E
Statistics Descriptions

This appendix describes the statistics stored in the `V$SESSTAT` and `V$SYSSTAT` dynamic performance tables. These statistics are useful in identifying and correcting performance problems.

This appendix contains the following topics:

E.1 Displaying Statistics

The `V$SESSTAT` view displays statistics on a per-session basis and is valid only for the session currently connected. When a session disconnects, all statistics for the session are updated in `V$SYSSTAT`. The values for the statistics are cleared until the next session uses them.
The `V$STATNAME` view contains all of the statistics for an Oracle release.

Many of these statistics are tied to the internal implementation of Oracle and therefore are subject to change or deletion without notice, even between patch releases. Application developers should be aware of this and write their code to tolerate missing or extra statistics.

> **See Also:**
>
> "V$SESSTAT", "V$STATNAME", and "V$SYSSTAT" for more information on these views

E.2 Statistics Descriptions

This section describes some of the statistics stored in the `V$SESSTAT` and `V$SYSSTAT` views.

The statistics are listed in alphabetical order.

The `CLASS` column contains a number representing one or more statistics classes. The following class numbers are additive:

- 1, User
- 2, Redo
- 4, Enqueue
- 8, Cache
- 16, OS
- 32, Real Application Clusters
- 64, SQL
- 128, Debug

- 256, Instance level

For example, a class value of 72 represents a statistic that relates to SQL statements and caching.

Some statistics are populated only if the `TIMED_STATISTICS` initialization parameter is set to `true`. Those statistics are flagged with a `Y` in the right-hand column.

Table E-1 Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
application wait time	1	Total wait time (in centiseconds) for waits that belong to the Application wait class	
background checkpoints completed	8	Number of checkpoints completed by the background process. This statistic is incremented when the background process successfully advances the thread checkpoint.	
background checkpoints started	8	Number of checkpoints started by the background process. This statistic can be larger than "background checkpoints completed" if a new checkpoint overrides an incomplete checkpoint or if a checkpoint is currently under way. This statistic includes only checkpoints of the redo thread. It does not include: • Individual file checkpoints for operations such as offline or begin backup • Foreground (user-requested) checkpoints (for example, performed by `ALTER SYSTEM CHECKPOINT LOCAL` statements)	
background timeouts	128	Number of times a background process has set an alarm for itself and the alarm has timed out rather than the background process being posted by another process to do some work	
branch node splits	128	Number of times an index branch block was split because of the insertion of an additional value	
buffer is not pinned count	72	Number of times a buffer was free when visited. Useful only for internal debugging purposes.	
buffer is pinned count	72	Number of times a buffer was pinned when visited. Useful only for internal debugging purposes.	
bytes received via SQL*Net from client	1	Total number of bytes received from the client over Oracle Net Services	
bytes received via SQL*Net from dblink	1	Total number of bytes received from a database link over Oracle Net Services	
bytes sent via SQL*Net to client	1	Total number of bytes sent to the client from the foreground processes	
bytes sent via SQL*Net to dblink	1	Total number of bytes sent over a database link	
Cached Commit SCN referenced	128	Useful only for internal debugging purposes	
calls to get snapshot scn: kcmgss	32	Number of times a snapshot system change number (SCN) was allocated. The SCN is allocated at the start of a transaction.	
calls to kcmgas	128	Number of calls to routine kcmgas to get a new SCN	
calls to kcmgcs	128	Number of calls to routine kcmgcs to get a current SCN	
calls to kcmgrs	128	Number of calls to routine kcsgrs to get a recent SCN	
cell flash cache read hits	8	Number of read requests that were satisfied by the cache	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
cell flash cache read hits for smart IO	8	Number of read requests for smart IO that were satisfied by the cache	
cell flash cache read hits for temp IO	8	Number of read requests for temp IO that were satisfied by the cache	
cell IO uncompressed bytes	64	Total size of uncompressed data that is processed on the cell For operations on segments compressed using Exadata Hybrid Columnar Compression, this statistic is the size of the data after decompression.	
cell num bytes in passthru due to quarantine	8	Number of bytes that were not offloaded and sent back to the database for processing due to a quarantine on the cell	
cell num bytes in passthru during predicate offload	64	Number of bytes that were not offloaded and sent back to the database for processing	
cell overwrites in flash cache	8	Total number of write requests that overwrote an existing cacheline in Exadata Smart Flash Cache that had not been written to disk In effect, this is the amount of disk I/O saved by using Write-Back mode. This statistic is incremented once per mirror write.	
cell partial writes in flash cache	8	Total number of write requests written to both Exadata Smart Flash Cache and disk Part of the data was written to flash, and the rest was written to disk. This statistic is incremented once per mirror write.	
cell physical IO bytes added to storage index	8	Number of bytes added to the storage index during a Smart Scan This is an indication that the storage index is being built.	
cell physical IO bytes eligible for predicate offload	64	Number of bytes on-disk eligible for predicate offload	
cell physical IO bytes eligible for smart IOs	8	Number of actual bytes eligible for predicate offload For example, when using columnar cache, this is the size of columnar cache instead of the on-disk size.	
cell physical IO bytes processed for IM capacity	64	Number of bytes read from the columnar cache in `memcompress for capacity` format	
cell physical IO bytes processed for IM query	64	Number of bytes read from the columnar cache in `memcompress for query` format	
cell physical IO bytes processed for no memcompress	64	Number of bytes read from the columnar cache in `no memcompress` format	
cell physical IO bytes saved by columnar cache	8	Number of bytes saved by columnar cache, that is, the number of bytes of reading that were avoided	
cell physical IO bytes saved by storage index	8	Number of bytes saved by the storage index	
cell physical IO bytes saved during optimized file creation	64	Number of I/O bytes saved by the database host by offloading the file creation operation to the cells This statistic shows the benefit of optimized file creation operations.	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
cell physical IO bytes saved during optimized RMAN file restore	64	Number of I/O bytes saved by the database host by offloading the RMAN file restore operation to the cells This statistic shows the benefit of optimized RMAN file restore operations.	
cell physical IO bytes sent directly to DB node to balance CPU	64	Number of I/O bytes sent back to the database server for processing due to high storage server CPU usage	
cell physical IO interconnect bytes	64	Number of I/O bytes exchanged over the interconnect between the database host and the cells	
cell physical IO interconnect bytes returned by smart scan	64	Number of I/O bytes that are returned by the cell for Smart Scan operations This value does not include bytes for other database I/O.	
cell pmem cache read hits	8	Number of non-RDMA read requests processed by `cellsrv` resulting in a PMEM cache hit	
cell pmem cache writes	264	Number of non-RDMA write requests processed by `cellsrv` resulting in a PMEM cache write	
cell pmem log writes	258	Number of redo log write requests that used PMEM log	
cell ram cache read hits	8	Number of read requests that hit the RAM cache on the cell	
cell RDMA reads	8	Number of PMEM cache read requests using RDMA	
cell RDMA writes	264	Number of PMEM cache write requests using RDMA	
cell writes to flash cache	8	Total number of write requests written entirely to Exadata Smart Flash Cache This statistic is incremented once per mirror write.	
cell writes to flash cache for temp IO	8	Number of write requests for temporary segments that were absorbed by Exadata Smart Flash Cache	
change write time	8	Elapsed redo write time for changes made to `CURRENT` blocks in 10s of milliseconds.	Y
cleanouts and rollbacks - consistent read gets	128	Number of "consistent gets" that require both block rollbacks and block cleanouts	
cleanouts only - consistent read gets	128	Number of "consistent gets" that require only block cleanouts, no rollbacks	
cluster key scan block gets	64	Number of blocks obtained in a cluster scan	
cluster key scans	64	Number of cluster scans that were started	
cluster wait time	1	Total wait time (in centiseconds) for waits that belong to the Cluster wait class	
cold recycle reads	8	Number of buffers that were read through the least recently used end of the recycle cache with fast aging strategy	
commit cleanout failures: block lost	8	Number of times Oracle attempted a cleanout at commit but could not find the correct block due to forced write, replacement, or switch `CURRENT`	
commit cleanout failures: buffer being written	8	Number of times Oracle attempted a cleanout at commit, but the buffer was currently being written	
commit cleanout failures: callback failure	8	Number of times the cleanout callback function returns `FALSE`	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
commit cleanout failures: cannot pin	8	Total number of times a commit cleanout was performed but failed because the block could not be pinned	
commit cleanout failures: hot backup in progress	8	Number of times Oracle attempted block cleanout at commit during hot backup. The image of the block must be logged before the buffer can be made dirty.	
commit cleanout failures: pmem only	264	Number of time Oracle attempted a cleanout at commit, but could not modify PMEM buffers directly without a DRAM version	
commit cleanout failures: write disabled	8	Number of times a cleanout block at commit was performed but the writes to the database had been temporarily disabled	
commit cleanouts	8	Total number of times the cleanout block at commit function was performed	
commit cleanouts successfully completed	8	Number of times the cleanout block at commit function completed successfully	
commit nowait performed	1	Number of asynchronous commits that were actually performed. These commits did not wait for the commit redo to be flushed and be present on disk before returning.	
commit nowait requested	1	Number of no-wait commit or asynchronous commit requests that were made either using SQL or the OCI transaction control API	
Commit SCN cached	128	Number of times the system change number of a commit operation was cached	
commit wait/nowait performed	1	Number of asynchronous/synchronous commits that were actually performed	
commit wait/nowait requested	1	Number of no-wait or wait commits that were made either using SQL or the OCI transaction control API	
commit wait performed	1	Number of synchronous commits that were actually performed. These commits waited for the commit redo to be flushed and be present on disk before returning.	
commit wait requested	1	Number of waiting or synchronous commit requests that were made either using SQL or the OCI transaction control API	
concurrency wait time	1	Total wait time (in centiseconds) for waits that belong to the Concurrency wait class	
consistent changes	8	Number of times a user process has applied rollback entries to perform a consistent read on the block Work loads that produce a great deal of consistent changes can consume a great deal of resources. The value of this statistic should be small in relation to the "consistent gets" statistic.	
consistent gets	8	Number of times a consistent read was requested for a block. **See Also:** "consistent changes" and "session logical reads"	
consistent gets direct	8	Number of times a consistent read was requested for a block bypassing the buffer cache (for example, direct load operation). This is a subset of "consistent gets".	
consistent gets from cache	8	Number of times a consistent read was requested for a block from buffer cache. This is a subset of "consistent gets".	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
consistent gets from pmem	8	Number of direct-mapped blocks accessed in CR (consistent read) mode from PMEM. This is a subset of "consistent gets".	
consistent gets pmem direct	8	Number of CR gets from pmem in direct read (e.g. tablescan) code path. This is a subset of "consistent gets from pmem".	
consistent gets pmem examination	8	Number of direct-mapped blocks accessed and examined in CR mode from PMEM, via regular path. This is a subset of "consistent gets from pmem".	
consistent gets pmem examination (fastpath)	8	Number of direct-mapped blocks accessed and examined in CR mode from PMEM, via fast path. This is a subset of both "consistent gets from pmem" and "consistent gets pmem examination".	
consistent gets pmem pin	8	Number of direct-mapped blocks accessed and pinned in CR mode from PMEM, via regular path. This is a subset of "consistent gets from pmem".	
consistent gets pmem pin (fastpath)	8	Number of direct-mapped blocks accessed and pinned in CR mode from PMEM, via fast path. This is a subset of both "consistent gets from pmem" and "consistent gets pmem pin".	
CPU used by this session	1	Amount of CPU time (in 10s of milliseconds) used by a session from the time a user call starts until it ends. If a user call completes within 10 milliseconds, the start and end user-call time are the same for purposes of this statistic, and 0 milliseconds are added. A similar problem can exist in the reporting by the operating system, especially on systems that suffer from many context switches.	Y
CPU used when call started	128	Amount of CPU time (in 10s of milliseconds) used when the call is started **See Also:** " CPU used by this session"	Y
CR blocks created	8	Number of `CURRENT` blocks cloned to create CR (consistent read) blocks. The most common reason for cloning is that the buffer is held in a incompatible mode.	
cumulative begin requests	32	Total number of begin requests received by the Oracle database allowing the database to detect and process requests	
cumulative DB time in requests	288	Total DB Time (in microseconds) within requests to the Oracle database, measuring DB time used from the beginning to the end of each request. **See Also:** "DB time"	
cumulative DB time protected in requests	288	Total DB Time (in microseconds) within requests to the Oracle database protected by Application Continuity or Transparent Application Continuity, measuring DB time used from the beginning to the end of each request. **See Also:** "DB time" This is a subset of "cumulative DB time in requests", which enables you to calculate the ratio of total time in requests to protected time in requests. See also Application Continuity Protection Check (ACCHK).	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
cumulative end requests	32	Total number of end requests received by the Oracle database allowing the database to detect and process requests	
cumulative time in requests	32	Total wall clock time (in microseconds) within requests to the Oracle database, measuring time spent from the beginning to the end of each request. This statistic is used by draining and planned failover.	
cumulative user calls in requests	32	Total number of user calls received by the Oracle database within requests to the Oracle database	
cumulative user calls protected by Application Continuity	32	Total number of user calls received by the Oracle database within requests to the Oracle database that were protected by Application Continuity or Transparent Application Continuity	
current blocks converted for CR	8	Number `CURRENT` blocks converted to CR state	
cursor authentications	128	Number of privilege checks conducted during execution of an operation	
data blocks consistent reads - undo records applied	128	Number of undo records applied to data blocks that have been rolled back for consistent read purposes	
data warehousing cooling action	8	Number of times that cooling occurred on this instance	
data warehousing evicted objects	8	Number of times that objects got evicted by automatic big table caching on this instance	
data warehousing evicted objects - cooling	8	Number of times that objects got evicted on this instance due to a cooling action	
data warehousing evicted objects - replace	8	Number of times that objects got evicted due to caching replacement, that is, when an object is evicted because a hotter object forces it to be evicted from the cache	
data warehousing scanned blocks	8	Number of blocks scanned by automatic big table caching on this instance using parallel query	
data warehousing scanned blocks - disk	8	Number of blocks scanned by automatic big table caching on this instance by direct read from disk	
data warehousing scanned blocks - memory	8	Number of blocks scanned by automatic big table caching on this instance by cache read from memory	
data warehousing scanned blocks - offload	8	Number of blocks scanned by automatic big table caching on this instance by Exadata offloading	
data warehousing scanned objects	8	Number of times the objects in automatic big table caching are scanned using parallel query	
db block changes	8	Closely related to "consistent changes", this statistic counts the total number of changes that were part of an update or delete operation that were made to all blocks in the SGA. Such changes generate redo log entries and hence become permanent changes to the database if the transaction is committed. This approximates total database work. This statistic indicates the rate at which buffers are being dirtied (on a per-transaction or per-second basis, for example).	
db block gets	8	Number of times a `CURRENT` block was requested **See Also:** "consistent gets"	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
db block gets direct	8	Number of times a `CURRENT` block was requested bypassing the buffer cache (for example, a direct load operation). This is a subset of "db block gets".	
db block gets from cache	8	Number of times a `CURRENT` block was requested from the buffer cache. This is a subset of "db block gets".	
db block gets from pmem	8	Number of direct-mapped blocks accessed in `CURRENT` mode from PMEM, via regular path	
db block gets from pmem (fastpath)	8	Number of direct-mapped blocks accessed in `CURRENT` mode from PMEM, via fast path	
DB time	1	Total time (in microseconds) spent in database calls for foreground sessions. This statistic is an indicator of the total instance workload. It is measured cumulatively from the time of instance startup and is calculated by aggregating the CPU and wait times of all foreground sessions not waiting on idle wait events (non-idle user sessions).	
DB time of LWTs for this session	1	Total DB time (in microseconds) used by lightweight threads (LWTs) in a session. **See Also:** "DB time"	
DBWR checkpoint buffers written	8	Number of buffers that were written for checkpoints	
DBWR checkpoints	8	Number of times the DBWR was asked to scan the cache and write all blocks marked for a checkpoint or the end of recovery. This statistic is always larger than "background checkpoints completed".	
DBWR lru scans	8	Number of times that DBWR scans the LRU queue looking for buffers to write. This count includes scans to fill a batch being written for another purpose (such as a checkpoint).	
DBWR revisited being-written buffer	8	Number of times that DBWR tried to save a buffer for writing and found that it was already in the write batch. This statistic measures the amount of "useless" work that DBWR had to do in trying to fill the batch. Many sources contribute to a write batch. If the same buffer from different sources is considered for adding to the write batch, then all but the first attempt will be "useless" because the buffer is already marked as being written.	
DBWR transaction table writes	8	Number of rollback segment headers written by DBWR. This statistic indicates how many "hot" buffers were written, causing a user process to wait while the write completed.	
DBWR undo block writes	8	Number of rollback segment blocks written by DBWR	
DDL statements parallelized	32	Number of DDL statements that were executed in parallel	
deferred (CURRENT) block cleanout applications	128	Number of times cleanout records are deferred, piggyback with changes, always current get	
DFO trees parallelized	32	Number of times a serial execution plan was converted to a parallel plan	
dirty buffers inspected	8	Number of dirty buffers found by the user process while it is looking for a buffer to reuse	
DML statements parallelized	32	Number of DML statements that were executed in parallel	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
DML statements retried	64	When a long-running DML is executing, the cursor may get invalidated due to some concurrent DDL on one of the cursor's dependencies. In this case, an internal ORA-14403 error is thrown and is caught and cleared in one of the calling functions. The current work is rolled back and the DML is restarted without the user being notified of this. The statistic counts the number of times that the thrown, caught, and cleared (ORA-14403) sequence occurred for DML statements. Should a DML vary widely in execution time, check this statistic to see if it increments during the DML execution. If so, then concurrent DDL may be the cause of the extra elapsed time.	
enqueue conversions	4	Total number of conversions of the state of table or row lock	
enqueue deadlocks	4	Total number of deadlocks between table or row locks in different sessions	
enqueue releases	4	Total number of table or row locks released	
enqueue requests	4	Total number of table or row locks acquired	
enqueue timeouts	4	Total number of table and row locks (acquired and converted) that timed out before they could complete	
enqueue waits	4	Total number of waits that occurred during an enqueue convert or get because the enqueue get was deferred	
exchange deadlocks	8	Number of times that a process detected a potential deadlock when exchanging two buffers and raised an internal, restartable error. Index scans are the only operations that perform exchanges.	
execute count	64	Total number of calls (user and recursive) that executed SQL statements	
fbda woken up	128	Number of times the flashback data archive background process was woken up to do archiving	
file io wait time	1	Total time spent in wait (in microseconds) for I/O to datafiles, excluding the service time for such I/O. This is cumulative for all I/Os for all datafiles. The service time for one I/O operation is estimated as the minimum time spent in the I/O call seen so far. This service time is subtracted from the time spent in each I/O call to get the wait time for that I/O.	
flash cache eviction: aged out	8	Flash cache buffer is aged out of the Database Smart Flash Cache	
flash cache eviction: buffer pinned	8	Database Smart Flash Cache buffer is invalidated due to object or range reuse, and so on. The Database Flash Cache Buffer was in use at the time of eviction.	
flash cache eviction: invalidated	8	Database Smart Flash Cache buffer is invalidated due to object or range reuse, and so on. The Database Smart Flash Cache buffer was not in use at the time of eviction.	
flash cache insert skip: corrupt	8	In-memory buffer was skipped for insertion into the Database Smart Flash Cache because the buffer was corrupted	
flash cache insert skip: DBWR overloaded	8	In-memory buffer was skipped for insertion into the Database Smart Flash Cache because DBWR was busy writing other buffers	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
flash cache insert skip: exists	8	In-memory buffer was skipped for insertion into the Database Smart Flash Cache because it was already in the flash cache	
flash cache insert skip: modification	8	In-memory buffer was skipped for insertion into the Database Smart Flash Cache because it was being modified	
flash cache insert skip: not current	8	In-memory buffer was skipped for insertion into the Database Smart Flash Cache because it was not current	
flash cache insert skip: not useful	8	In-memory buffer was skipped for insertion into the Database Smart Flash Cache because the type of buffer was not useful to keep	
flash cache inserts	8	Total number of in-memory buffers inserted into the Database Smart Flash Cache	
flashback log write bytes	2	Total size in bytes of flashback database data written by RVWR to flashback database logs	
flashback log writes	2	Total number of writes by RVWR to flashback database logs	
foreground propagated tracked transactions	128	Number of transactions modifying tables enabled for flashback data archive which were archived by a foreground process	
free buffer inspected	8	Number of buffers skipped over from the end of an LRU queue to find a reusable buffer. The difference between this statistic and "dirty buffers inspected" is the number of buffers that could not be used because they had a user, a waiter, or were being read or written, or because they were busy or needed to be written after rapid aging out.	
free buffer inspected for pmem	8	Number of PMEM buffers skipped over from the end of an LRU queue to find a reusable PMEM buffer	
free buffer requested	8	Number of times a reusable buffer or a free buffer was requested to create or load a block	
free buffer requested for pmem	8	Number of times a reusable PMEM buffer or a free PMEM buffer was requested to directly access a block	
gc read wait failures	40	A read wait is when a CR server waits for a disk read to complete before serving a block to another instance. This statistic displays the number of times a read wait ended in failure, that is, after waiting it was unable to serve a block.	
gc read wait timeouts	40	A read wait is when a CR server waits for a disk read to complete before serving a block to another instance. This statistic displays the number of times a read wait timed out, that is, the disk read did not complete in time, so the wait was terminated.	
gc read waits	40	Number of times a CR server waited for a disk read, and then successfully served a block	
global enqueue CPU used by this session	32	Amount of CPU time (in 10s of milliseconds) used by synchronous and asynchronous global enqueue activity in a session from the time a user call starts until it ends. If a user call completes within 10 milliseconds, the start and end user-call time are the same for purposes of this statistic, and 0 milliseconds are added.	
global enqueue get time	32	Total elapsed time in 10s of milliseconds of all synchronous and asynchronous global enqueue gets and converts	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
global enqueue gets async	32	Total number of asynchronous global enqueue gets and converts	
global enqueue gets sync	32	Total number of synchronous global enqueue gets and converts	
global enqueue releases	32	Total number of synchronous global enqueue releases	
HCC analyze table CUs	128	Number of Compression Units that were decompressed for `ANALYZE TABLE`	
HCC analyzer calls	128	Number of times `ANALYZE TABLE` has been run on Hybrid Columnar Compression (HCC) compressed tables	
HCC block compressions attempted	128	Number of times the system failed to recompress an OLTP compressed block in HCC format	
HCC block compressions completed	128	Number of times an OLTP compressed block was successfully recompressed in HCC format	
HCC DML conventional	128	Number of searched DML statements (`UPDATE` and `DELETE`) issued against HCC compressed tables	
HCC DML CUs	128	Number of Compression Units that were decompressed for searched DMLs (`UPDATE` and `DELETE`)	
HCC fetch by rowid CUs	128	Number of Compression Units that were decompressed by single row fetch	
HCC load conventional bytes compressed	128	Length of the data after compression stored in Compression Units by conventional loads. The amount of space used on disk is the sum of the data stored plus the metadata needed to locate the stored data.	
HCC load conventional bytes uncompressed	128	Length of the data before compression stored in Compression Units by conventional loads. The amount of space used on disk is the sum of the data stored plus the metadata needed to locate the stored data.	
HCC load conventional CUs	128	Number of Compression Units that were successfully compressed in any format from conventional loads	
HCC load conventional CUs archive high	128	Number of Compression Units that were successfully compressed in HCC `ARCHIVE HIGH` format from conventional loads	
HCC load conventional CUs archive low	128	Number of Compression Units that were successfully compressed in HCC `ARCHIVE LOW` format from conventional loads	
HCC load conventional CUs query high	128	Number of Compression Units that were successfully compressed in HCC `QUERY HIGH` format from conventional loads	
HCC load conventional CUs query low	128	Number of Compression Units that were successfully compressed in HCC `QUERY LOW` format from conventional loads	
HCC load conventional rows	128	Number of rows that were successfully compressed in HCC format from conventional loads	
HCC load conventional rows not compressed	128	Number of rows that were unable to be compressed in HCC format from conventional loads because the compression resulted in taking more space, not less	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
HCC load direct bytes compressed	128	Length of the data after compression stored in Compression Units by Direct Path Loads. The amount of space used on disk is the sum of the data stored plus the metadata needed to locate the stored data.	
HCC load direct bytes uncompressed	128	Length of the data before compression stored in Compression Units by Direct Path Loads. The amount of space used on disk is the sum of the data stored plus the metadata needed to locate the stored data.	
HCC load direct CUs	128	Number of Compression Units that were successfully HCC compressed by Direct Path Loads	
HCC load direct CUs archive high	128	Number of rows from Direct Path Loads that were HCC compressed with `ARCHIVE HIGH` successfully	
HCC load direct CUs archive low	128	Number of rows from Direct Path Loads that were HCC compressed with `ARCHIVE LOW` successfully	
HCC load direct CUs query high	128	Number of rows from Direct Path Loads that were HCC compressed with `QUERY HIGH` successfully	
HCC load direct CUs query low	128	Number of rows from Direct Path Loads that were HCC compressed with `QUERY LOW` successfully	
HCC load direct rows	128	Number of rows from Direct Path Loads that were HCC compressed successfully	
HCC load direct rows not compressed	128	Number of rows from Direct Path Loads that were OLTP compressed instead of HCC compressed due to negative compression	
HCC scan cell bytes compressed	128	Length of the compressed data stored in Compression Units prior to decompression. This is the length of all columns, not just the columns accessed.	
HCC scan cell bytes decompressed	128	Length of the data after decompression stored in Compression Units. This is the length of all columns, not just the columns accessed.	
HCC scan cell CUs archive high	128	Number of Compression Units scanned that had been compressed with `ARCHIVE HIGH`	
HCC scan cell CUs archive low	128	Number of Compression Units scanned that had been compressed with `ARCHIVE LOW`	
HCC scan cell CUs columns accessed	128	Number of columns that needed to be decompressed on the storage server to process the Compression Unit	
HCC scan cell CUs decompressed	128	Number of Compression Units that needed to be decompressed on the storage server	
HCC scan cell CUs decompression time	128	Total time spent decompressing the columns that were needed on the storage server	
HCC scan cell CUs optimized read	128	Number of Compression Units where all the rows were within the stored Min/Max values. Evaluation of those predicates was optimized out.	
HCC scan cell CUs pruned	128	Number of Compression Units that were pruned based on the Min/Max value checks	
HCC scan cell CUs query high	128	Number of Compression Units scanned that had been compressed with `QUERY HIGH`	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
HCC scan cell CUs query low	128	Number of Compression Units scanned that had been compressed with `QUERY LOW`	
HCC scan cell CUs sent compressed	64	Number of Compression Units that the storage server returned to the RDBMS still compressed after predicate evaluation was done	
HCC scan cell CUs sent head piece	64	Number of Compression Units that the storage server was unable to process and the block header was returned to the RDBMS to be processed	
HCC scan cell CUs sent uncompressed	64	Number of Compression Units that the storage server returned to the RDBMS in uncompressed columnar format	
HCC scan cell rows	128	Number of rows from columnar formatted data that were returned by the storage server to the RDBMS	
HCC scan CUs pcode aggregation pushdown	128	Number of Compression Units that were aggregated on the cell using P(ortable)-byte code	
HCC scan CUs pcode pred evaled	128	Number of predicates that were evaluated using P(ortable)-byte code	
HCC scan CUs pcode pred evaled using rowsets	128	Number of predicates that were evaluated using P(ortable)-byte code using vectorized data	
HCC scan CUs predicates applied	128	Number of predicates where at least some value passed the Min/Max check	
HCC scan CUs predicates optimized	128	Number of predicates that could be evaluated directly against columnar data	
HCC scan CUs predicates received	128	Number of predicates sent to the storage server that could be evaluated directly against HCC format data after decompression	
HCC scan rdbms bytes compressed	128	Length of the compressed data stored in Compression Units prior to decompression. This is the length of all columns, not just the columns accessed.	
HCC scan rdbms bytes decompressed	128	Length of the data after decompression stored in Compression Units. This is the length of all columns, not just the columns accessed.	
HCC scan rdbms CUs archive high	128	Number of HCC Compression Units decompressed by the RDBMS table scan that had been compressed with `ARCHIVE HIGH`	
HCC scan rdbms CUs archive low	128	Number of HCC Compression Units decompressed by the RDBMS table scan that had been compressed with `ARCHIVE LOW`	
HCC scan rdbms CUs columns accessed	128	Number of columns that were decompressed	
HCC scan rdbms CUs decompressed	128	Number of Compression Units that were decompressed on the RDBMS server	
HCC scan rdbms CUs decompression time	128	Total time spent decompressing the columns that were needed on the RDBMS server	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
HCC scan rdbms CUs normal	128	Number of HCC Compression Units decompressed by the RDBMS regular table scan, i.e., 'Table Access Full' in a query plan. This is typically used in specialized scans that, for example, use row versions, row SCN, or use the `LONG RAW` datatype.	
HCC scan rdbms CUs pruned	128	Number of Compression Units that were eliminated by comparing the predicates to the stored Min and Max values for that Comprssion Unit	
HCC scan rdbms CUs query high	128	Number of HCC Compression Units decompressed by the RDBMS table scan that had been compressed with `QUERY HIGH` (which is the default)	
HCC scan rdbms CUs query low	128	Number of HCC Compression Units decompressed by the RDBMS table scan that had been compressed with `QUERY LOW`	
HCC scan rdbms CUs turbo	128	Number of HCC Compression Units decompressed by the RDBMS fast table scan (TurboScan). This is separate to decompression that happens on an Exadata cell.	
HCC scan rdbms rows	128	Number of rows returned from HCC blocks or that were returned from an Exadata cell in columnar format that passed the predicates on that table	
HCC scan rows pcode aggregated	128	Number of rows from HCC blocks or from columnar format blocks returned by an Exadata cell that could be aggregated using the P(ortable)-code style of query compilation	
HCC usage cloud	128	Internal count of how often TurboScan is invoked for HCC data stored in an Oracle Cloud environment. It does not include Oracle on other vendor's clouds. This may or may not correspond with the number of queries or granules scanned.	
HCC usage pillar	128	Internal count of how often TurboScan is invoked for HCC data stored on Pillar storage. This may or may not correspond with the number of queries or granules scanned.	
HCC usage ZFS	128	Internal count of how often TurboScan is invoked for HCC data stored on ZFS storage. This may or may not correspond with the number of queries or granules scanned.	
hot buffers moved to head of LRU	8	When a hot buffer reaches the tail of its replacement list, Oracle moves it back to the head of the list to keep it from being reused. This statistic counts such moves.	
hot buffers moved to head of LRU for pmem	8	When a hot PMEM buffer reaches the tail of its replacement list, Oracle moves it back to the head of the list to keep it from being reused. This statistic counts such moves.	
hot pmem block exchange with dram attempts	264	Number of hot PMEM buffers that were attempted to be exchanged with colder DRAM buffers, to ensure both blocks are still cached	
hot pmem block exchange with dram successes	264	Number of hot PMEM buffers that were successfully exchanged with colder DRAM buffers, with both blocks still cached	
hot pmem block migration to dram attempts	8	Number of hot PMEM buffer cache blocks that were attempted to be migrated to DREM to reduce access latency	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
hot pmem block migration to dram successes	264	Number of hot PMEM buffer cache blocks that were successfully migrated to DRAM to reduce access latency	
immediate (CR) block cleanout applications	128	Number of times cleanout records are applied immediately during consistent-read requests	
immediate (CURRENT) block cleanout applications	128	Number of times cleanout records are applied immediately during current gets. Compare this statistic with "deferred (CURRENT) block cleanout applications".	
IM (HPK4SQL) hash joins attempted	384	Number of in-memory vectorized hash joins attempted	
IM (HPK4SQL) hash joins completed	384	Number of in-memory vectorized hash joins completed	
IM (hybrid) scan blocks on hybrid list	384	Number of blocks on the hybrid In-Memory scan list	
IM (hybrid) scan rows on hybrid list	384	Number of rows on the hybrid In-Memory scan list	
IM default area resized	128	Amount of memory by which the column store got resized	
IM populate accumulated time (ms)	128	Total amount of DB time (in milliseconds) spent populating CUs into the IM column store due to segment scans	
IM populate bytes in-memory EU data	128	Size in bytes of in-memory EU data populated due to segment scans	
IM populate bytes uncompressed EU data	128	Uncompressed size in bytes of in-memory EU data populated due to segment scans	
IM populate CUs	128	Number of CUs populated in the IM column store due to segment scans	
IM populate CUs memcompress for capacity high	128	Number of CUs populated in the IM column store due to segment scans using memcompress for capacity high	
IM populate CUs memcompress for capacity low	128	Number of CUs populated in the IM column store due to segment scans using memcompress for capacity low	
IM populate CUs memcompress for dml	128	Number of CUs populated in the IM column store due to segment scans using memcompress for DML	
IM populate CUs memcompress for query high	128	Number of CUs populated in the IM column store due to segment scans using memcompress for query high	
IM populate CUs memcompress for query low	128	Number of CUs populated in the IM column store due to segment scans using memcompress for query low	
IM populate CUs no memcompress	128	Number of CUs populated in the IM column store due to segment scans without compression	
IM populate CUs requested	128	Number of CUs requested to be populated due to segment scans	
IM populate EUs	128	Number of EUs populated in the IM column store due to segment scans	
IM populate EUs accumulated time (ms)	128	Total amount of DB time (in milliseconds) spent populating EUs into the IM column store due to segment scans	
IM populate EUs columns	128	Number of columns populated in EUs due to segment scans	
IM populate EUs memcompress for capacity high	128	Number of EUs populated in the IM column store due to segment scans at memcompress for capacity high	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
IM populate EUs memcompress for capacity low	128	Number of EUs populated in the IM column store due to segment scans at memcompress for capacity low	
IM populate EUs memcompress for dml	128	Number of EUs populated in the IM column store due to segment scans at memcompress for dml	
IM populate EUs memcompress for query high	128	Number of EUs populated in the IM column store due to segment scans at memcompress for query high	
IM populate EUs memcompress for query low	128	Number of EUs populated in the IM column store due to segment scans at memcompress for query low	
IM populate EUs no memcompress	128	Number of EUs populated in the IM column store without compression due to segment scans	
IM populate EUs requested	128	Number of EUs requested to be populated in the IM column store due to segment scans	
IM populate no contiguous inmemory space	128	Number of CUs that fail to populate due to lack of contiguous space in the In-Memory Area	
IM populate segments	128	Number of segments populated due to segment scan	
IM populate segments requested	128	Number of segments requested to be populated due to segment scan	
IM populate segments wall clock time (ms)	128	Total amount of wall clock time (in milliseconds) spent populating CUs into the IM column store due to segment scans	
IM prepopulate accumulated time (ms)	128	Total amount of DB time (in milliseconds) spent prepopulating CUs into the IM column store priority	
IM prepopulate bytes in-memory EU data	128	Size in bytes of in-memory EU data populated due to priority	
IM prepopulate bytes uncompressed EU data	128	Uncompressed size in bytes of in-memory EU data populated due to priority	
IM prepopulate CUs	128	Number of CUs prepopulated in the IM column store due to priority	
IM prepopulate CUs memcompress for capacity high	128	Number of CUs prepopulated in the IM column store due to priority using memcompress for capacity high	
IM prepopulate CUs memcompress for capacity low	128	Number of CUs prepopulated in the IM column store due to priority using memcompress for capacity low	
IM prepopulate CUs memcompress for dml	128	Number of CUs prepopulated in the IM column store due to priority using memcompress for DML	
IM prepopulate CUs memcompress for query high	128	Number of CUs prepopulated in the IM column store due to priority using memcompress for query high	
IM prepopulate CUs memcompress for query low	128	Number of CUs prepopulated in the IM column store due to priority using memcompress for query low	
IM prepopulate CUs no memcompress	128	Number of CUs prepopulated in the IM column store due to priority without compression	
IM prepopulate CUs requested	128	Number of CUs requested to be prepopulated due to priority	
IM prepopulate EUs	128	Number of EUs populated in the IM column store due to priority	
IM prepopulate EUs accumulated time (ms)	128	Total amount of DB time (in milliseconds) spent populating EUs into the IM column store due to priority	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
IM prepopulate EUs columns	128	Number of columns populated in EUs due to priority	
IM prepopulate EUs memcompress for capacity high	128	Number of EUs populated in the IM column store due to priority at memcompress for capacity high	
IM prepopulate EUs memcompress for capacity low	128	Number of EUs populated in the IM column store due to priority at memcompress for capacity low	
IM prepopulate EUs memcompress for dml	128	Number of EUs populated in the IM column store due to priority at memcompress for dml	
IM prepopulate EUs memcompress for query high	128	Number of EUs populated in the IM column store due to priority at memcompress for query high	
IM prepopulate EUs memcompress for query low	128	Number of EUs populated in the IM column store due to priority at memcompress for query low	
IM prepopulate EUs no memcompress	128	Number of EUs populated in the IM column store without compression due to priority	
IM prepopulate EUs requested	128	Number of EUs requested to be populated in the IM column store due to priority	
IM prepopulate segments	128	Number of segments prepopulated due to priority	
IM prepopulate segments requested	128	Number of segments requested to be prepopulated due to priority	
IM repopulate accumulated time (ms)	128	Total amount of DB time (in milliseconds) spent repopulating CUs into the IM column store due to DML changes	
IM repopulate bytes in-memory EU data	128	Size in bytes of in-memory EU data repopulated due to EU reaching staleness threshold	
IM repopulate CUs	128	Total number of CUs requested to be repopulated due to CU reaching staleness threshold	
IM repopulate CUs memcompress for capacity high	128	Number of CUs repopulated in the IM column store using memcompress for capacity high due to CU reaching staleness threshold	
IM repopulate CUs memcompress for capacity low	128	Number of CUs repopulated in the IM column store using memcompress for capacity low due to CU reaching staleness threshold	
IM repopulate CUs memcompress for dml	128	Number of CUs repopulated in the IM column store using memcompress for DML due to CU reaching staleness threshold	
IM repopulate CUs memcompress for query high	128	Number of CUs repopulated in the IM column store using memcompress for query high due to CU reaching staleness threshold	
IM repopulate CUs memcompress for query low	128	Number of CUs repopulated in the IM column store using memcompress for query low due to CU reaching staleness threshold	
IM repopulate CUs no memcompress	128	Number of CUs repopulated in the IM column store without compression due to CU reaching staleness threshold	
IM repopulate CUs requested	128	Total number of CUs requested to be repopulated due to CU reaching staleness threshold	
IM repopulate (doublebuffering) CUs	128	Number of CUs repopulated with double-buffering enabled on the earlier version of the CUs	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
IM repopulate (doublebuffering) CUs requested	128	Number of CUs requested to be repopulated with double-buffering enabled on the earlier version of the CUs	
IM repopulate EUs	128	Number of EUs requested to be repopulated due to EU reaching staleness threshold	
IM repopulate EUs accumulated time (ms)	128	Total amount of DB time (in milliseconds) spent repopulating EUs into the IM column store due to DML changes	
IM repopulate EUs columns	128	Number of columns repopulated in EUs due to EU reaching staleness threshold	
IM repopulate EUs memcompress for capacity high	128	Number of EUs repopulated in the IM column store at memcompress for capacity high due to EU reaching staleness threshold	
IM repopulate EUs memcompress for capacity low	128	Number of EUs repopulated in the IM column store at memcompress for capacity low due to EU reaching staleness threshold	
IM repopulate EUs memcompress for dml	128	Number of EUs repopulated in the IM column store at memcompress for DML due to EU reaching staleness threshold	
IM repopulate EUs memcompress for query high	128	Number of EUs repopulated in the IM column store at memcompress for query high due to EU reaching staleness threshold	
IM repopulate EUs memcompress for query low	128	Number of EUs repopulated in the IM column store at memcompress for query low due to EU reaching staleness threshold	
IM repopulate EUs no memcompress	128	Number of EUs repopulated in the IM column store without compression due to EU reaching staleness threshold	
IM repopulate EUs requested	128	Total number of EUs requested to be repopulated due to EU reaching staleness threshold	
IM repopulate (incremental) CUs	128	Number of CUs repopulated incrementally from earlier versions of the CUs	
IM repopulate (incremental) CUs requested	128	Number of CUs requested to be repopulated incrementally from earlier versions of the CUs	
IM repopulate (incremental) EUs	128	Number of EUs repopulated using unchanged data from the current EU due to EU reaching staleness threshold	
IM repopulate (incremental) EUs requested	128	Number of EUs requested to be repopulated using unchanged data from the current EU due to EU reaching staleness threshold	
IM repopulate no contiguous inmemory space	128	Number of CUs that failed to repopulate due to lack of contigunous space in the In-Memory Area	
IM repopulate (scan) CUs	128	Number of CUs repopulated in the IM column store due to scans	
IM repopulate (scan) CUs requested	128	Number of CUs requested to be repopulated in the IM column store due to scans	
IM repopulate (scan) EUs	128	Number of EUs repopulated in the IM column store that were triggered by scans on the EU	
IM repopulate (scan) EUs requested	128	Number of EUs requested for repopulation in the IM column store that were triggered by scans on the EU	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
IM repopulate segments	128	Number of segments repopulated	
IM repopulate segments requested	128	Indicates the number of segments requested to be repopulated	
IM repopulate (trickle) accumulated time (ms)	128	Total amount of DB time (in milliseconds) spent trickle repopulating CUs into the IM column store due to DML changes	
IM repopulate (trickle) bytes in-memory EU data	128	Size in bytes of in-memory EU data repopulated due to DML changes	
IM repopulate (trickle) bytes uncompressed EU data	128	Uncompressed size in bytes of in-memory EU data repopulated due to EU reaching staleness threshold	
IM repopulate (trickle) CUs	128	Number of CUs trickle repopulated in the IM column store due to DML changes	
IM repopulate (trickle) CUs memcompress for capacity high	128	Number of CUs trickle repopulated in the IM column store using memcompress for capacity high due to DML changes	
IM repopulate (trickle) CUs memcompress for capacity low	128	Number of CUs trickle repopulated in the IM column store using memcompress for capacity low due to DML changes	
IM repopulate (trickle) CUs memcompress for dml	128	Number of CUs trickle repopulated in the IM column store using memcompress for DML due to DML changes	
IM repopulate (trickle) CUs memcompress for query high	128	Number of CUs trickle repopulated in the IM column store using memcompress for query high due to DML changes	
IM repopulate (trickle) CUs memcompress for query low	128	Number of CUs trickle repopulated in the IM column store using memcompress for query low due to DML changes	
IM repopulate (trickle) CUs no memcompress	128	Number of CUs trickle repopulated in the IM column store without compression due to DML changes	
IM repopulate (trickle) CUs requested	128	Total number of CUs requested to be trickle repopulated due to DML changes	
IM repopulate (trickle) CUs resubmitted	128	Number of CUs trickle repopulate tasks submitted	
IM repopulate (trickle) EUs	128	Number of EUs trickle repopulated in the IM column store due to DML changes	
IM repopulate (trickle) EUs accumulated time (ms)	128	Total amount of DB time (in milliseconds) spent trickle repopulating EUs into the IM column store due to DML changes	
IM repopulate (trickle) EUs columns	128	Number of columns repopulated in EUs due to DML changes	
IM repopulate (trickle) EUs memcompress for capacity high	128	Number of EUs trickle repopulated in the IM column store due to DML changes at memcompress for capacity high	
IM repopulate (trickle) EUs memcompress for capacity low	128	Number of EUs trickle repopulated in the IM column store due to DML changes at memcompress for capacity low	
IM repopulate (trickle) EUs memcompress for dml	128	Number of EUs trickle repopulated in the IM column store due to DML changes at memcompress for dml	
IM repopulate (trickle) EUs memcompress for query high	128	Number of EUs trickle repopulated in the IM column store due to DML changes at memcompress for query high	
IM repopulate (trickle) EUs memcompress for query low	128	Number of EUs trickle repopulated in the IM column store due to DML changes at memcompress for query low	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
IM repopulate (trickle) EUs no memcompress	128	Number of EUs trickle repopulated in the IM column store without compression due to DML changes	
IM repopulate (trickle) EUs requested	128	Number of EUs requested to be trickle repopulated in the IM column store due to DML changes	
IM scan CUs column not in memory	128	Number of extents that could not be read from the IM column store because one of the columns required was not in memory	
IM scan CUs invalid or missing revert to on disk extent	128	Number of extents where no IMCU exists	
IM scan CUs memcompress for query low	128	Number of memcompress for query high CUs scanned in the IM column store	
IM scan CUs memcompress for query high	128	Number of memcompress for query high CUs scanned in the IM column store	
IM scan CUs memcompress for capacity low	128	Number of memcompress for capacity low CUs scanned in the IM column store	
IM scan CUs memcompress for capacity high	128	Number of memcompress for capacity high CUs scanned in the IM column store	
IM scan CUs memcompress for dml	128	Number of memcompress for DML CUs scanned in the IM column store	
IM scan CUs predicates applied	128	Number of where clause predicates applied to the In-Memory storage index	
IM scan CUs predicates optimized	128	Number of where clause predicates applied to the IM column store for which either all rows pass min/max pruning via an In-Memory storage index or no rows pass min/max pruning	
IM scan CUs pruned	128	Number of CUs pruned by the storage index	
IM scan (dynamic) multi-threaded scans	128	Number of In-Memory table scans which benefited from In-Memory dynamic scans	
IM scan (dynamic) tasks processed by parent	128	Number of IMCUs processed normally because of Resource Manager limit	
IM scan (dynamic) tasks processed by thread	128	Number of IMCUs processed in parallel by a worker thread	
IM scan (dynamic) rows	128	Number of rows processed by In-Memory dynamic scans	
IM scan EU bytes in-memory	128	Size in bytes of in-memory EU data accessed by scans	
IM scan EU bytes uncompressed	128	Uncompressed size in bytes of in-memory EU data accessed by scans	
IM scan EU rows	128	Number of rows scanned from EUs in the IM column store before where clause predicate applied	
IM scan EUs columns accessed	128	Number of columns in the EUs accessed by scans	
IM scan EUs columns decompressed	128	Number of columns in the EUs decompressed by scans	
IM scan EUs columns theoretical max	128	Number of columns that would have been accessed from the EU if the scans looked at all columns	
IM scan EUs memcompress for capacity high	128	Number of memcompress for capacity high EUs scanned in the IM column store	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
IM scan EUs memcompress for capacity low	128	Number of memcompress for capacity low EUs scanned in the IM column store	
IM scan EUs memcompress for dml	128	Number of memcompress for DML EUs scanned in the IM column store	
IM scan EUs memcompress for query high	128	Number of memcompress for query high EUs scanned in the IM column store	
IM scan EUs memcompress for query low	128	Number of memcompress for query low EUs scanned in the IM column store	
IM scan EUs no memcompress	128	Number of uncompressed EUs scanned in the IM column store	
IM scan EUs split pieces	128	Number of split EU pieces among all IM EUs	
IM scan rows	128	Number of rows in scanned In-Memory Compression Units (IMCUs)	
IM scan rows optimized	128	Number of rows that were not scanned in the IM column store as they were pruned via a number of optimizations such as min/max pruning via In-Memory storage indexes	
IM scan rows projected	128	Number of rows returned from the IM column store	
IM scan rows valid	128	Number of rows scanned from the IM column store after applying valid vector	
IM scan segments minmax eligible	128	Number of CUs that are eligible for min/max pruning via storage index	
IM space CU bytes allocated	128	Number of In-Memory bytes allocated	
IM space CU creations initiated	128	Number of space requests for CUs	
IM space CU extents allocated	128	Number of In-Memory extents allocated	
IM space segments allocated	128	Number of snapshot segments created	
IM space segments freed	128	Number of snapshot segments deleted	
IM transactions	128	Number of transactions that triggered data to be journaled in the IM column store	
IM transactions CUs invalid	128	Number of CUs in the IM column store invalidated by transactions	
IM transactions rows invalidated	128	Number of rows in the IM column store invalidated by transactions	
IM transactions rows journaled	128	Number of rows logged in the transaction journal	
in call idle wait time	1	Total wait time (in microseconds) for waits that belong to the Idle wait class. **See Also:** "non-idle wait count" and "non-idle wait time"	
index cmph cu, uncomp sentinels	128	Number of CUs created with uncompressed sentinels	
index cmph dm, cu lock expand	128	Number of times CU lock structure expanded	
index cmph dm, cu migrate row	128	Number of times a row migrated from a CU	
index cmph dm, insert unpurge CU row	128	Number of times a CU row was unpurged during insert	
index cmph dm, purge dummy CU	128	Number of times dummy CU purged from leaf block	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
index cmph dm, split for cu lock expand	128	Number of times leaf block split for CU lock expansion	
index cmph dm, split for cu migrate row	128	Number of leaf block splits due to CU row migration	
index cmph ld, CU fit	128	Number of times load created a well sized CU, no space for uncompressed rows	
index cmph ld, CU fit, add rows	128	Number of times load created a well sized CU, with space for uncompressed rows	
index cmph ld, CU negative comp	128	Number of times load CU gave negative compression	
index cmph ld, CU over-est	128	Number of times load created an oversized CU	
index cmph ld, CU under-est	128	Number of times load created a small CU	
index cmph ld, infinite loop	128	Number of times shrink CU attempts resulted in uncompressed rows	
index cmph ld, lf blks flushed	128	Number of leaf blocks flushed by load	
index cmph ld, lf blks w/ und CU	128	Number of leaf blocks flushed with small CU	
index cmph ld, lf blks w/o CU	128	Number of leaf blocks flushed without a CU	
index cmph ld, lf blks w/o unc r	128	Number of leaf blocks flushed without uncompressed rows	
index cmph ld, retry in over-est	128	Number of times CU was resized after creating an oversized CU	
index cmph ld, rows compressed	128	Number of rows compressed by load	
index cmph ld, rows uncompressed	128	Number of rows left uncompressed by load	
index cmph sc, ffs decomp buffers	128	Number of blocks decompressed for fast scan	
index cmph sc, ffs decomp buffers released and found valid	128	Number of times decompressed CU buffer was reused by fast scan	
index cmph sc, ffs decomp buffers rows avail	128	Number of rows in decompressed buffer for fast scan	
index cmph sc, ffs decomp buffers rows used	128	Number of rows used from decompressed buffer for fast scan	
index cmph sc, ffs decomp failures	128	Number of time decompress CU was not possible for fast scan	
	128	Number of times 90-10 leaf block CU splits were made 50-50	
index cmph sp, leaf norecomp limit	128	Number of times leaf block recompression reached the recompression limit	
index cmph sp, leaf norecomp negcomp	128	Number of times leaf block recompression returned negative compression	
index cmph sp, leaf norecomp nospace	128	Number of times leaf block recompression returned not enough space	
index cmph sp, leaf norecomp notry	128	Number of times leaf block recompression not attempted	
index cmph sp, leaf norecomp oversize	128	Number of times leaf block recompression returned an oversized CU	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
index cmph sp, leaf norecomp zerocur	128	Number of times leaf block recompression returned a CU with 0 rows	
index cmph sp, leaf recomp fewer ucs	128	Number of CUs created with reduced number of sentinels	
index cmph sp, leaf recomp zero ucs	128	Number of CUs created with zero sentinels	
index cmph sp, leaf recompress	128	Number of times a leaf block CU was recompressed	
index cmpl co, prefix mismatch	128	Number of times reorg found a neighboring block prefix count mismatch	
index cmpl ro, blocks not compressed	128	Number of times prefix compression was not applied to avoid negative compression	
index cmpl ro, prefix change at block	128	Number of times prefix count was changed to an optimal value	
index cmpl ro, prefix no change at block	128	Number of times prefix count was already the optimal value	
index cmpl ro, reorg avoid load new block	128	Number of times a block reorg avoided a new block being created during load	
index cmpl ro, reorg avoid split	128	Number of times a block reorg avoided a block split during DML	
index fast full scans (direct read)	64	Number of fast full scans initiated using direct read	
index fast full scans (full)	64	Number of fast full scans initiated for full segments	
index fast full scans (rowid ranges)	64	Number of fast full scans initiated with rowid endpoints specified	
large tracked transactions	128	For tables tracked by flashback data archive, the number of transactions modifying rows in those tables which are large in terms of size or number of changes	
leaf node splits	128	Number of times an index leaf node was split because of the insertion of an additional value	
lob reads	8	Number of LOB API read operations performed in the session/system. A single LOB API read may correspond to multiple physical/logical disk block reads.	
lob writes	8	Number of LOB API write operations performed in the session/system. A single LOB API write may correspond to multiple physical/logical disk block writes.	
lob writes unaligned	8	Number of LOB API write operations whose start offset or buffer size is not aligned to the internal chunk size of the LOB. Writes aligned to chunk boundaries are the most efficient write operations. The internal chunk size of a LOB is available through the LOB API (for example, `DBMS_LOB.GETCHUNKSIZE()`).	
logons cumulative	1	Total number of logons since the instance started. Useful only in `V$SYSSTAT`. It gives an instance overview of all processes that logged on.	
logons current	1	Total number of current logons. Useful only in `V$SYSSTAT`.	
memopt r failed puts	128	Total failed puts on hash index	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
memopt r failed reads on blocks	128	Total lookup failures due to read failure on blocks because of concurrent changes	
memopt r failed reads on buckets	128	Total lookup failures due to concurrent hash bucket changes	
memopt r hits	128	Total hits on hash index – primary key found	
memopt r lookup detected CR buffer	128	Total lookup failures due to block pointed to by hash index being no longer the current version	
memopt r lookups	128	Total number of lookups on hash index	
memopt r misses	128	Total misses on hash index due to primary key not found	
memopt r puts	128	Total puts on hash index	
memopt r successful puts	128	Total successful puts on hash index	
messages received	128	Number of messages sent and received between background processes	
messages sent	128	Number of messages sent and received between background processes	
MLE JIT compilation duration cumulative	1	Total amount of cpu time (in milliseconds) spent by asynchronous just-in-time compilations for a session	
MLE JIT compilation duration max	1	Duration (in milliseconds) of the most time-consuming just-in-time compilation per session (not applicable for `V$SYSSTAT`)	
MLE JIT compilation error count	1	Number of just-in-time compilation operations in a session that failed with an error	
MLE JIT compilation success count	1	Number of successful just-in-time compilations in a session	
MLE full GC accumulated time	1	Cumulative time (in milliseconds) spent collecting the entire MLE heap	
MLE full GC count	1	Number of times the entire MLE heap was collected	
MLE incremental GC accumulated time	1	Cumulative time (in milliseconds) spent collecting the young-generation portion of the MLE heap	
MLE incremental GC count	1	Number of times the young-generation portion of the MLE heap was collected	
MLE total memory in use	1	Total amount of memory (in bytes) used for the MLE instance in a session, across all contexts	
no buffer to keep pinned count	72	Number of times a visit to a buffer attempted, but the buffer was not found where expected. Like "buffer is not pinned count" and "buffer is pinned count", this statistic is useful only for internal debugging purposes.	
no work - consistent read gets	128	Number of "consistent gets" that require neither block cleanouts nor rollbacks.	
non-idle wait count	1	Total number of waits performed with wait events that were not part of the Idle wait class. **See Also:** "in call idle wait time" and "non-idle wait time"	
non-idle wait time	1	Total wait time (in microseconds) for waits that do not belong to the Idle wait class. **See Also:** "in call idle wait time" and "non-idle wait count"	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
OLAP Aggregate Function Calc	64	Number of times the `AGGREGATE` function computes a parent value based on the values of its children.	
OLAP Aggregate Function Logical NA	64	Number of times an `AGGREGATE` function evaluates to a logical NA value. This could be because the `AGGINDEX` is on and the composite tuple does not exist.	
OLAP Aggregate Function Precompute	64	Number of times the `AGGREGATE` function is to compute a value and finds it precomputed in the cube.	
OLAP Custom Member Limit	64	Number of times an OLAP table function issues a custom member limit	
OLAP Engine Calls	64	Total number of OLAP transactions executed within the session. This value provides a general indication of the level of OLAP activity in the session.	
OLAP Fast Limit	64	Number of times an OLAP table function issues a fast limit	
OLAP Full Limit	64	Number of times an OLAP table function issues a full limit	
OLAP GID Limit	64	Number of times an OLAP table function issues a Cube Grouping ID (CGID) limit. Typically, this type of limit occurs for query rewrite transformations that resolve to a cube organized materialized view.	
OLAP Import Rows Loaded	64	Number of OLAP import rows loaded. This statistic provides the number of rows of the source cursor that are actually loaded into an Analytic Workspace (AW). The difference between the OLAP Import Rows Pushed and OLAP Import Rows Loaded provides the number of rejected rows.	
OLAP Import Rows Pushed	64	Number of OLAP import rows pushed. This statistic refers to the number of rows encountered from a source cursor and is useful during cube build operations.	
OLAP INHIER Limit	64	Number of times an OLAP table function issues an in-hierarchy limit. This type of limit can occur when you use cube dimension hierarchy views.	
OLAP Limit Time	64	Total time taken by all the OLAP Limit operations that were performed during the last call to the OLAP table function	
OLAP Paging Manager Cache Changed Page	64	Number of times the OLAP page pool is changed for any attached AW.	
OLAP Paging Manager Cache Hit	64	Number of times a requested page is found in the OLAP page pool. Use this statistic in conjunction with "OLAP Paging Manager Cache Miss" to determine the OLAP page pool efficiency ratio.	
OLAP Paging Manager Cache Miss	64	Number of times a requested page is not found in the OLAP page pool. Use this statistic in conjunction with "OLAP Paging Manager Cache Hit" to determine the OLAP page pool efficiency ratio.	
OLAP Paging Manager Cache Write	64	Number of times the OLAP paging manager writes to a page in the OLAP page pool	
OLAP Paging Manager New Page	64	Number of newly-created pages in the OLAP page pool that have not yet been written to the workspace LOB	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
OLAP Paging Manager Pool Size	64	Size (in bytes) of the OLAP page pool allocated to a session and the sum of all OLAP page pools in the system	
OLAP Perm LOB Read	64	Number of times data was read from the table where the AW is stored. These are permanent LOB reads.	
OLAP Row Id Limit	64	Number of times an OLAP table function issues a row ID limit	
OLAP Row Load Time	64	Total time spent loading rows into an AW during cube build and OLAP SQL import operations Use this statistic along with the "OLAP engine elapsed time" to measure time spent running OLAP engine routines that involve loading data into AWs from a SQL source. This statistic has the following levels of precision: • Low precision timer This captures the elapsed time of the entire fetch phase of the SQL cursor that is being loaded into AWs. It includes the SQL execution time that occurs during a fetch operation from a source cursor and time taken by the OLAP engine to populate AWs. • High precision timer This captures the elapsed time, excluding the SQL processing of the cursor being loaded. It records the time spent in the OLAP engine only. • Default timer precision: This is based on the `STATISTIC_LEVEL` parameter. If the low precision is used, then `STATISTICS_LEVEL` is `TYPICAL`. The high precision timer is used when `STATISTIC_LEVEL` is set to `ALL`. No timing is captured when `STATISTICS_LEVEL` is `BASIC`.	
OLAP Row Source Rows Processed	64	Number of rows processed by the OLAP row source	
OLAP Session Cache Hit	64	Number of times the requested, dynamically-aggregated value of an AW object, was found in the OLAP session cache.	
OLAP Session Cache Miss	64	Number of times the requested, dynamically-aggregated value of an AW object, was not found in the OLAP session cache.	
OLAP Temp Segment Read	64	Number of times data was read from a temporary segment and not from the OLAP page pool	
OLAP Temp Segments	64	Number of OLAP pages stored in temporary segments for analytic workspaces	
OLAP Unique Key Attribute Limit	64	Number of times an OLAP table function issues a unique key attribute limit	
opened cursors cumulative	1	In `V$SYSSTAT`: Total number of cursors opened since the instance started In `V$SESSTAT`: Total number of cursors opened since the start of the session	
opened cursors current	1	Total number of current open cursors	
OS CPU Qt wait time	1	Amount of time a session spends on the CPU run queue (in microseconds), waiting to get the CPU to run	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
OS Involuntary context switches	16	Number of context switches that were enforced by the operating system	
OS Signals received	16	Number of signals received	
OS Swaps	16	Number of swap pages	
OS Voluntary context switches	16	Number of voluntary context switches (for example, when a process gives up the CPU by a SLEEP() system call)	
Parallel operations downgraded 1 to 25 pct	32	Number of times parallel execution was requested and the degree of parallelism was reduced because of insufficient parallel execution servers	
Parallel operations downgraded 25 to 50 pct	32	Number of times parallel execution was requested and the degree of parallelism was reduced because of insufficient parallel execution servers	
Parallel operations downgraded 50 to 75 pct	32	Number of times parallel execution was requested and the degree of parallelism was reduced because of insufficient parallel execution servers	
Parallel operations downgraded 75 to 99 pct	32	Number of times parallel execution was requested and the degree of parallelism was reduced because of insufficient parallel execution servers	
Parallel operations downgraded to serial	32	Number of times parallel execution was requested but execution was serial because of insufficient parallel execution servers	
Parallel operations not downgraded	32	Number of times parallel execution was executed at the requested degree of parallelism	
parse count (describe)	64	Total number of parse calls on a describe cursor. This operation is a less expensive than a hard parse and more expensive than a soft parse.	
parse count (hard)	64	Total number of parse calls (real parses). A hard parse is a very expensive operation in terms of memory use, because it requires Oracle to allocate a workheap and other memory structures and then build a parse tree.	
parse count (total)	64	Total number of parse calls (hard, soft, and describe). A soft parse is a check on an object already in the shared pool, to verify that the permissions on the underlying object have not changed.	
parse time cpu	64	Total CPU time used for parsing (hard and soft) in 10s of milliseconds	Y
parse time elapsed	64	Total elapsed time for parsing, in 10s of milliseconds. Subtract "parse time cpu" from this statistic to determine the total waiting time for parse resources.	Y
physical maps pmem	264	Number of direct-mapped references acquired from FsDirect. Note that in steady state this is approximately the same as the number of unmaps, because each map requires an unmap (e.g. reuses a PMEM buffer).	
physical read bytes	8	Total size in bytes of all disk reads by application activity (and not other instance activity) only.	
physical read flash cache hits	8	Total number of reads from flash cache instead of disk	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
physical read IO requests	8	Number of read requests for application activity (mainly buffer cache and direct load operation) which read one or more database blocks per request. This is a subset of "physical read total IO requests".	
physical read requests optimized	8	Number of read requests that read one or more database blocks from the Database Smart Flash Cache or the Exadata Smart Flash Cache.	
physical read total bytes	8	Total size in bytes of disk reads by all database instance activity including application reads, backup and recovery, and other utilities. The difference between this value and "physical read bytes" gives the total read size in bytes by non-application workload.	
physical read total IO requests	8	Number of read requests which read one or more database blocks for all instance activity including application, backup and recovery, and other utilities. The difference between this value and "physical read total multi block requests" gives the total number of small I/O requests which are less than 128 kilobytes down to single block read requests.	
physical read total multi block requests	8	Total number of Oracle instance read requests which read 128 kilobytes or more in two or more database blocks per request for all instance activity including application, backup and recovery, and other utilities.	
physical reads	8	Total number of data blocks read from disk. This value can be greater than the value of "physical reads direct" plus "physical reads cache" as reads into process private buffers also included in this statistic.	
physical reads cache	8	Total number of data blocks read from disk into the buffer cache. This is a subset of "physical reads".	
physical reads cache prefetch	8	Number of contiguous and noncontiguous blocks that were prefetched.	
physical reads direct	8	Number of reads directly from disk, bypassing the buffer cache. For example, in high bandwidth, data-intensive operations such as parallel query, reads of disk blocks bypass the buffer cache to maximize transfer rates and to prevent the premature aging of shared data blocks resident in the buffer cache.	
physical reads direct (lob)	8	Number of buffers that were read directly for LOBs	
physical reads direct temporary tablespace	8	Number of buffers that were read directly from temporary tablespaces	
physical reads for flashback new	8	Number of blocks read for newing (that is, preparing a data block for a completely new change) blocks while flashback database is enabled	
physical reads pmem	8	Number of blocks read/copied from PMEM into DRAM cache	
physical reads pmem decrypt	8	Number of blocks read/copied from PMEM into DRAM cache due to the need for decryption. This is a subset of "physical reads pmem".	
physical reads pmem direct path	8	Number of blocks read from PMEM into PGA memory due to direct path access. This is a subset of "physical reads pmem".	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
physical reads pmem exclusive	8	Number of blocks read/copied from PMEM into DRAM cache due to exclusive access requests, which prevents other processes from accessing the block on PMEM directly. This is a subset of "physical reads pmem".	
physical reads pmem modification	8	Number of blocks read/copied from PMEM into DRAM cache due to modification. This is a subset of "physical reads pmem".	
physical reads pmem promote	8	Number of blocks read/copied from PMEM into DRAM cache due to hot block promotion. This is a subset of "physical reads pmem". This statistic is further categorized into "hot pmem block migration to dram successes" and ""hot pmem block exchange with dram successes".	
physical reads pmem rollback	8	Number of blocks read/copied from PMEM into DRAM cache due to CR rollback. This is a subset of "physical reads pmem".	
physical reads prefetch warmup	8	Number of data blocks that were read from the disk during the automatic prewarming of the buffer cache.	
physical unmaps pmem forced	264	Number of forfeited direct-mapped references requested by FsDirect due to PMEM memory pressure	
physical write bytes	8	Total size in bytes of all disk writes from the database application activity (and not other kinds of instance activity).	
physical write IO requests	8	Number of write requests for application activity (mainly buffer cache and direct load operation) which wrote one or more database blocks per request.	
physical write total bytes	8	Total size in bytes of all disk writes for the database instance including application activity, backup and recovery, and other utilities. The difference between this value and "physical write bytes" gives the total write size in bytes by non-application workload.	
physical write total IO requests	8	Number of write requests which wrote one or more database blocks from all instance activity including application activity, backup and recovery, and other utilities. The difference between this stat and "physical write total multi block requests" gives the number of single block write requests.	
physical write total multi block requests	8	Total number of Oracle instance write requests which wrote two or more blocks per request to the disk for all instance activity including application activity, recovery and backup, and other utilities.	
physical writes	8	Total number of data blocks written to disk. This value equals the sum of "physical writes direct" and "physical writes from cache".	
physical writes direct	8	Number of writes directly to disk, bypassing the buffer cache (as in a direct load operation)	
physical writes direct (lob)	8	Number of buffers that were directly written for LOBs	
physical writes direct temporary tablespace	8	Number of buffers that were directly written for temporary tablespaces	
physical writes from cache	8	Total number of data blocks written to disk from the buffer cache. This is a subset of "physical writes".	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
physical writes non checkpoint	8	Number of times a buffer is written for reasons other than advancement of the checkpoint. Used as a metric for determining the I/O overhead imposed by setting the `FAST_START_IO_TARGET` parameter to limit recovery I/Os. (Note that `FAST_START_IO_TARGET` is a deprecated parameter.) Essentially this statistic measures the number of writes that would have occurred had there been no checkpointing. Subtracting this value from "physical writes" gives the extra I/O for checkpointing.	
pinned buffers inspected	8	Number of times a user process, when scanning the tail of the replacement list looking for a buffer to reuse, encountered a cold buffer that was pinned or had a waiter that was about to pin it. This occurrence is uncommon, because a cold buffer should not be pinned very often.	
pinned buffers inspected for pmem	8	Number of times a user process, when scanning the tail of the replacement list looking for a PMEM buffer to reuse in DRAM, encountered a cold buffer that was pinned or had a waiter that was about to pin it. This occurrence is uncommon, because a cold buffer should not be pinned very often.	
prefetched blocks aged out before use	8	Number of contiguous and noncontiguous blocks that were prefetched but aged out before use	
process last non-idle time	128	The last time this process executed	Y
PX local messages recv'd	32	Number of local messages received for parallel execution within the instance local to the current session	
PX local messages sent	32	Number of local messages sent for parallel execution within the instance local to the current session	
PX remote messages recv'd	32	Number of remote messages received for parallel execution within the instance local to the current session	
PX remote messages sent	32	Number of remote messages sent for parallel execution within the instance local to the current session	
queries parallelized	32	Number of `SELECT` statements executed in parallel	
recovery array read time	8	Elapsed time of I/O during recovery	
recovery array reads	8	Number of reads performed during recovery	
recovery blocks read	8	Number of blocks read during recovery	
recovery blocks read for lost write detection	8	Number of blocks read for lost write checks during recovery.	
recovery blocks skipped lost write checks	8	Number of Block Read Records that skipped the lost write check during recovery.	
recursive calls	1	Number of recursive calls generated at both the user and system level. Oracle maintains tables used for internal processing. When Oracle needs to make a change to these tables, it internally generates an internal SQL statement, which in turn generates a recursive call.	
recursive cpu usage	1	Total CPU time used by non-user calls (recursive calls). Subtract this value from "CPU used by this session" to determine how much CPU time was used by the user calls.	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
redo blocks checksummed by FG (exclusive)	2	Number of exclusive redo blocks that were checksummed by the generating foreground processes. An exclusive redo block is the one whose entire redo content belongs to a single redo entry.	
redo blocks checksummed by LGWR	2	Number of redo blocks that were checksummed by the LGWR.	
redo blocks written	2	Total number of redo blocks written. This statistic divided by "redo writes" equals number of blocks per write.	
redo buffer allocation retries	2	Total number of retries necessary to allocate space in the redo buffer. Retries are needed either because the redo writer has fallen behind or because an event such as a log switch is occurring.	
redo entries	2	Number of times a redo entry is copied into the redo log buffer	
redo entries for lost write detection	2	Number of times a Block Read Record is copied into the log buffer.	
redo log space requests	2	Number of times the active log file is full and Oracle must wait for disk space to be allocated for the redo log entries. Such space is created by performing a log switch. Log files that are small in relation to the size of the SGA or the commit rate of the work load can cause problems. When the log switch occurs, Oracle must ensure that all committed dirty buffers are written to disk before switching to a new log file. If you have a large SGA full of dirty buffers and small redo log files, a log switch must wait for DBWR to write dirty buffers to disk before continuing. Also examine the **log file space** and **log file space switch** wait events in `V$SESSION_WAIT`.	
redo log space wait time	2	Total time waited in centiseconds for available space in the redo log buffer. See also "redo log space requests"	Y
redo ordering marks	2	Number of times that a system change number was allocated to force a redo record to have a higher SCN than a record generated in another thread using the same block	
redo size	2	Total amount of redo generated in bytes	
redo size for lost write detection	2	Total amount of Block Read Records generated in bytes	
redo synch time	8	Elapsed time of all "redo synch writes" calls in 10s of milliseconds	Y
redo synch writes	8	Number of times the redo is forced to disk, usually for a transaction commit. The log buffer is a circular buffer that LGWR periodically flushes. Usually, redo that is generated and copied into the log buffer need not be flushed out to disk immediately.	
redo wastage	2	Number of bytes wasted because redo blocks needed to be written before they are completely full. Early writing may be needed to commit transactions, to be able to write a database buffer, or to switch logs.	
redo write broadcast ack count	2	Number of times a commit broadcast acknowledgment has not been received by the time when the corresponding log write is completed. This is only for Oracle RAC.	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
redo write broadcast ack time	2	Total amount of the latency associated with broadcast on commit beyond the latency of the log write (in microseconds). This is only for Oracle RAC.	Y
redo write time	2	Total elapsed time of the write from the redo log buffer to the current redo log file in 10s of milliseconds	Y
redo writes	2	Total number of writes by LGWR to the redo log files. "redo blocks written" divided by this statistic equals the number of blocks per write	
rollback changes - undo records applied	128	Number of undo records applied to user-requested rollback changes (not consistent-read rollbacks)	
rollbacks only - consistent read gets	128	Number of "consistent gets" that require only block rollbacks, no block cleanouts.	
rows fetched via callback	64	Rows fetched via callback. Useful primarily for internal debugging purposes.	
scheduler wait time	1	Total wait time (in microseconds) for waits that belong to the Scheduler wait class	
SCN increments due to another database	128	SCN increments due to communication with another database	
serializable aborts	1	Number of times a SQL statement in a serializable isolation level had to terminate	
session connect time	1	The connect time for the session in 10s of milliseconds. This value is useful only in `V$SESSTAT`. It is the wall clock time since the logon to this session occurred.	Y
session cursor cache count	64	Total number of cursors cached. This statistic is incremented only if `SESSION_CACHED_CURSORS` > 0. This statistic is the most useful in `V$SESSTAT`. If the value for this statistic in `V$SESSTAT` is close to the setting of the `SESSION_CACHED_CURSORS` parameter, the value of the parameter should be increased.	
session cursor cache hits	64	Number of hits in the session cursor cache. A hit means that the SQL (including recursive SQL) or PL/SQL statement did not have to be reparsed. Subtract this statistic from "parse count (total)" to determine the real number of parses that occurred.	
session logical reads	1	The sum of "db block gets" plus "consistent gets". This includes logical reads of database blocks from either the buffer cache or process private memory.	
session logical reads - IM	128	Number of database blocks read from the IM column store (number of blocks in IMCU - number of blocks with invalid rows)	
session pga memory	1	Current PGA size for the session (in bytes). Useful only in `V$SESSTAT`; it has no meaning in `V$SYSSTAT`.	
session pga memory max	1	Peak PGA size for the session (in bytes). Useful only in `V$SESSTAT`; it has no meaning in `V$SYSSTAT`.	
session stored procedure space	1	Amount of memory this session is using for stored procedures	
session uga memory	1	Current UGA size for the session (in bytes). Useful only in `V$SESSTAT`; it has no meaning in `V$SYSSTAT`.	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
session uga memory max	1	Peak UGA size for a session (in bytes). Useful only in V$SESSTAT; it has no meaning in V$SYSSTAT.	
shared hash latch upgrades - no wait	8	A shared hash latch upgrade is when a hash latch is upgraded from shared mode to exclusive mode. This statistic displays the number of times the upgrade completed immediately.	
shared hash latch upgrades - wait	8	A shared hash latch upgrade is when a hash latch is upgraded from shared mode to exclusive mode. This statistic displays the number of times the upgrade did not complete immediately.	
shared io pool buffer get failure	128	Number of unsuccessful buffer gets from the shared I/O pool from instance startup time.	
shared io pool buffer get success	128	Number of successful buffer gets from the shared I/O pool from instance startup time.	
slave propagated tracked transactions	128	Number of transactions modifying tables enabled for flashback data archive which were archived by a worker process	
sorts (disk)	64	Number of sort operations that required at least one disk write Sorts that require I/O to disk are quite resource intensive. Try increasing the size of the initialization parameter SORT_AREA_SIZE.	
sorts (memory)	64	Number of sort operations that were performed completely in memory and did not require any disk writes You cannot do much better than memory sorts, except maybe no sorts at all. Sorting is usually caused by selection criteria specifications within table join SQL operations.	
sorts (rows)	64	Total number of rows sorted	
SQL*Net roundtrips to/from client	1	Total number of Oracle Net Services messages sent to and received from the client	
SQL*Net roundtrips to/from dblink	1	Total number of Oracle Net Services messages sent over and received from a database link	
summed dirty queue length	8	The sum of the dirty LRU queue length after every write request. Divide by **write requests** to get the average queue length after write completion.	
switch current from pmem	8	Number of times the CURRENT PMEM block moved to a DRAM buffer, leaving a CR block in the original PMEM buffer. This happens when one Oracle process pins a CURRENT PMEM buffer for read, and another Oracle process wants to pin the same buffer for modification. The latter process clones the PMEM buffer into a new DRAM buffer for modification, while converting the original PMEM current buffer into a PMEM CR buffer.	
switch current to new buffer	8	Number of times the CURRENT block moved to a different buffer, leaving a CR block in the original buffer	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
table fetch by rowid	64	Number of rows that are fetched using a ROWID (usually recovered from an index) This occurrence of table scans usually indicates either non-optimal queries or tables without indexes. Therefore, this statistic should increase as you optimize queries and provide indexes in the application.	
table fetch continued row	64	Number of times a chained or migrated row is encountered during a fetch Retrieving rows that span more than one block increases the logical I/O by a factor that corresponds to the number of blocks than need to be accessed. Exporting and re-importing may eliminate this problem. Evaluate the settings for the storage parameters `PCTFREE` and `PCTUSED`. This problem cannot be fixed if rows are larger than database blocks (for example, if the `LONG` data type is used and the rows are extremely large).	
table scan blocks gotten	64	During scanning operations, each row is retrieved sequentially by Oracle. This statistic counts the number of blocks encountered during the scan. This statistic tells you the number of database blocks that you had to get from the buffer cache for the purpose of scanning. Compare this value with the value of "consistent gets" to determine how much of the consistent read activity can be attributed to scanning.	
table scan disk IMC fallback	128	Number of rows fetched from the buffer cache because they were not present in the IM column store (in a scan that was otherwise performed in memory)	
table scan disk non-IMC rows gotten	128	Number of rows fetched during non-In-Memory scan	
table scan rows gotten	64	Number of rows that are processed during scanning operations	
table scans (cache partitions)	64	Number of range scans performed on tables that have the CACHE option enabled	
table scans (direct read)	64	Number of table scans performed with direct read (bypassing the buffer cache)	
table scans (IM)	128	Number of segments / granules scanned using In-Memory	
table scans (long tables)	64	Long (or conversely short) tables can be defined as tables that do not meet the short table criteria as described in "table scans (short tables)"	
table scans (rowid ranges)	64	During parallel query, the number of table scans conducted with specified ROWID ranges	
table scans (short tables)	64	Long (or conversely short) tables can be defined by optimizer hints coming down into the row source access layer of Oracle. The table must have the CACHE option set.	
tracked rows	128	Number of rows modified in tables enabled for flashback data archive	
tracked transactions	128	Number of transactions which modified a table enabled for flashback data archive	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
transaction lock background get time	128	Useful only for internal debugging purposes	
transaction lock background gets	128	Useful only for internal debugging purposes	
transaction lock foreground requests	128	Useful only for internal debugging purposes	
transaction lock foreground wait time	128	Useful only for internal debugging purposes	
transaction rollbacks	128	Number of transactions being successfully rolled back	
transaction tables consistent read rollbacks	128	Number of times rollback segment headers are rolled back to create consistent read blocks	
transaction tables consistent reads - undo records applied	128	Number of undo records applied to transaction tables that have been rolled back for consistent read purposes	
True Cache potentially current buffer made CR	264	Count of data blocks arriving at True Cache as potentially current buffers due to timing conditions, and later deemed to be good only as consistent read buffers, so they will be aged out.	
True Cache potentially current buffer made current	264	Count of data blocks arriving at True Cache as potentially current buffers due to timing conditions, and later confirmed to be real current buffers, so redo will continue to apply.	
True Cache: message count data send	256	Total number of messages this primary instance sends to True Cache for returning data blocks.	
True Cache: message count request send	256	Total number of messages this True Cache sends to the primary database for requesting data blocks.	
True Cache: message roundtrip time data send	256	Cumulative, elapsed, round-trip messaging time in microseconds of this primary instance sending data blocks to True Cache.	
True Cache: message roundtrip time request send	256	Cumulative, elapsed, round-trip messaging time in microseconds of this True Cache sending data block fetching requests to the primary database.	
TrueCache: block requests to preferred primary	264	Number of block fetch requests sent to the preferred primary Oracle RAC instance with object or undo affinity.	
TrueCache: block requests to primary	264	Total number of block fetch requests sent to the primary database.	
txns rollback priority_txns_high_wait_target	384	Total number of times a `LOW` or `MEDIUM` priority transaction was rolled back by the Priority Transactions feature because the wait time specified by the `PRIORITY_TXNS_HIGH_WAIT_TARGET` initialization parameter was exceeded for a `HIGH` priority transaction This statistic is updated when the Priority Transactions feature is enabled, that is, when the `PRIORITY_TXNS_MODE` initialization parameter is set to `ROLLBACK`.	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
txns rollback priority_txns_medium_wait_target	384	Total number of times a LOW priority transaction was rolled back by the Priority Transactions feature because the wait time specified by the PRIORITY_TXNS_MEDIUM_WAIT_TARGET initialization parameter was exceeded for a MEDIUM priority transaction This statistic is updated when the Priority Transactions feature is enabled, that is, when the PRIORITY_TXNS_MODE initialization parameter is set to ROLLBACK.	
txns track mode priority_txns_high_wait_target	384	Total number of times a LOW or MEDIUM priority transaction would potentially be rolled back by the Priority Transactions feature because the wait time specified by the PRIORITY_TXNS_HIGH_WAIT_TARGET initialization parameter was exceeded for a HIGH priority transaction This statistic is updated when the Priority Transactions feature is running in tracking mode, that is, when the PRIORITY_TXNS_MODE initialization parameter is set to TRACK.	
txns track mode priority_txns_medium_wait_target	384	Total number of times a LOW priority transaction would potentially be rolled back by the Priority Transactions feature because the wait time specified by the PRIORITY_TXNS_MEDIUM_WAIT_TARGET initialization parameter was exceeded for a MEDIUM priority transaction This statistic is updated when the Priority Transactions feature is running in tracking mode, that is, when the PRIORITY_TXNS_MODE initialization parameter is set to TRACK.	
user calls	1	Number of user calls such as login, parse, fetch, or execute When determining activity, the ratio of user calls to RPI calls, give you an indication of how much internal work gets generated because of the type of requests the user is sending to Oracle.	
user commits	1	Number of user commits. When a user commits a transaction, the redo generated that reflects the changes made to database blocks must be written to disk. Commits often represent the closest thing to a user transaction rate.	
user I/O wait time	1	Total wait time (in centiseconds) for waits that belong to the User I/O wait class	
user rollbacks	1	Number of times users manually issue the ROLLBACK statement or an error occurs during a user's transactions	
very large tracked transactions	128	For tables tracked by flashback data archive, number of transactions modifying those tables which are very large in terms of size or number of changes	
write clones created in background	8	Number of times a background or foreground process clones a CURRENT buffer that is being written. The clone becomes the new, accessible CURRENT buffer, leaving the original buffer (now the clone) to complete writing.	

Table E-1 (Cont.) Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
write clones created in foreground	8	Number of times a background or foreground process clones a `CURRENT` buffer that is being written. The clone becomes the new, accessible `CURRENT` buffer, leaving the original buffer (now the clone) to complete writing.	