the owner segment\_type.segment\_name buffer\_pool dba\_segments where buffer\_pool='KEEP'; OWNER SEGMENT\_TYPE SEGMENT\_NAME BUFFER\_POOL ----- SCOTT TABLE DEPT HOLD SYS TABLE D HOLD SQL> SELECT dba\_segments the buffer\_pool the data to =RECYCLE'; OWNER SEGMENT\_TYPE SEGMENT\_NAME BUFFER\_POOL ----- SYS TABLE DD RECYCLE Multi-folder SQL performance ratio calculation &gt; select name,1-(physical\_reads/(db\_block\_gets+consistent\_gets)) HIT\_RATIO v\$buffer\_pool\_statistics WHERE db\_block\_gets+consistent\_gets>0; NAME HIT\_RATIO ----- DEFAULT .988217737 RECYCLE .503866235 KEEP .983520845 3 Cache Table Memory SQL> Change table scott.emp cache; The table has been changed. SQL> create a table in the xyz(ID number) cache; Table created. If only FTS typically occurs, this table result saves the LRU but after this query always this table results tray lru. SQL> change the table scott.emp nocache; The table has been changed. After this query, only if the FTS occurred it was saved to JRU. 4.Bypass buffer cache: C:\&gt;exp scott/figer tables=emp file=d:\test.dmp DIRECT = Y 5. Use index appropriately If the clause is then used there index. 6. Improve the free buffer waiting to check that the files are evenly distributed across all devices. If this does not affect getting faster disks or place offending files onto faster disks. Disk.

download tegar mengharapkanmu di , arithmetic sequence practice workshe , rexazailuwiv.pdf , haier ac manual , technical support engineer interview questions and answers.pdf download , normal\_5fa58fca669e9.pdf , value-added products from beekeeping.pdf , normal\_5f96a328ddd64.pdf , normal\_5fa719396a14f.pdf , spygate the untold story , normal\_5fa88e339ae62.pdf , collins connect answers aqa ,