9

Dynamic Performance (V$) Views: V$IM_COLUMN_LEVEL to V$RULE_SET_AGGREGATE_STATS

This chapter contains the dynamic performance views `V$IM_COLUMN_LEVEL` to `V$RULE_SET_AGGREGATE_STATS`.

9.1 V$IM_COLUMN_LEVEL

`V$IM_COLUMN_LEVEL` presents the selective column compression levels that are defined using the `inmemory_memcompress` clause of the `inmemory_column_clause` of the `CREATE TABLE` statement.

This view returns no rows for a table which has no associated selective column compression levels.

Column	Datatype	Description
`INST_ID`	`NUMBER`	Instance ID
`OWNER`	`VARCHAR2(31)`	Username of the table owner
`OBJ_NUM`	`NUMBER`	Table object number
`TABLE_NAME`	`VARCHAR2(31)`	Table name
`SEGMENT_COLUMN_ID`	`NUMBER`	Segment column number
`COLUMN_NAME`	`VARCHAR2(31)`	Column name
`INMEMORY_COMPRESSION`	`VARCHAR2(26)`	Column compression level. Possible values: • `DEFAULT`: This value appears for physical columns of a table that do not have a selective column clause. It also appears when a virtual column has been explicitly enabled for in-memory storage using `ALTER TABLE` *table-name* `INMEMORY(VC)`. • `UNSPECIFIED`: This value appears for virtual columns of a table that do not have a selective column clause. If the `INMEMORY_VIRTUAL_COLUMNS` initialization parameter is set to `MANUAL`, a virtual column with a `DEFAULT INMEMORY_COMPRESSION` clause will be materialized while a virtual column with an `UNSPECIFIED` value will not be.
`CON_ID`	`NUMBER`	The ID of the container to which the data pertains. Possible values include: • `0`: This value is used for rows containing data that pertain to the entire multitenant container database (CDB). This value is also used for rows in non-CDBs. • `1`: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

> **Note:**
>
> The SYS_IME hidden virtual columns automatically added by the In-Memory Expression (IME) infrastructure will not be shown in this view.

> **See Also:**
>
> - "INMEMORY_CLAUSE_DEFAULT"
> - "V$IM_SEGMENTS"
> - "V$IM_USER_SEGMENTS"
> - *Oracle Database In-Memory Guide* for an introduction to the IM column store
> - *Oracle Database SQL Language Reference* for more information about the `inmemory_column_clause` of the SQL `CREATE TABLE` statement

9.2 V$IM_SEGMENTS

`V$IM_SEGMENTS` presents information about all the in-memory segments in the database.

Only segments that have an in-memory representation are displayed. If a segment is marked for the In-Memory Column Store (IM column store) but is not populated, no corresponding row for that segment is displayed in this view.

Column	Datatype	Description
OWNER	VARCHAR2(128)	User name of the segment owner
SEGMENT_NAME	VARCHAR2(128)	Name of the segment, if any
PARTITION_NAME	VARCHAR2(128)	Object partition name (set to NULL for non-partitioned objects)
SEGMENT_TYPE	VARCHAR2(18)	Type of segment: • TABLE • TABLE PARTITION • TABLE SUBPARTION
TABLESPACE_NAME	VARCHAR2(30)	Name of the tablespace containing the segment
INMEMORY_SIZE	NUMBER	Size of the in-memory version of the segment, in bytes
BYTES	NUMBER	Number of on-disk data bytes for the segment that could be represented in memory (no space metadata blocks)
BYTES_NOT_POPULATED	NUMBER	Size of the portion of the on-disk segment that is not populated in memory, in bytes.
POPULATE_STATUS	VARCHAR2(13)	Status of segment population: • STARTED: A populate task for the segment has started • COMPLETED: There are no populate tasks pending for the segment • OUT OF MEMORY: A populate task for the segment failed due to lack of space in the IM column store

Column	Datatype	Description
INMEMORY_PRIORITY	VARCHAR2(8)	Indicates the priority for IM column store population: • LOW • MEDIUM • HIGH • CRITICAL • NONE
INMEMORY_DISTRIBUTE	VARCHAR2(15)	Indicates how the IM column store is distributed in an Oracle Real Application Clusters (Oracle RAC) environment: • AUTO • BY ROWID RANGE • BY PARTITION • BY SUBPARTITION
INMEMORY_DUPLICATE	VARCHAR2(13)	Indicates the duplicate setting for the IM column store in an Oracle RAC environment: • NO DUPLICATE • DUPLICATE • DUPLICATE ALL
INMEMORY_COMPRESSION	VARCHAR2(17)	Compression level for the IM column store: • NO MEMCOMPRESS • FOR DML • FOR QUERY [LOW | HIGH] • FOR CAPACITY [LOW | HIGH] • AUTO
INMEMORY_SERVICE	VARCHAR2(12)	Specifies how the IM-enabled table is populated on various instances: • DEFAULT: Pre-Oracle Database 12c Release 2 (12.2.0.1) behavior • NONE: Do not populate on any instance • ALL: Populate on all instances • USER_DEFINED: Populate only on the instances on which the user-specified service is active. The service name corresponding to this is stored in the INMEMORY_SERVICE_NAME column.
INMEMORY_SERVICE_NAME	VARCHAR2(129)	Specifies the service name on which the IM-enabled table should be populated. This column has a value only when the corresponding INMEMORY_SERVICE column has a value of USER_DEFINED.
IS_EXTERNAL	VARCHAR2(5)	Indicates whether the IM segment is for an external table. Possible values: • TRUE: The IM segment is for an external table. • FALSE: The IM segment is not for an external table.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire multitenant container database (CDB). This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

See Also:

- "INMEMORY_CLAUSE_DEFAULT"
- "V$IM_COLUMN_LEVEL"
- "V$IM_USER_SEGMENTS"
- *Oracle Database In-Memory Guide* for an introduction to the IM column store

9.3 V$IM_USER_SEGMENTS

`V$IM_USER_SEGMENTS` presents information about the in-memory segments for the current user in the database.

Only segments that have an in-memory representation are displayed. If a segment is marked for the In-Memory Column Store (IM column store) but is not populated, no corresponding row for that segment is displayed in this view.

Column	Datatype	Description
SEGMENT_NAME	VARCHAR2(128)	Name of the segment, if any
PARTITION_NAME	VARCHAR2(128)	Object partition name (set to NULL for non-partitioned objects)
SEGMENT_TYPE	VARCHAR2(18)	Type of segment: • TABLE • TABLE PARTITION • TABLE SUBPARTION
TABLESPACE_NAME	VARCHAR2(30)	Name of the tablespace containing the segment
INMEMORY_SIZE	NUMBER	Size of the in-memory version of the segment, in bytes
BYTES	NUMBER	Total size of the on-disk segment, in bytes
BYTES_NOT_POPULATED	NUMBER	Size of the portion of the on-disk segment that is not populated in memory, in bytes
POPULATE_STATUS	VARCHAR2(13)	Status of segment population: • STARTED: A populate task for the segment has started • COMPLETED: There are no populate tasks pending for the segment • OUT OF MEMORY: A populate task for the segment failed due to lack of space in the IM column store
INMEMORY_PRIORITY	VARCHAR2(8)	Indicates the priority for IM column store population: • LOW • MEDIUM • HIGH • CRITICAL • NONE
INMEMORY_DISTRIBUTE	VARCHAR2(15)	Indicates how the IM column store is distributed in an Oracle Real Application Clusters (Oracle RAC) environment: • AUTO • BY ROWID RANGE • BY PARTITION • BY SUBPARTITION

Column	Datatype	Description
INMEMORY_DUPLICATE	VARCHAR2(13)	Indicates the duplicate setting for the IM column store in an Oracle RAC environment: • NO DUPLICATE • DUPLICATE • DUPLICATE ALL
INMEMORY_COMPRESSION	VARCHAR2(17)	Compression level for the IM column store: • NO MEMCOMPRESS • FOR DML • FOR QUERY [LOW | HIGH] • FOR CAPACITY [LOW | HIGH] • AUTO
INMEMORY_SERVICE	VARCHAR2(12)	Specifies how the IM-enabled table is populated on various instances: • DEFAULT: Pre-Oracle Database 12c Release 2 (12.2.0.1) behavior • NONE: Do not populate on any instance • ALL: Populate on all instances • USER_DEFINED: Populate only on the instances on which the user-specified service is active. The service name corresponding to this is stored in the INMEMORY_SERVICE_NAME column.
INMEMORY_SERVICE_NAME	VARCHAR2(129)	Specifies the service name on which the IM-enabled table should be populated. This column has a value only when the corresponding INMEMORY_SERVICE column has a value of USER_DEFINED.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire multitenant container database (CDB). This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

See Also:

- "INMEMORY_CLAUSE_DEFAULT"
- "V$IM_COLUMN_LEVEL"
- "V$IM_SEGMENTS"
- *Oracle Database In-Memory Guide* for an introduction to the IM column store

9.4 V$INDEX_USAGE_INFO

V$INDEX_USAGE_INFO keeps track of index usage since the last flush. A flush occurs every 15 minutes. After each flush, ACTIVE_ELEM_COUNT is reset to 0 and LAST_FLUSH_TIME is updated to the current time.

Column	Datatype	Description
INDEX_STATS_ENABLED	NUMBER	Indicates whether the index usage statistics are enabled. Possible values: • 0: Index statistics are disabled • 1: Index statistics are enabled
INDEX_STATS_COLLECTION_TYPE	NUMBER	Indicates the type of collection used for the index usage statistics. Possible values: • 0: Indicates the ALL collection type. With this type of collection, the statistics are collected for each execution that has index access. Selecting this statistics collection type may have some impact on performance. • 1: Indicates the SAMPLED collection type. With this type of collection, the statistics are collected based on sampling (only a few of the executions are considered when collecting the statistics). This is the default statistics collection type. Index statistics collected with the SAMPLED collection type are less accurate than index statistics collected with the ALL collection type.
ACTIVE_ELEM_COUNT	NUMBER	The number of active indexes since the last flush
ALLOC_ELEM_COUNT	NUMBER	The number of index entries allocated
MAX_ELEM_COUNT	NUMBER	The maximum number of active indexes that can be tracked
FLUSH_COUNT	NUMBER	Number of successful flushes since the database started
TOTAL_FLUSH_DURATION	NUMBER	Cumulative elapsed time taken to complete the index usage statistics flush since the database start
LAST_FLUSH_TIME	TIMESTAMP(3)	The time of the last flush
STATUS_MSG	VARCHAR2(256)	Status messages, if any. Flush errors are reported here.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This row is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

See Also:

"DBA_INDEX_USAGE"

9.5 V$INDEXED_FIXED_COLUMN

V$INDEXED_FIXED_COLUMN displays the columns in dynamic performance tables that are indexed (X$ tables).

The X$ tables can change without notice. Use this view only to write queries against fixed views (V$ views) more efficiently.

Column	Datatype	Description
TABLE_NAME	VARCHAR2(128)	Name of the dynamic performance table that is indexed
INDEX_NUMBER	NUMBER	Number that distinguishes to which index a column belongs
COLUMN_NAME	VARCHAR2(128)	Name of the column that is being indexed
COLUMN_POSITION	NUMBER	Position of the column in the index key (this is mostly relevant for multicolumn indexes)
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire multitenant container database (CDB). This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.6 V$INMEMORY_AREA

V$INMEMORY_AREA contains information on the space allocation inside the In-Memory Area.

The In-Memory Area is sub-divided into three pools:

- 1MB POOL - stores the actual column formatted data populated into memory
- 64K POOL - stores metadata about the objects that are populated into the In-Memory Column Store (IM column store)
- IM POOL METADATA - contains information about In-Memory Compression Unit (IMCU) to data block mappings

The amount of available memory in each pool is visible in the V$INMEMORY_AREA view. The relative size of the three pools is determined by internal heuristics. The majority of the In-Memory Area memory is allocated to the 1MB POOL

Column	Datatype	Description
POOL	VARCHAR2(26)	Name of the pools in the In-Memory Area
ALLOC_BYTES	NUMBER	Total amount of memory allocated to this pool
USED_BYTES	NUMBER	Amount of memory currently used in this pool
POPULATE_STATUS	VARCHAR2(26)	Shows the status of the IM column store, for example, whether it is currently being populated or if it is done
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire multitenant container database (CDB). This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

See Also:

- "INMEMORY_CLAUSE_DEFAULT"
- "V$IM_COLUMN_LEVEL"
- "V$IM_SEGMENTS"
- "V$IM_USER_SEGMENTS"
- *Oracle Database In-Memory Guide* for an introduction to the IM column store

9.7 V$INMEMORY_FASTSTART_AREA

`V$INMEMORY_FASTSTART_AREA` provides information about the In-Memory FastStart (IM FastStart) area.

Column	Datatype	Description
`CON_ID`	`NUMBER`	The ID of the container to which the data pertains. Possible values include: • `0`: This value is used for rows containing data that pertain to the entire multitenant container database (CDB). This value is also used for rows in non-CDBs. • `1`: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data
`TABLESPACE_NAME`	`VARCHAR2(128)`	IM FastStart tablespace name. When IM FastStart is not enabled, the value of `TABLESPACE_NAME` is `INVALID_TABLESPACE` and the value of `STATUS` is `DISABLE`.
`STATUS`	`VARCHAR2(10)`	IM FastStart status. Possible values include: • `ENABLE`: An IM FastStart tablespace has been specified and the content of the IM column is being periodically checkpointed to disk. • `ENABLING`: An IM FastStart tablespace has been specified and the database is creating the IM FastStart area. • `DISABLE`: An IM FastStart tablespace has not been specified. This is the default. • `MIGRATING`: A user has requested the IM FastStart area be migrated from one tablespace to another. • `DISABLING`: A user has requested IM FastStart to be disabled.
`ALLOCATED_SIZE`	`NUMBER`	The allocated size of an IM FastStart tablespace in bytes
`USED_SIZE`	`NUMBER`	The currently used size of an IM FastStart area (in bytes) within the tablespace.
`LAST_CHECKPOINT_TIME`	`TIMESTAMP(6)`	The time when the last IMCU was checkpointed to the IM FastStart area
`LAST_POPULATE_TIME`	`TIMESTAMP(6)`	The time of the last population from the IM FastStart area
`NUM_DEFERRED_WRITES`	`NUMBER`	The number of pending deferred writes to the IM FastStart area

9.8 V$INMEMORY_SIZE_ADVICE

V$INMEMORY_SIZE_ADVICE estimates usage and performance statistics for different simulated sizes of the In-Memory Column Store (IM Column Store).

Column	Datatype	Description
IM_POOL_SIZE_FOR_ESTIMATE	NUMBER	Simulated size of the IM Column Store (in megabytes)
SIZE_FACTOR	NUMBER	Size factor with respect to the current IM Column Store size
ESTD_IM_USED_SIZE	NUMBER	Estimated amount of space the IM Column Store would use for this simulation (in megabytes)
ESTD_IM_SCAN_TIME	NUMBER	Estimated in-memory scan time for this simulation (in seconds)
ESTD_TOTAL_DB_TIME	NUMBER	Estimated total database time for this simulation (in seconds) This value estimates the cumulative time spent by the database on processing user requests. It includes the wait time and CPU time of all non-idle user sessions.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

> **Note:**
>
> This view is available starting with Oracle Database 23ai.

9.9 V$INSTANCE

V$INSTANCE displays the state of the current instance.

Column	Datatype	Description
INSTANCE_NUMBER	NUMBER	Instance number used for instance registration (corresponds to the INSTANCE_NUMBER initialization parameter) **See Also:** "INSTANCE_NUMBER"
INSTANCE_NAME	VARCHAR2(16)	Name of the instance
HOST_NAME	VARCHAR2(64)	Name of the host machine
VERSION	VARCHAR2(17)	Database version
VERSION_LEGACY	VARCHAR2(17)	The legacy database version used before Oracle Database 18c. This column displays the same value as the VERSION column
VERSION_FULL	VARCHAR2(17)	The version string with the new Oracle Database version scheme introduced in Oracle Database 18c.

Column	Datatype	Description
STARTUP_TIME	DATE	Time when the instance was started
STATUS	VARCHAR2(12)	Status of the instance: • STARTED - After STARTUP NOMOUNT • MOUNTED - After STARTUP MOUNT or ALTER DATABASE CLOSE • OPEN - After STARTUP or ALTER DATABASE OPEN • OPEN MIGRATE - After ALTER DATABASE OPEN { UPGRADE | DOWNGRADE }
PARALLEL	VARCHAR2(3)	Indicates whether the instance is mounted in cluster database mode (YES) or not (NO)
THREAD#	NUMBER	Redo thread opened by the instance
ARCHIVER	VARCHAR2(7)	Automatic archiving status: • STOPPED • STARTED • FAILED - Archiver failed to archive a log last time but will try again within 5 minutes
LOG_SWITCH_WAIT	VARCHAR2(15)	Event that log switching is waiting for: • ARCHIVE LOG • CLEAR LOG • CHECKPOINT • NULL - ALTER SYSTEM SWITCH LOGFILE is hung but there is room in the current online redo log
LOGINS	VARCHAR2(10)	Indicates whether the instance is in unrestricted mode, allowing logins by all users (ALLOWED, or in restricted mode, allowing logins by database administrators only (RESTRICTED)
SHUTDOWN_PENDING	VARCHAR2(3)	Indicates whether a shutdown is pending (YES) or not (NO)
DATABASE_STATUS	VARCHAR2(17)	Status of the database: • ACTIVE • SUSPENDED • INSTANCE RECOVERY
INSTANCE_ROLE	VARCHAR2(18)	Indicates whether the instance is an active instance (PRIMARY_INSTANCE) or an inactive secondary instance (SECONDARY_INSTANCE), or UNKNOWN if the instance has been started but not mounted

Column	Datatype	Description
ACTIVE_STATE	VARCHAR2(9)	Quiesce state of the instance: • NORMAL - Database is in a normal state. • QUIESCING - ALTER SYSTEM QUIESCE RESTRICTED has been issued: no new user transactions, queries, or PL/SQL statements are processed in this instance. User transactions, queries, or PL/SQL statements issued before the ALTER SYSTEM QUIESCE RESTRICTED statement are unaffected. DBA transactions, queries, or PL/SQL statements are also unaffected. • QUIESCED - ALTER SYSTEM QUIESCE RESTRICTED has been issued: no user transactions, queries, or PL/SQL statements are processed. DBA transactions, queries, or PL/SQL statements are unaffected. User transactions, queries, or PL/ SQL statements issued after the ALTER SYSTEM QUIESCE RESTRICTED statement are not processed. A single ALTER SYSTEM QUIESCE RESTRICTED statement quiesces all instances in an Oracle RAC environment. After this statement has been issued, some instances may enter into a quiesced state before other instances; the system is quiesced when all instances enter the quiesced state.
BLOCKED	VARCHAR2(3)	Indicates whether all services are blocked (YES) or not (NO)
CON_ID	NUMBER	The ID of the container to which the data pertains: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data
INSTANCE_MODE	VARCHAR2(11)	The instance mode of the current instance: • REGULAR: A regular Oracle RAC instance. This value is also always used for any non-Oracle RAC instance. • READ MOSTLY: An Oracle RAC instance that performs very few database writes • READ ONLY: A read-only Oracle RAC instance
EDITION	VARCHAR2(7)	The edition of the database: • PO: Personal Edition • FREE: Oracle Database Free • SE2: Standard Edition or Standard Edition 2 • EE: Enterprise Edition • HP: Enterprise Edition - High Performance • XP: Enterprise Edition - Extreme Performance
FAMILY	VARCHAR2(80)	For internal use only.
DATABASE_TYPE	VARCHAR2(15)	Database type: • RAC: If the database is a regular Oracle RAC database which may have multiple instances. • RACONENODE: If the database is Oracle RAC, but allows only one instance to run at any time - the RAC One Node mode. • SINGLE: If the database is running as a single instance. • UNKNOWN: If the database's type can't be determined. This might happen when the database is registered as a DB resource with CRS but the CRS service has failed to return valid database type information. Typically, this indicates that either the CRS service is down or it is in a faulty state.

9.10 V$INSTANCE_CACHE_TRANSFER

V$INSTANCE_CACHE_TRANSFER displays statistics for the cache blocks transferred among instances.

Oracle keeps multiple versions of data buffered in the buffer cache. The current buffer (or block), CURRENT_BLOCK, is the most up-to-date copy, containing all recent modifications. A consistent read buffer (or block), CR_BLOCK, contains the version of the data at a particular time prior to the current buffer. It is read-consistent (that is, all the data shown in that buffer are consistent for the start time of a query).

Therefore, for the same data block there can be multiple copies in the buffer cache: one current copy, and one or more consistent read copies with data consistent as of different snapshot times.

Column	Datatype	Description
INSTANCE	NUMBER	Instance from which the blocks are transferred
CLASS	VARCHAR2(18)	Class of the cache block
LOST	NUMBER	The number of blocks that were sent by a particular instance but that never arrived in this instance
LOST_TIME	NUMBER	The time waited for blocks that were sent by a particular instance but that never arrived in this instance
CR_BLOCK	NUMBER	CR Block transfers not affected by remote processing delays
CR_BLOCK_TIME	NUMBER	Total time waited for CR blocks from a particular instance (includes the other times)
CR_2HOP	NUMBER	The count of CR blocks which were received by this instance from a particular instance after a 2-way round-trip
CR_2HOP_TIME	NUMBER	The time waited for CR blocks which were received by this instance from a particular instance after a 2-way round-trip
CR_3HOP	NUMBER	The count of CR blocks which were received by this instance from a particular instance after a 3-way round-trip
CR_3HOP_TIME	NUMBER	The time waited for CR blocks which were received by this instance from a particular instance after a 3-way round-trip
CR_RDMA	NUMBER	The count of CR Blocks which were directly read from a remote instance via RDMA
CR_RDMA_TIME	NUMBER	Total time waited to directly read CR blocks from a remote instance via RDMA (in centiseconds)
CR_BUSY	NUMBER	CR Block transfers affected by remote contention
CR_BUSY_TIME	NUMBER	The time waited for CR blocks which were received by this instance from a particular instance and which were delayed by a log flushed on the sending instance
CR_CONGESTED	NUMBER	CR Block transfers affected by remote system load
CR_CONGESTED_TIME	NUMBER	The time waited for CR blocks which were received by this instance from a particular instance and which were delayed because LMS was busy
CURRENT_BLOCK	NUMBER	Current block transfers not affected by remote processing delays
CURRENT_BLOCK_TIME	NUMBER	Total time waited for current blocks from a particular instance (includes the other times

Column	Datatype	Description
CURRENT_2HOP	NUMBER	The count of current blocks which were received by this instance from a particular instance after a 2-way round-trip
CURRENT_2HOP_TIME	NUMBER	The time waited for current blocks which were received by this instance from a particular instance after a 2-way round-trip
CURRENT_3HOP	NUMBER	The count of current blocks which were received by this instance from a particular instance after a 3-way round-trip
CURRENT_3HOP_TIME	NUMBER	The time waited for current blocks which were received by this instance from a particular instance after a 3-way round-trip
CURRENT_RDMA	NUMBER	The count of current blocks which were directly read from a remote instance via RDMA
CURRENT_RDMA_TIME	NUMBER	Total time waited to directly read current blocks from a remote instance via RDMA (in centiseconds)
CURRENT_BUSY	NUMBER	Current block transfers affected by remote contention
CURRENT_BUSY_TIME	NUMBER	The time waited for current blocks which were received by this instance from a particular instance and which were delayed by a log flushed on the sending instance
CURRENT_CONGESTED	NUMBER	Current block transfers affected by remote system load
CURRENT_CONGESTED_TIME	NUMBER	The time waited for current blocks which were received by this instance from a particular instance and which were delayed because LMS was busy
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.11 V$INSTANCE_PING

V$INSTANCE_PING provides information about measured latency of the interconnect for all instances in an Oracle Real Application Clusters (Oracle RAC) environment.

In an Oracle RAC environment, every few seconds the PING process of each instance checks the response of the interconnect to all instances of the same database.

It sends two messages. One message is 500 bytes in size (referred to in the column descriptions as 500B), and the other is 8 kilobytes in size (referred to in column descriptions as 8K).

For each message sent to each instance, the amount of time it took to get a response back is measured (in microseconds). The view records the latest measurements as well as cumulative data since instance startup.

Column	Datatype	Description
INSTANCE	NUMBER	The instance communicated with. In the V$ view for each instance, there is one row for that instance, and in the GV$ view for each instance, there is also one row for that instance, so there are n^2 rows for *n* instances.

Column	Datatype	Description
CURRENT_500B	NUMBER	The latest response time for the 500B message (in microseconds)
AVERAGE_500B	NUMBER	The average response time for the 500B messages since instance startup (in microseconds). Note that AVERAGE_500B should be the same as WAIT_TIME_500B/COUNT_500B.
MAX_500B	NUMBER	The maximal response time for the 500B messages since instance startup (in microseconds)
COUNT_500B	NUMBER	The number of measurements for the 500B messages since instance startup
WAIT_TIME_500B	NUMBER	The sum of all response times for the 500B messages since instance startup (in microseconds)
WAIT_TIME_SQUARED_500B	NUMBER	The sum of the response time squared for 500B messages since instance startup. The unit is in microseconds squared and divided by 1000.
CURRENT_8K	NUMBER	The latest response time for the 8K message (in microseconds)
AVERAGE_8K	NUMBER	The average response time for the 8K messages since instance startup (in microseconds). Note that AVERAGE_8K should be the same as WAIT_TIME_8K/COUNT_8K.
MAX_8K	NUMBER	The maximal response time for the 8K messages since instance startup (in microseconds)
COUNT_8K	NUMBER	The number of measurements for the 8K messages since instance startup
WAIT_TIME_8K	NUMBER	The sum of all response times for the 8K messages since instance startup (in microseconds)
WAIT_TIME_SQUARED_8K	NUMBER	The sum of the response time squared for 8K messages since instance startup. The unit is in microseconds squared and divided by 16.
CON_ID	NUMBER	The ID of the container to which the data pertains. For this view, the possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs.

9.12 V$INSTANCE_RECOVERY

V$INSTANCE_RECOVERY monitors the mechanisms available to users to limit recovery I/O. Those mechanisms are:

- Set the LOG_CHECKPOINT_TIMEOUT initialization parameter
- Set the LOG_CHECKPOINT_INTERVAL initialization parameter
- Set the FAST_START_MTTR_TARGET initialization parameter
- Set the size of the smallest redo log

Column	Datatype	Description
RECOVERY_ESTIMATED_IOS	NUMBER	Number of dirty buffers in the buffer cache.
ACTUAL_REDO_BLKS	NUMBER	Current actual number of redo blocks required for recovery
TARGET_REDO_BLKS	NUMBER	Current target number of redo blocks that must be processed for recovery. This value is the minimum value of the following 3 columns, and identifies which of the 3 user-defined limits determines checkpointing.

Column	Datatype	Description
LOG_FILE_SIZE_REDO_BLKS	NUMBER	Maximum number of redo blocks required to guarantee that a log switch does not occur before the checkpoint completes.
LOG_CHKPT_TIMEOUT_REDO_BLKS	NUMBER	Number of redo blocks that need to be processed during recovery to satisfy the LOG_CHECKPOINT_TIMEOUT parameter. The value displayed is not meaningful unless that parameter has been set.
LOG_CHKPT_INTERVAL_REDO_BLKS	NUMBER	Number of redo blocks that need to be processed during recovery to satisfy the LOG_CHECKPOINT_INTERVAL parameter. The value displayed is not meaningful unless that parameter has been set.
FAST_START_IO_TARGET_REDO_BLKS	NUMBER	This column is obsolete and maintained for backward compatibility. The value of this column is always null.
TARGET_MTTR	NUMBER	Effective MTTR (mean time to recover) target value in seconds. The TARGET_MTTR value is calculated based on the value of the FAST_START_MTTR_TARGET parameter (the TARGET_MTTR value is used internally), and is usually an approximation of the parameter's value. However, if the FAST_START_MTTR_TARGET parameter value is very small (for example, one second), or very large (for example, 3600 seconds), the calculation will produce a target value dictated by system limitations. In such cases, the TARGET_MTTR value will be the shortest calculated time, or the longest calculated time that recovery is expected to take.
ESTIMATED_MTTR	NUMBER	Current estimated mean time to recover (MTTR) based on the number of dirty buffers and log blocks. Basically, this value tells you how long you could expect recovery to take based on the work your system is doing right now.
CKPT_BLOCK_WRITES	NUMBER	Number of blocks written by checkpoint writes
OPTIMAL_LOGFILE_SIZE	NUMBER	Redo log file size (in megabytes) that is considered optimal based on the current setting of FAST_START_MTTR_TARGET. It is recommended that the user configure all online redo logs to be at least this value. Note that redo log files must be at least 4 megabytes in size; otherwise an error is generated.
ESTD_CLUSTER_AVAILABLE_TIME	NUMBER	Estimated time (in seconds) that the cluster would become partially available should this instance fail. This column is only meaningful in an Oracle Real Application Clusters (Oracle RAC) environment. In a non-Oracle RAC environment, the value of this column is null.
WRITES_MTTR	NUMBER	Number of writes driven by the FAST_START_MTTR_TARGET initialization parameter
WRITES_LOGFILE_SIZE	NUMBER	Number of writes driven by the smallest redo log file size
WRITES_LOG_CHECKPOINT_SETTINGS	NUMBER	Number of writes driven by the LOG_CHECKPOINT_INTERVAL or LOG_CHECKPOINT_TIMEOUT initialization parameter
WRITES_OTHER_SETTINGS	NUMBER	Number of writes driven by other reasons (such as the deprecated FAST_START_IO_TARGET initialization parameter)
WRITES_AUTOTUNE	NUMBER	Number of writes due to auto-tune checkpointing
WRITES_FULL_THREAD_CKPT	NUMBER	Number of writes due to full thread checkpoints
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.13 V$IO_CALIBRATION_STATUS

V$IO_CALIBRATION_STATUS displays the status of I/O calibration in the instance.

Column	Datatype	Description
STATUS	VARCHAR2(13)	Calibration status: • IN PROGRESS - Calibration in Progress (Results from a previous calibration run are displayed, if available) • READY Results are ready and available from an earlier run • NOT AVAILABLE Calibration results are not available
CALIBRATION_TIME	TIMESTAMP(3)	End time of the last calibration run
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.14 V$IO_OUTLIER

V$IO_OUTLIER contains entries corresponding to I/Os that have taken a long time (more than 500 ms) to complete. Use this view to see if there any occasional delays in serving disk I/O requests by the storage subsystem.

Column	Datatype	Description
FUNCTION_NAME	VARCHAR2(18)	I/O function name of the delayed I/O
IO_SIZE	NUMBER	Size of the I/O in bytes
WAIT_EVENT	VARCHAR2(64)	Wait event name that was used to track the I/O
FILE_NAME	VARCHAR2(513)	Name of the file to which the I/O was targeted
IO_LATENCY	NUMBER	Time taken to complete the I/O (in milliseconds)
DISK1_NAME	VARCHAR2(255)	For Oracle ASM, the name of the first disk to which the I/O was issued
DISK1_LATENCY	NUMBER	Latency seen on the first disk (in milliseconds)
DISK2_NAME	VARCHAR2(255)	For Oracle ASM, the name of the second disk to which the I/O was issued
DISK2_LATENCY	NUMBER	Latency seen on the second disk (in milliseconds)
DISK3_NAME	VARCHAR2(255)	For Oracle ASM, the name of the third disk to which the I/O was issued
DISK3_LATENCY	NUMBER	Latency seen on the third disk (in milliseconds)
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

> **See Also:**
>
> - "V$KERNEL_IO_OUTLIER"
> - "V$LGWRIO_OUTLIER"

9.15 V$IOFUNCMETRIC

`V$IOFUNCMETRIC` displays I/O statistics information by database function for the most recent time interval period.

Column	Datatype	Description
`BEGIN_TIME`	`DATE`	Begin time of the interval
`END_TIME`	`DATE`	End time of the interval
`INTSIZE_CSEC`	`NUMBER`	Interval size (in hundredths of a second)
`FUNCTION_ID`	`NUMBER`	Function ID
`FUNCTION_NAME`	`VARCHAR2(18)`	Function name
`SMALL_READ_MBPS`	`NUMBER`	Single block megabytes read per second
`SMALL_WRITE_MBPS`	`NUMBER`	Single block megabytes written per second
`LARGE_READ_MBPS`	`NUMBER`	Multiblock megabytes read per second
`LARGE_WRITE_MBPS`	`NUMBER`	Multiblock megabytes written per second
`SMALL_READ_IOPS`	`NUMBER`	Single block read requests per second
`SMALL_WRITE_IOPS`	`NUMBER`	Single block write requests per second
`LARGE_READ_IOPS`	`NUMBER`	Multiblock read requests per second
`LARGE_WRITE_IOPS`	`NUMBER`	Multiblock write requests per second
`AVG_WAIT_TIME`	`NUMBER`	Average wait time (in milliseconds)
`CON_ID`	`NUMBER`	The ID of the container to which the data pertains. Possible values include: • `0`: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • `1`: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.16 V$IOFUNCMETRIC_HISTORY

`V$IOFUNCMETRIC_HISTORY` displays a recent history of the I/O statistics information by database function for the most recent time interval period.

The columns for `V$IOFUNCMETRIC_HISTORY` are the same as those for `V$IOFUNCMETRIC`.

> **See Also:**
>
> "V$IOFUNCMETRIC"

9.17 V$IOS_CLIENT

V$IOS_CLIENT provides more information about IO Server clients.

Column	Datatype	Description
CLIENT_ID	NUMBER	The unique ID for this client instance
CLUSTER_ID	VARCHAR2(33)	The GUID of the cluster where the client instance is running
CLUSTER_NAME	VARCHAR2(16)	When the query is executed on an Oracle IOServer (IOS) instance, this column shows the name of the cluster where the client (database) instance is running. When the query is executed on a database instance, this column shows the name of the cluster where the IOS instance that the database is connected to is running.
NODE	NUMBER	Number of the node within the cluster where the client instance is running
INSTANCE_NAME	VARCHAR2(64)	Instance name of the database client instance
DB_NAME	VARCHAR2(64)	Database name of the database client instance
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire multitenant container database (CDB). This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

> **Note:**
>
> In an Oracle Database instance, this view returns 0 rows when queried from a PDB.

9.18 V$IOSTAT_CONSUMER_GROUP

V$IOSTAT_CONSUMER_GROUP displays disk I/O statistics for consumer groups.

If the resource manager is enabled, then I/O statistics for all consumer groups that are part of the currently enabled resource plan are captured.

Column	Datatype	Description
CONSUMER_GROUP_ID	NUMBER	Consumer group ID
SMALL_READ_MEGABYTES	NUMBER	Number of single block megabytes read

Column	Datatype	Description
SMALL_WRITE_MEGABYTES	NUMBER	Number of single block megabytes written
LARGE_READ_MEGABYTES	NUMBER	Number of multiblock megabytes read
LARGE_WRITE_MEGABYTES	NUMBER	Number of multiblock megabytes written
SMALL_READ_REQS	NUMBER	Number of single block read requests
SMALL_WRITE_REQS	NUMBER	Number of single block write requests
LARGE_READ_REQS	NUMBER	Number of multiblock read requests
LARGE_WRITE_REQS	NUMBER	Number of multiblock write requests
NUMBER_OF_WAITS	NUMBER	Number of I/O waits by consumer group
WAIT_TIME	NUMBER	Total wait time (in milliseconds)
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.19 V$IOSTAT_FILE

V$IOSTAT_FILE displays information about disk I/O statistics of database files (including data files, temp files, and other types of database files).

I/O statistics for Data files and Temp files are provided for each file. All other file types (for example, control files, log files, archive logs, and so on) have their statistics consolidated into one entry in the view.

Column	Datatype	Description
FILE_NO	NUMBER	File identification number
FILETYPE_ID	NUMBER	Type of file (for example, log file, data file, and so on)
FILETYPE_NAME	VARCHAR2(28)	Name of the file, in the case of a data file or temp file. For all other files, a corresponding string to be displayed (for example, ARCHIVELOG).
SMALL_READ_MEGABYTES	NUMBER	Number of single block megabytes read
SMALL_WRITE_MEGABYTES	NUMBER	Number of single block megabytes written
LARGE_READ_MEGABYTES	NUMBER	Number of multiblock megabytes read
LARGE_WRITE_MEGABYTES	NUMBER	Number of multiblock megabytes written
SMALL_READ_REQS	NUMBER	Number of single block read requests
SMALL_WRITE_REQS	NUMBER	Number of single block write requests
SMALL_SYNC_READ_REQS	NUMBER	Number of synchronous single block read requests
LARGE_READ_REQS	NUMBER	Number of multiblock read requests
LARGE_WRITE_REQS	NUMBER	Number of multiblock write requests
SMALL_READ_SERVICETIME	NUMBER	Total service time (in milliseconds) for single block read requests
SMALL_WRITE_SERVICETIME	NUMBER	Total service time (in milliseconds) for single block write requests
SMALL_SYNC_READ_LATENCY	NUMBER	Latency for single block synchronous reads (in milliseconds)

Column	Datatype	Description
LARGE_READ_SERVICETIME	NUMBER	Total service time (in milliseconds) for multiblock read requests
LARGE_WRITE_SERVICETIME	NUMBER	Total service time (in milliseconds) for multiblock write requests
ASYNCH_IO	VARCHAR2(9)	Indicates whether asynchronous I/O is available for the file (ASYNC_ON) or not (ASYNC_OFF)
ACCESS_METHOD	VARCHAR2(11)	I/O library used to access the file. Possible values include: • OS_LIB - Operating system calls are used to access the file • ODM_LIB - Oracle Disk Manager library is used to access the file • ASM_MANAGED - The file is managed and accessed though ASM • DNFS_LIB - The file is accessed through Direct NFS library
RETRIES_ON_ERROR	NUMBER	Number of read retries on error
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.20 V$IOSTAT_FUNCTION

V$IOSTAT_FUNCTION displays disk I/O statistics for database functions (such as the LGWR and DBWR).

Column	Datatype	Description
FUNCTION_ID	NUMBER	Function ID
FUNCTION_NAME	VARCHAR2(18)	Function name: • RMAN • DBWR • LGWR • ARCH • XDB • Streams AQ • Data Pump • Recovery • Buffer Cache Reads • Direct Reads • Direct Writes • Smart Scan • Archive Manager • Others
SMALL_READ_MEGABYTES	NUMBER	Number of megabytes read via single block read requests
SMALL_WRITE_MEGABYTES	NUMBER	Number of megabytes written via single block write requests
LARGE_READ_MEGABYTES	NUMBER	Number of megabytes read via multiblock read requests
LARGE_WRITE_MEGABYTES	NUMBER	Number of megabytes written via multiblock write requests
SMALL_READ_REQS	NUMBER	Number of single block read requests

Column	Datatype	Description
SMALL_WRITE_REQS	NUMBER	Number of single block write requests
LARGE_READ_REQS	NUMBER	Number of multiblock read requests
LARGE_WRITE_REQS	NUMBER	Number of multiblock write requests
NUMBER_OF_WAITS	NUMBER	Number of synchronous I/O waits by functionality
WAIT_TIME	NUMBER	Total synchronous I/O wait time (in milliseconds)
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.21 V$IOSTAT_FUNCTION_DETAIL

V$IOSTAT_FUNCTION_DETAIL displays disk I/O statistics for database functions (such as the LGWR and DBWR), broken down by file type.

Column	Datatype	Description
FUNCTION_ID	NUMBER	Function ID
FUNCTION_NAME	VARCHAR2(18)	Function name: • RMAN • DBWR • LGWR • ARCH • XDB • Streams AQ • Data Pump • Recovery • Buffer Cache Reads • Direct Reads • Direct Writes • Smart Scan • Archive Manager • Others
FILETYPE_ID	NUMBER	File type ID

Column	Datatype	Description
FILETYPE_NAME	VARCHAR2(28)	File type name: • Control File • Data File • Log File • Archive Log • Temp File • Data File Backup • Data File Incremental Backup • Archive Log Backup • Data File Copy • Flashback Log • Data Pump Dump File • Other
SMALL_READ_MEGABYTES	NUMBER	Number of megabytes read via single block read requests
SMALL_WRITE_MEGABYTES	NUMBER	Number of megabytes written via single block write requests
LARGE_READ_MEGABYTES	NUMBER	Number of megabytes read via multiblock read requests
LARGE_WRITE_MEGABYTES	NUMBER	Number of megabytes written via multiblock write requests
SMALL_READ_REQS	NUMBER	Number of single block read requests
SMALL_WRITE_REQS	NUMBER	Number of single block write requests
LARGE_READ_REQS	NUMBER	Number of multiblock read requests
LARGE_WRITE_REQS	NUMBER	Number of multiblock write requests
NUMBER_OF_WAITS	NUMBER	Number of synchronous I/O waits by functionality
WAIT_TIME	NUMBER	Total synchronous I/O wait time (in milliseconds)
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.22 V$IOSTAT_NETWORK

V$IOSTAT_NETWORK displays information about network I/O statistics that were caused by accessing files on a remote database instance.

Column	Datatype	Description
CLIENT	VARCHAR2(32)	Database client name initiating the network I/O (for example, RMAN or PL/SQL)
READS#	NUMBER	Number of read operations issued
WRITES#	NUMBER	Number of write operations issued
KBYTES_READ	NUMBER	Total number of kilobytes read
KBYTES_WRITTEN	NUMBER	Total number of kilobytes written
READ_LATENCY	NUMBER	Total read wait time (in milliseconds)

Column	Datatype	Description
WRITE_LATENCY	NUMBER	Total write wait time (in milliseconds)
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.23 V$IP_ACL

V$IP_ACL provides information about access control to database services from network hosts.

Column	Datatype	Description
SERVICE_NAME	VARCHAR2(512)	The database service name
HOST	VARCHAR2(64)	The host being granted access to the SERVICE_NAME. The host can be a hostname, dotted-decimal IPv4 or heximal IPv6 address. Wildcard "*" for IPv4 and Classless Inter-Domain Routing (CIDR) format is allowed.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs • 1: This value is used for rows containing data that pertain only to the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.24 V$JAVA_LIBRARY_CACHE_MEMORY

V$JAVA_LIBRARY_CACHE_MEMORY displays information about memory allocated to library cache memory objects in different namespaces for Java objects.

A memory object is an internal grouping of memory for efficient management. A library cache object may consist of one or more memory objects.

Column	Datatype	Description
LC_NAMESPACE	VARCHAR2(15)	Library cache namespace
LC_INUSE_MEMORY_OBJECTS	NUMBER	Number of library cache memory objects currently in use in the Java pool
LC_INUSE_MEMORY_SIZE	NUMBER	Total size of library cache in-use memory objects (in megabytes)
LC_FREEABLE_MEMORY_OBJECTS	NUMBER	Number of freeable library cache memory objects in the Java pool
LC_FREEABLE_MEMORY_SIZE	NUMBER	Size of library cache freeable memory objects (in megabytes)

Column	Datatype	Description
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.25 V$JAVA_POOL_ADVICE

V$JAVA_POOL_ADVICE displays information about estimated parse time in the Java pool for different pool sizes.

The sizes range from 10% of the current Java pool size or the amount of pinned Java library cache memory (whichever is higher) to 200% of the current Java pool size, in equal intervals. The value of the interval depends on the current size of the Java pool.

Parse time saved refers to the amount of time saved by keeping library cache memory objects in the Java pool, as opposed to having to reload these objects.

Column	Datatype	Description
JAVA_POOL_SIZE_FOR_ESTIMATE	NUMBER	Java pool size for the estimate (in megabytes)
JAVA_POOL_SIZE_FACTOR	NUMBER	Size factor with respect to the current Java pool size
ESTD_LC_SIZE	NUMBER	Estimated memory in use by the library cache (in megabytes)
ESTD_LC_MEMORY_OBJECTS	NUMBER	Estimated number of library cache memory objects in the Java pool of the specified size
ESTD_LC_TIME_SAVED	NUMBER	Estimated elapsed parse time saved (in seconds), owing to library cache memory objects being found in a Java pool of the specified size. This is the time that would have been spent in reloading the required objects in the Java pool had they been aged out due to insufficient amount of available free memory.
ESTD_LC_TIME_SAVED_FACTOR	NUMBER	Estimated parse time saved factor with respect to the current Java pool size
ESTD_LC_LOAD_TIME	NUMBER	Estimated elapsed time (in seconds) for parsing in a Java pool of the specified size
ESTD_LC_LOAD_TIME_FACTOR	NUMBER	Estimated load time factor with respect to the current Java pool size
ESTD_LC_MEMORY_OBJECT_HITS	NUMBER	Estimated number of times a library cache memory object was found in a Java pool of the specified size
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.26 V$KERNEL_IO_OUTLIER

V$KERNEL_IO_OUTLIER contains entries corresponding to I/Os that have taken a long time (more than 500 ms) to complete.

Use this view to see the individual kernel components of I/Os for which there are any occasional delays in serving disk I/O requests by the storage subsystem.

> **Note:**
>
> Although this view exists on all platforms in Oracle Database 12*c*, it is only populated on the Solaris platform.

Column	Datatype	Description
TIMESTAMP	NUMBER	Number of seconds elapsed since 00:00 UTC, January 1, 1970
IO_SIZE	NUMBER	Size of the I/O, in KB.
IO_OFFSET	NUMBER	Offset into the device of the I/O
DEVICE_NAME	VARCHAR2(513)	Name of the device to which the I/O was targeted
PROCESS_NAME	VARCHAR2(64)	Name of the process that issued the I/O
TOTAL_LATENCY	NUMBER	Total time the I/O spent in the kernel (in milliseconds)
SETUP_LATENCY	NUMBER	Time spent during initial I/O setup before sending to SCSI target device driver (in milliseconds)
QUEUE_TO_HBA_LATENCY	NUMBER	Time spent in the SCSI target device driver before being sent to the Host Bus Adaptor (in milliseconds)
TRANSFER_LATENCY	NUMBER	Time spent in the Host Bus Adaptor and physically transferring the I/O to the storage device (in milliseconds)
CLEANUP_LATENCY	NUMBER	Time spent freeing resources used by the completed I/O (in milliseconds)
PID	NUMBER	Process ID that issued the I/O
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

> **See Also:**
>
> - "V$IO_OUTLIER"
> - "V$LGWRIO_OUTLIER"

9.27 V$KEY_VECTOR

V$KEY_VECTOR provides debugging information related to the data structures used by in-memory aggregation for current and recent queries using key vectors.

Column	Datatype	Description
SESSION_ID	NUMBER	Session ID
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data
TRANSLATE_ID	NUMBER	Translation vector ID
SQL_ID	VARCHAR2(13)	SQL ID that uses the translation vector
SQL_EXEC_START	DATE	Time when the execution of the SQL started
SQL_EXEC_ID	NUMBER	SQL execution identifier
PROCESS	VARCHAR2(64)	Operating system client process ID
STATE	VARCHAR2(9)	State of the in-memory aggregation operation: • BYPASS Refers to key vectors that are currently in use by queries that have undergone the vector transform. • BYPASSED This is a historical entry for key vectors whose queries have already finished. • FINISHED This is a historical entry for key vectors whose queries have already finished. • RUNNING Refers to key vectors that are currently in use by queries that have undergone the vector transform. One query will probably have multiple key vectors if it has gone through the vector transform. Also, if the query is operating in parallel, there may be multiple entries per PX server. The PROCESS column can be joined to V$SESSION to differentiate these cases.
TYPE	VARCHAR2(9)	Type of translation vector created: • DOUBLEIND • PAGED • SIMPLE • INDIRECT • HASH • OFFSET

Column	Datatype	Description
WIDTH	NUMBER	Width (number of bits) of DOUBLEIND, SIMPLE, INDIRECT, or OFFSETvector translation array: • 1 • 4 • 8 • 16 • 32
KEY_DATA_TYPE	VARCHAR2(13)	The internal join column key data type of either DOUBLEIND, SIMPLE, INDIRECT, or OFFSET translation vectors: • BINARY • BINARY_FLOAT • NUMBER • PACKED_BINARY • PACKED_DATE • PACKED_NUMBER Also, a value of NULL is displayed for HASH translation vectors with non-numeric data types and for HASH translation vectors with more than one join key column.
JOIN_COLUMN_COUNT	NUMBER	Number of join columns. PAGED and HASHED are the only types which can have greater than one join column.
JOIN_KEY_COUNT	NUMBER	Number of join key values used to build translation vector
DUP_JOIN_KEY_COUNT	NUMBER	Indicates the number of key values in the key vector that have more than one parent value
MIN_JOIN_KEY	NUMBER	Min join key value in translation vector
MAX_JOIN_KEY	NUMBER	Max join key value in translation vector
GROUP_KEY_COUNT	NUMBER	Max dense grouping key value. This value is computed while the data that creates the translation vector is processed.
FILTERED	NUMBER	Number of rows filtered by translation vector
PROBED	NUMBER	Number of rows that probed the translation vector in key vector use row source
ACTIVE	NUMBER	Number of active translation vectors used as filters across all workers
DISABLED	NUMBER	Number of translation vector filters across all workers that were disabled
MEMORY_ALLOCATED	NUMBER	Amount of memory allocated for the key vector
JOIN_STRUCTURE_SIZE	NUMBER	Amount of memory used out of the allocated space
FACT_OWNER	VARCHAR2(128)	The owner of the fact table. Null when the fact table is null.
FACT_NAME	VARCHAR2(128)	The table that contains measure data. May be null if more than one fact table is used in the query.
DIMENSION_OWNER	VARCHAR2(128)	When a single table is used to construct the key vector, the table owner will appear in this column. If multiple tables are joined to serve as the dimension (more of a snowflake than a star schema shape, for one example), then this column will be null.
DIMENSION_NAME	VARCHAR2(128)	The table that contains attribute data and is joined to the fact table. May be null if two or more tables are used (for example, snowflake style dimension tables).
CREATION_DURATION	NUMBER	Records the total time in seconds that it took to create the key vector
PAYLOAD_COLUMN_COUNT	NUMBER	The number of columns being carried from the dimension table to the fact table scan for processing without joinback

Column	Datatype	Description
OFFLOADED	VARCHAR2(10)	Indicates whether and how the key vector has been sent to an Exadata cell. Possible values: • NO: The key vector has not been sent to an Exadata cell. • YES: The key vector was fully sent to an Exadata cell. • COMPRESSED: The key vector was fully sent to an Exadata cell and was compressed. • EFILTER: The key vector could not be sent to an Exadata cell due to the size of another constraint. Instead, a bloom filter-like data structure was sent to aid filtration.
COMPRESSED_SIZE	NUMBER	If the value of OFFLOADED is COMPRESSED, then this column displays the compressed size of metadata transmitted to the Exadata cell (in bytes)
EFILTER_SIZE	NUMBER	If the value of OFFLOADED is EFILTER, then this column displays the size of the data structure sent to the Exadata cell (in bytes)

See Also:

Oracle Database SQL Tuning Guide for more information about in-memory aggregation

9.28 V$LATCH

V$LATCH displays aggregate latch statistics for both parent and child latches, grouped by latch name.

Individual parent and child latch statistics are broken down in the views V$LATCH_PARENT and V$LATCH_CHILDREN.

Column	Datatype	Description
ADDR	RAW(4 | 8)	Address of the latch object
LATCH#	NUMBER	Latch number
LEVEL#	NUMBER	Latch level
NAME	VARCHAR2(64)	Latch name
HASH	NUMBER	Latch hash
GETS	NUMBER	Number of times the latch was requested in willing-to-wait mode
MISSES	NUMBER	Number of times the latch was requested in willing-to-wait mode and the requester had to wait
SLEEPS	NUMBER	Number of times a willing-to-wait latch request resulted in a session sleeping while waiting for the latch
IMMEDIATE_GETS	NUMBER	Number of times a latch was requested in no-wait mode
IMMEDIATE_MISSES	NUMBER	Number of times a no-wait latch request did not succeed (that is, missed)
WAITERS_WOKEN	NUMBER	This column has been deprecated and is present only for compatibility with previous releases of Oracle. No data is accumulated for this column; it will always have a value of zero.

Column	Datatype	Description
WAITS_HOLDING_LATCH	NUMBER	This column has been deprecated and is present only for compatibility with previous releases of Oracle. No data is accumulated for this column; it will always have a value of zero.
SPIN_GETS	NUMBER	Willing-to-wait latch requests which missed the first try but succeeded while spinning
SLEEP[1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11]	NUMBER	These columns have been deprecated and are present only for compatibility with previous releases of Oracle. No data is accumulated for these columns; they will always have a value of zero. As a substitute for these columns you can query the appropriate rows of the V$EVENT_HISTOGRAM view where the EVENT column has a value of latch free or latch:%.
WAIT_TIME	NUMBER	Elapsed time spent waiting for the latch (in microseconds)
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

See Also:

- "V$LATCH_CHILDREN"
- "V$LATCH_PARENT"

9.29 V$LATCH_CHILDREN

V$LATCH_CHILDREN displays statistics about child latches.

This view includes all columns of V$LATCH plus the CHILD# column. Note that child latches have the same parent if their LATCH# columns match each other.

Column	Datatype	Description
ADDR	RAW(4 | 8)	Address of the latch object
LATCH#	NUMBER	Latch number of the parent latch
CHILD#	NUMBER	Child latch number (unique only to each parent latch)
LEVEL#	NUMBER	Latch level
NAME	VARCHAR2(64)	Latch name
HASH	NUMBER	Latch hash
GETS	NUMBER	Number of times the latch was requested in willing-to-wait mode
MISSES	NUMBER	Number of times the latch was requested in willing-to-wait mode and the requester had to wait
SLEEPS	NUMBER	Number of times a willing-to-wait latch request resulted in a session sleeping while waiting for the latch

Column	Datatype	Description
IMMEDIATE_GETS	NUMBER	Number of times a latch was requested in no-wait mode
IMMEDIATE_MISSES	NUMBER	Number of times a no-wait latch request did not succeed (that is, missed)
WAITERS_WOKEN	NUMBER	This column has been deprecated and is present only for compatibility with previous releases of Oracle. No data is accumulated for this column; it will always have a value of zero.
WAITS_HOLDING_LATCH	NUMBER	This column has been deprecated and is present only for compatibility with previous releases of Oracle. No data is accumulated for this column; it will always have a value of zero.
SPIN_GETS	NUMBER	Willing-to-wait latch requests which missed the first try but succeeded while spinning
SLEEP[1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11]	NUMBER	These columns have been deprecated and are present only for compatibility with previous releases of Oracle. No data is accumulated for these columns; they will always have a value of zero. As a substitute for these columns you can query the appropriate rows of the V$EVENT_HISTOGRAM view where the EVENT column has a value of latch free or latch:%.
WAIT_TIME	NUMBER	Elapsed time spent waiting for the latch (in microseconds)
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

See Also:

"V$LATCH"

9.30 V$LATCH_MISSES

V$LATCH_MISSES displays statistics about missed attempts to acquire a latch.

Column	Datatype	Description
PARENT_NAME	VARCHAR2(64)	Latch name of a parent latch
WHERE	VARCHAR2(80)	This column is obsolete and maintained for backward compatibility. The value of this column is always equal to the value in LOCATION.
NWFAIL_COUNT	NUMBER	Number of times that no-wait acquisition of the latch failed
SLEEP_COUNT	NUMBER	Number of times that acquisition attempts caused sleeps
WTR_SLP_COUNT	NUMBER	Number of times a waiter slept
LONGHOLD_COUNT	NUMBER	Number of times someone held a latch for the entire duration of someone else's sleep
LOCATION	VARCHAR2(80)	Location that attempted to acquire the latch

Column	Datatype	Description
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.31 V$LATCH_PARENT

V$LATCH_PARENT displays statistics about parent latches.

\ The columns for V$LATCH_PARENT are the same as those for V$LATCH.

> **See Also:**
>
> "V$LATCH"

9.32 V$LATCHHOLDER

V$LATCHHOLDER displays information about the current latch holders.

Column	Datatype	Description
PID	NUMBER	Identifier of the process holding the latch
SID	NUMBER	Identifier of the session that owns the latch
LADDR	RAW(4 | 8)	Latch address
NAME	VARCHAR2(64)	Name of the latch being held
GETS	NUMBER	Number of times that the latch was obtained in either wait mode or no-wait mode
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.33 V$LATCHNAME

V$LATCHNAME displays information about decoded latch names for the latches shown in V$LATCH.

The rows of V$LATCHNAME have a one-to-one correspondence to the rows of V$LATCH.

Column	Datatype	Description
LATCH#	NUMBER	Latch number
NAME	VARCHAR2(64)	Latch name
DISPLAY_NAME	VARCHAR2(64)	A clearer and more descriptive name for the latch that appears in the NAME column. Names that appear in the DISPLAY_NAME column can change across Oracle Database releases, therefore customer scripts should not rely on names that appear in the DISPLAY_NAME column across releases.
HASH	NUMBER	Latch hash
TYPE	VARCHAR2(4)	Type of the latch (SGA, PDB, or OSP):
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

See Also:

"V$LATCH"

9.34 V$LCR_CACHE

V$LCR_CACHE displays information about entries in the logical change record (LCR) cache.

Column	Datatype	Description
LCR_TYPE	CHAR(12)	Indicates whether the LCR is INTERNAL or EXTERNAL
COMPONENT_NAME	VARCHAR2(128)	Component name
COMPONENT_TYPE	VARCHAR2(40)	Component type. Possible values: • APPLY COORDINATOR • APPLY READER • APPLY SERVER • CAPTURE COORDINATOR • CAPTURE READER • CAPTURE SERVER
NUM_COLUMNS	NUMBER	Number of columns in the LCR
FREE_COUNT	NUMBER	Number of available LCRs
REQUEST_COUNT	NUMBER	Total request count
CACHE_HITS	NUMBER	Cache hit count
ALT_CACHE_HITS	NUMBER	Hit count from an alternate LCR cache entry
PURGED_COUNT	NUMBER	Cache purge count
SID	NUMBER	Session ID of the component process

Column	Datatype	Description
SERIAL#	NUMBER	Session serial number of the component process
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.35 V$LGWRIO_OUTLIER

V$LGWRIO_OUTLIER contains entries corresponding to Log Writer (LGWR) process I/Os that have taken a long time (more than 500 ms) to complete. Use this view to see if there any occasional delays in serving disk I/O requests by the storage subsystem.

Column	Datatype	Description
FUNCTION_NAME	VARCHAR2(18)	I/O function name of the delayed I/O
IO_SIZE	NUMBER	Size of the I/O in bytes
WAIT_EVENT	VARCHAR2(64)	Wait event name that was used to track the I/O
FILE_NAME	VARCHAR2(513)	Name of the file to which the I/O was targeted
IO_LATENCY	NUMBER	Time taken to complete the I/O (in milliseconds)
DISK1_NAME	VARCHAR2(255)	For Oracle ASM, the name of the first disk to which the I/O was issued
DISK1_LATENCY	NUMBER	Latency seen on the first disk (in milliseconds)
DISK2_NAME	VARCHAR2(255)	For Oracle ASM, the name of the second disk to which the I/O was issued
DISK2_LATENCY	NUMBER	Latency seen on the second disk (in milliseconds)
DISK3_NAME	VARCHAR2(255)	For Oracle ASM, the name of the third disk to which the I/O was issued
DISK3_LATENCY	NUMBER	Latency seen on the third disk (in milliseconds)
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

See Also:

- "V$IO_OUTLIER"
- "V$KERNEL_IO_OUTLIER"

9.36 V$LIBCACHE_LOCKS

V$LIBCACHE_LOCKS displays information about the library cache locks and pins. Locks and pins are distinguished based on the value of the TYPE column.

Column	Datatype	Description
TYPE	VARCHAR2(4)	LOCK or PIN
ADDR	RAW(4 | 8)	Address of the lock/pin
HOLDING_USER_SESSION	RAW(4 | 8)	User session holding this lock/pin
HOLDING_SESSION	RAW(4 | 8)	Session holding this lock/pin
OBJECT_HANDLE	RAW(4 | 8)	Handle address for which the lock/pin is acquired
LOCK_HELD	RAW(4 | 8)	If the type is LOCK, then LOCK_HELD represents the pin that is pinning the object. If the type is PIN, then LOCK_HELD represents the lock that is locking the object.
REFCOUNT	NUMBER	Reference count for this lock/pin
MODE_HELD	NUMBER	Lock/pin mode held: • 0 - No lock/pin held • 1 - Null mode • 2 - Share mode • 3 - Exclusive mode
MODE_REQUESTED	NUMBER	Lock/pin mode requested: • 0 - No lock/pin requested • 1 - Null mode • 2 - Share mode • 3 - Exclusive mode
SAVEPOINT_NUMBER	NUMBER	Kernel transaction savepoint number at the time the lock/pin was acquired
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.37 V$LIBRARY_CACHE_MEMORY

V$LIBRARY_CACHE_MEMORY displays information about memory allocated to library cache memory objects in different namespaces.

A memory object is an internal grouping of memory for efficient management. A library cache object may consist of one or more memory objects.

Column	Datatype	Description
LC_NAMESPACE	VARCHAR2(15)	Library cache namespace

Column	Datatype	Description
LC_INUSE_MEMORY_OBJECTS	NUMBER	Number of library cache memory objects currently in use in the shared pool
LC_INUSE_MEMORY_SIZE	NUMBER	Total size of library cache in-use memory objects (in megabytes)
LC_FREEABLE_MEMORY_OBJECTS	NUMBER	Number of freeable library cache memory objects in the shared pool
LC_FREEABLE_MEMORY_SIZE	NUMBER	Size of library cache freeable memory objects (in megabytes)
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.38 V$LIBRARYCACHE

V$LIBRARYCACHE contains statistics about library cache performance and activity.

Column	Datatype	Description
NAMESPACE	VARCHAR2(64)	Library cache namespace
GETS	NUMBER	Number of times a lock was requested for objects of this namespace
GETHITS	NUMBER	Number of times an object's handle was found in memory
GETHITRATIO	NUMBER	Ratio of GETHITS to GETS
PINS	NUMBER	Number of times a PIN was requested for objects of this namespace
PINHITS	NUMBER	Number of times all of the metadata pieces of the library object were found in memory
PINHITRATIO	NUMBER	Ratio of PINHITS to PINS
RELOADS	NUMBER	Any PIN of an object that is not the first PIN performed since the object handle was created, and which requires loading the object from disk
INVALIDATIONS	NUMBER	Total number of times objects in this namespace were marked invalid because a dependent object was modified
DLM_LOCK_REQUESTS	NUMBER	Number of GET requests lock instance locks
DLM_PIN_REQUESTS	NUMBER	Number of PIN requests lock instance locks
DLM_PIN_RELEASES	NUMBER	Number of release requests PIN instance locks
DLM_INVALIDATION_REQUESTS	NUMBER	Number of GET requests for invalidation instance locks
DLM_INVALIDATIONS	NUMBER	Number of invalidation pings received from other instances
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.39 V$LICENSE

V$LICENSE displays information about license limits.

Column	Datatype	Description
SESSIONS_MAX	NUMBER	Maximum number of concurrent user sessions allowed for the instance
SESSIONS_WARNING	NUMBER	Warning limit for concurrent user sessions for the instance
SESSIONS_CURRENT	NUMBER	Current number of concurrent user sessions
SESSIONS_HIGHWATER	NUMBER	Highest number of concurrent user sessions since the instance started
USERS_MAX	NUMBER	Maximum number of named users allowed for the database
CPU_COUNT_CURRENT	NUMBER	Current number of logical CPUs or processors on the system
CPU_CORE_COUNT_CURRENT	NUMBER	Current number of CPU cores on the system (includes subcores of multicore CPUs, as well as single-core CPUs)
CPU_SOCKET_COUNT_CURRENT	NUMBER	Current number of CPU sockets on the system (represents an absolute count of CPU chips on the system, regardless of multithreading or multicore architectures)
CPU_COUNT_HIGHWATER	NUMBER	Highest number of logical CPUs or processors on the system since the instance started
CPU_CORE_COUNT_HIGHWATER	NUMBER	Highest number of CPU cores on the system since the instance started (includes subcores of multicore CPUs, as well as single-core CPUs)
CPU_SOCKET_COUNT_HIGHWATER	NUMBER	Highest number of CPU sockets on the system since the instance started (represents an absolute count of CPU chips on the system, regardless of multithreading or multicore architectures)
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

> **Note:**
>
> The availability of the CPU core count and CPU socket count statistics is subject to the operating system platform on which the Oracle Database is running. If a statistic is unavailable, the view will return NULL for the statistic value.

9.40 V$LOADISTAT

V$LOADISTAT contains errors that occurred when updating indexes on a table during a load using the Direct Path API.

Column	Datatype	Description
OWNER	VARCHAR2(31)	Schema name

Column	Datatype	Description
TABNAME	VARCHAR2(31)	Table name
INDEXNAME	VARCHAR2(31)	Index name
SUBNAME	VARCHAR2(31)	Index sub name
MESSAGE_NUM	NUMBER	Error message number
MESSAGE	VARCHAR2(4000)	Error message
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.41 V$LOADPSTAT

V$LOADPSTAT contains statistics about the number of rows loaded into a partition, or subpartition, during a load using the Direct Path API.

Column	Datatype	Description
OWNER	VARCHAR2(31)	Schema name
TABNAME	VARCHAR2(31)	Table name
PARTNAME	VARCHAR2(31)	Partition name
LOADED	NUMBER	Number of rows loaded
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.42 V$LOCK

V$LOCK lists the locks currently held by the Oracle Database and outstanding requests for a lock or latch.

Column	Datatype	Description
ADDR	RAW(4 | 8)	Address of lock state object
KADDR	RAW(4 | 8)	Address of lock
SID	NUMBER	Identifier for session holding or acquiring the lock

Column	Datatype	Description
TYPE	VARCHAR2(2)	Type of user or system lock The locks on the user types are obtained by user applications. Any process that is blocking others is likely to be holding one of these locks. The user type locks are: TM - DML enqueue TX - Transaction enqueue UL - User supplied The system type locks are listed in Table 9-1. Be aware that not all types of locks are documented. To find a complete list of locks for the current release, query the V$LOCK_TYPE data dictionary view, described on "V$LOCK_TYPE".
ID1	NUMBER	Lock identifier #1 (depends on type)
ID2	NUMBER	Lock identifier #2 (depends on type)
LMODE	NUMBER	Lock mode in which the session holds the lock: • 0 - none • 1 - null (NULL) • 2 - row-S (SS) • 3 - row-X (SX) • 4 - share (S) • 5 - S/Row-X (SSX) • 6 - exclusive (X)
REQUEST	NUMBER	Lock mode in which the process requests the lock: • 0 - none • 1 - null (NULL) • 2 - row-S (SS) • 3 - row-X (SX) • 4 - share (S) • 5 - S/Row-X (SSX) • 6 - exclusive (X)
CTIME	NUMBER	Time since current mode was granted
BLOCK	NUMBER	Indicates whether the lock in question is blocking other processes. The possible values are: • 0 - The lock is not blocking any other processes • 1 - The lock is blocking other processes • 2 - The lock is not blocking any blocked processes on the local node, but it may or may not be blocking processes on remote nodes. This value is used only in Oracle Real Application Clusters (Oracle RAC) configurations (not in single instance configurations).
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

Table 9-1 Values for the TYPE Column: System Types

System Type	Description	System Type	Description
AE	Edition enqueue	MR	Media recovery
AT	Lock held for the ALTER TABLE statement	NA..NZ	Library cache pin instance (A..Z = namespace)
BL	Buffer hash table instance	PF	Password File
CF	Control file schema global enqueue	PI, PS	Parallel operation
CI	Cross-instance function invocation instance	PR	Process startup
CU	Cursor bind	QA..QZ	Row cache instance (A..Z = cache)
DF	datafile instance	RT	Redo thread global enqueue
DL	Direct loader parallel index create	SC	System change number instance
DM	Mount/startup db primary/secondary instance	SM	SMON
DR	Distributed recovery process	SN	Sequence number instance
DX	Distributed transaction entry	SQ	Sequence number enqueue
FS	File set	SS	Sort segment
HW	Space management operations on a specific segment	ST	Space transaction enqueue
IN	Instance number	SV	Sequence number value
IR	Instance recovery serialization global enqueue	TA	Generic enqueue
IS	Instance state	TS	Temporary segment enqueue (ID2=0)
IV	Library cache invalidation instance	TS	New block allocation enqueue (ID2=1)
JQ	Job queue	TT	Temporary table enqueue
KK	Thread kick	UN	User name
LA .. LP	Library cache lock instance lock (A..P = namespace)	US	Undo segment DDL
MM	Mount definition global enqueue	WL	Being-written redo log instance

9.43 V$LOCK_ACTIVITY

V$LOCK_ACTIVITY is deprecated. The information that was provided in this view is now provided in the V$INSTANCE_CACHE_TRANSFER and V$SEGMENT_STATISTICS views.

Column	Datatype	Description
FROM_VAL	CHAR(4)	Global Cache Resource initial state; always NULL
TO_VAL	CHAR(1)	Global Cache Resource initial state; always S
ACTION_VAL	CHAR(21)	Description of the conversion; always Lock buffers for read
COUNTER	NUMBER	Number of times the lock operation executed

Column	Datatype	Description
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

See Also:

- "V$INSTANCE_CACHE_TRANSFER"
- "V$SEGMENT_STATISTICS"

9.44 V$LOCK_TYPE

V$LOCK_TYPE describes the type of locks available.

Column	Datatype	Description
TYPE	VARCHAR2(64)	A two-letter internal resource identifier
NAME	VARCHAR2(64)	Resource type name. This is a short (less than 32 characters) enqueue type name.
ID1_TAG	VARCHAR2(64)	Description of the enqueue type.
ID2_TAG	VARCHAR2(64)	Further description of the enqueue type.
IS_USER	VARCHAR2(3)	User enqueue. These are enqueues that are acquired as a direct result of a SQL statement. Applications may get deadlocks on these enqueues. Such deadlocks are considered application errors.
IS_RECYCLE	VARCHAR2(3)	Indicates whether the enqueue requires caching of the associated resource in the DLM's resource cache. Possible values: • NO: Means the enqueue requires caching of the associated resource in the DLM's resource cache • YES: Means the enqueue does not require caching of the associated resource in the DLM's resource cache
DESCRIPTION	VARCHAR2(4000)	Explanation of how or for what purpose the enqueue is used.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.45 V$LOCKDOWN_RULES

V$LOCKDOWN_RULES displays information about lockdown profile rules that are applicable in the pluggable database (PDB) where this view is queried.

Column	Datatype	Description
RULE_TYPE	VARCHAR2(128)	Type of the rule: • STATEMENT • FEATURE • OPTION
RULE	VARCHAR2(128)	Rule to be enabled or disabled
CLAUSE	VARCHAR2(128)	Clause of the statement
CLAUSE_OPTION	VARCHAR2(128)	Option of the clause
STATUS	VARCHAR2(7)	Status of the lockdown profile: • ENABLE • DISABLE • EMPTY
USERS	VARCHAR2(6)	Type of users affected by the rule: • ALL • LOCAL • COMMON
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs • 1: This value is used for rows containing data that pertain to only the root. • *n*: Where *n* is the applicable container ID for the rows containing data

See Also:

Oracle Database Security Guide for more information about PDB lockdown profiles

9.46 V$LOCKED_OBJECT

V$LOCKED_OBJECT lists all locks acquired by every transaction on the system. It shows which sessions are holding DML locks (that is, TM-type enqueues) on what objects and in what mode.

Column	Datatype	Description
XIDUSN	NUMBER	Undo segment number
XIDSLOT	NUMBER	Slot number
XIDSQN	NUMBER	Sequence number

Column	Datatype	Description
OBJECT_ID	NUMBER	Object ID being locked
SESSION_ID	NUMBER	Session ID
ORACLE_USERNAME	VARCHAR2(128)	Oracle user name
OS_USER_NAME	VARCHAR2(128)	Operating system user name
PROCESS	VARCHAR2(24)	Operating system process ID
LOCKED_MODE	NUMBER	Lock mode. The numeric values for this column map to these text values for the lock modes for table locks: • 0 - NONE: lock requested but not yet obtained • 1 - NULL • 2 - ROWS_S (SS): Row Share Lock • 3 - ROW_X (SX): Row Exclusive Table Lock • 4 - SHARE (S): Share Table Lock • 5 - S/ROW-X (SSX): Share Row Exclusive Table Lock • 6 - Exclusive (X): Exclusive Table Lock **See Also:** *Oracle Database Concepts* for more information about lock modes for table locks
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.47 V$LOG

V$LOG displays log file information from the control file.

Column	Datatype	Description
GROUP#	NUMBER	Log group number
THREAD#	NUMBER	Log thread number
SEQUENCE#	NUMBER	Log sequence number
BYTES	NUMBER	Size of the log (in bytes)
BLOCKSIZE	NUMBER	Block size of the logfile (512 or 4096)
MEMBERS	NUMBER	Number of members in the log group
ARCHIVED	VARCHAR2(3)	Archive status (YES) or (NO)

Column	Datatype	Description
STATUS	VARCHAR2(16)	Log status: • UNUSED - Online redo log has never been written to. This is the state of a redo log that was just added, or just after a RESETLOGS, when it is not the current redo log. • CURRENT - Current redo log. This implies that the redo log is active. The redo log could be open or closed. • ACTIVE - Log is active but is not the current log. It is needed for crash recovery. It may be in use for block recovery. It may or may not be archived. • CLEARING - Log is being re-created as an empty log after an ALTER DATABASE CLEAR LOGFILE statement. After the log is cleared, the status changes to UNUSED. • CLEARING_CURRENT - Current log is being cleared of a closed thread. The log can stay in this status if there is some failure in the switch such as an I/O error writing the new log header. • INACTIVE - Log is no longer needed for instance recovery. It may be in use for media recovery. It may or may not be archived.
FIRST_CHANGE#	NUMBER	Lowest system change number (SCN) in the log
FIRST_TIME	DATE	Time of the first SCN in the log
NEXT_CHANGE#	NUMBER	Highest change number (SCN) in the log When STATUS=CURRENT, NEXT_CHANGE# is set to the highest possible SCN, 9295429630892703743.
NEXT_TIME	DATE	Time of the highest SCN in the log. When STATUS=CURRENT, NEXT_TIME is set to NULL.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.48 V$LOG_HISTORY

V$LOG_HISTORY displays log history information from the control file.

Column	Datatype	Description
RECID	NUMBER	Control file record ID
STAMP	NUMBER	Control file record stamp
THREAD#	NUMBER	Thread number of the archived log
SEQUENCE#	NUMBER	Sequence number of the archived log
FIRST_CHANGE#	NUMBER	Lowest system change number (SCN) in the log
FIRST_TIME	DATE	Time of the first entry (lowest SCN) in the log
NEXT_CHANGE#	NUMBER	Highest SCN in the log
RESETLOGS_CHANGE#	NUMBER	Resetlogs change number of the database when the log was written
RESETLOGS_TIME	DATE	Resetlogs time of the database when the log was written

Column	Datatype	Description
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.49 V$LOGFILE

V$LOGFILE contains information about redo log files.

Column	Datatype	Description
GROUP#	NUMBER	Redo log group identifier number
STATUS	VARCHAR2(7)	Status of the log member: • INVALID - File is inaccessible • STALE - File's contents are incomplete • DELETED - File is no longer used • null - File is in use
TYPE	VARCHAR2(7)	Type of the logfile: • ONLINE • STANDBY
MEMBER	VARCHAR2(513)	Redo log member name
IS_RECOVERY_DEST_FILE	VARCHAR2(3)	Indicates whether the file was created in the fast recovery area (YES) or not (NO)
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.50 V$LOGHIST

V$LOGHIST contains log history information from the control file. This view is retained for historical compatibility. Oracle recommends that you use V$LOG_HISTORY instead.

> **See Also:**
>
> "V$LOG_HISTORY"

Column	Datatype	Description
THREAD#	NUMBER	Log thread number
SEQUENCE#	NUMBER	Log sequence number
FIRST_CHANGE#	NUMBER	Lowest SCN in the log
FIRST_TIME	DATE	Time of first SCN in the log
SWITCH_CHANGE#	NUMBER	SCN at which the log switch occurred; one more than highest SCN in the log
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.51 V$LOGMNR_CONTENTS

V$LOGMNR_CONTENTS contains log history information. To query this view, you must have the LOGMINING privilege.

When a SELECT statement is executed against the V$LOGMNR_CONTENTS view, the archive redo log files are read sequentially. Translated records from the redo log files are returned as rows in the V$LOGMNR_CONTENTS view. This continues until either the filter criteria specified at startup (EndTime or endScn) are met or the end of the archive log file is reached.

When this view is queried from a PDB, it returns only redo generated by that PDB.

Column	Datatype	Description
SCN	NUMBER	System change number (SCN) when the database change was made
START_SCN	NUMBER	System change number (SCN) when the transaction that contains this change started; only meaningful if the COMMITTED_DATA_ONLY option was chosen in a DBMS_LOGMNR.START_LOGMNR() invocation, NULL otherwise. This column may also be NULL if the query is run over a time/SCN range that does not contain the start of the transaction.
COMMIT_SCN	NUMBER	System change number (SCN) when the transaction committed; only meaningful if the COMMITTED_DATA_ONLY option was chosen in a DBMS_LOGMNR.START_LOGMNR() invocation
TIMESTAMP	DATE	Timestamp when the database change was made
START_TIMESTAMP	DATE	Timestamp when the transaction that contains this change started; only meaningful if the COMMITTED_DATA_ONLY option was chosen in a DBMS_LOGMNR.START_LOGMNR() invocation, NULL otherwise. This column may also be NULL if the query is run over a time/SCN range that does not contain the start of the transaction.
COMMIT_TIMESTAMP	DATE	Timestamp when the transaction committed; only meaningful if the COMMITTED_DATA_ONLY option was chosen in a DBMS_LOGMNR.START_LOGMNR() invocation
XIDUSN	NUMBER	Transaction ID undo segment number of the transaction that generated the change
XIDSLT	NUMBER	Transaction ID slot number of the transaction that generated the change

Column	Datatype	Description
XIDSQN	NUMBER	Transaction ID sequence number of the transaction that generated the change
XID	RAW(8)	Raw representation of the transaction identifier
PXIDUSN	NUMBER	Parent transaction ID undo segment number of a parallel transaction
PXIDSLT	NUMBER	Parent transaction ID slot number of a parallel transaction
PXIDSQN	NUMBER	Parent transaction ID sequence number of a parallel transaction
PXID	RAW(8)	Raw representation of the parent transaction identifier
TX_NAME	VARCHAR2(256)	Name of the transaction that made the change; only meaningful if the transaction is a named transaction
OPERATION	VARCHAR2(32)	User level SQL operation that made the change: • INTERNAL - Change was caused by internal operations initiated by the database • INSERT - Change was caused by an insert statement • DELETE - Change was caused by a delete statement • UPDATE - Change was caused by an update statement • DDL - Change was caused by a DDL statement • START - Change was caused by the start of a transaction • COMMIT - Change was caused by the commit of a transaction • SEL_LOB_LOCATOR - Operation was a SELECT statement that returned a LOB locator • LOB_WRITE - Change was caused by an invocation of DBMS_LOB.WRITE • LOB_TRIM - Change was caused by an invocation of DBMS_LOB.TRIM • LOB_ERASE - Change was caused by an invocation of DBMS_LOB.ERASE • MISSING_SCN - LogMiner encountered a gap in the redo records. This is most likely because not all redo logs were registered with LogMiner. • ROLLBACK - Change was caused by a full rollback of a transaction • XML DOC BEGIN - Beginning of a change to an XMLType column or table • XML DOC WRITE - Data for an XML document • XML DOC END - End of the Data for an XML document • UNSUPPORTED - Change was caused by operations not currently supported by LogMiner (for example, changes made to nested tables) The OPERATION and OPERATION_CODE columns in this view are available for top-level user operations, for example, DML and DDL. Values that are not documented for these columns are internal to LogMiner or the RDBMS and do not reflect user operations.

Column	Datatype	Description
OPERATION_CODE	NUMBER	Number of the operation code: • 0 - INTERNAL • 1 - INSERT • 2 - DELETE • 3 - UPDATE • 5 - DDL • 6 - START • 7 - COMMIT • 9 - SELECT_LOB_LOCATOR • 10 - LOB_WRITE • 11 - LOB_TRIM • 29 - LOB_ERASE • 34 - MISSING_SCN • 36 - ROLLBACK • 68 - XML DOC BEGIN • 70 = XML DOC WRITE • 71 = XML DOC END • 255 - UNSUPPORTED
ROLLBACK	NUMBER	1 = if the redo record was generated because of a partial or a full rollback of the associated transaction 0 = otherwise
SEG_OWNER	VARCHAR2(386)	Owner of the modified data segment
SEG_NAME	VARCHAR2(256)	Name of the modified data segment
TABLE_NAME	VARCHAR2(386)	Name of the modified table (in case the redo pertains to a table modification)
SEG_TYPE	NUMBER	Type of the modified data segment: • 0 - UNKNOWN • 1 - INDEX • 2 - TABLE • 19 - TABLE PARTITION • 20 - INDEX PARTITION • 34 - TABLE SUBPARTITION • All other values - UNSUPPORTED
SEG_TYPE_NAME	VARCHAR2(32)	Segment type name: • INDEX • INDEX PARTITION • TABLE • TABLE PARTITION • TABLE SUBPARTITION • UNKNOWN • UNSUPPORTED
TABLE_SPACE	VARCHAR2(92)	Name of the tablespace containing the modified data segment. This column is not populated for rows where the value of the OPERATION column is DDL. This is because DDL may operate on more than one tablespace.
ROW_ID	VARCHAR2(18)	Row ID of the row modified by the change (only meaningful if the change pertains to a DML). This will be NULL if the redo record is not associated with a DML.
USERNAME	VARCHAR2(384)	Name of the user who executed the transaction

Column	Datatype	Description
OS_USERNAME	VARCHAR2(4000)	Name of the operating system user
MACHINE_NAME	VARCHAR2(4000)	Machine from which the user connected to the database
AUDIT_SESSIONID	NUMBER	Audit session ID associated with the user session making the change
SESSION#	NUMBER	Session number of the session that made the change
SERIAL#	NUMBER	Serial number of the session that made the change
SESSION_INFO	VARCHAR2(4000)	Information about the database session that executed the transaction. Contains process information, machine name from which the user logged in, and so on. A possible SESSION_INFO column may contain the following: • login_username = HR • client_info = • OS_username = jkundu • Machine_name = nirvan • OS_terminal = pts/31 • OS_program_name = sqlplus@nirvan (TNS V1-V3)
THREAD#	NUMBER	Number of the thread that made the change to the database
SEQUENCE#	NUMBER	Sequence number of the SQL statement within the transaction. If you are mining without the COMMITED_DATA_ONLY option set, then this value is 1.
RBASQN	NUMBER	Sequence# associated with the Redo Block Address (RBA) of the redo record associated with the change
RBABLK	NUMBER	RBA block number within the log file
RBABYTE	NUMBER	RBA byte offset within the block
UBAFIL	NUMBER	Undo Block Address (UBA) file number identifying the file containing the undo block
UBABLK	NUMBER	UBA block number for the undo block
UBAREC	NUMBER	UBA record index within the undo block
UBASQN	NUMBER	UBA undo block sequence number
ABS_FILE#	NUMBER	Data block absolute file number of the block changed by the transaction
REL_FILE#	NUMBER	Data block relative file number. The file number is relative to the tablespace of the object.
DATA_BLK#	NUMBER	Data block number within the file
DATA_OBJ#	NUMBER	Data block object number identifying the object
DATA_OBJV#	NUMBER	Version number of the table being modified
DATA_OBJD#	NUMBER	Data block data object number identifying the object within the tablespace
SQL_REDO	VARCHAR2(4000)	Reconstructed SQL statement that is equivalent to the original SQL statement that made the change. Refer to *Oracle Database Utilities* before executing SQL_REDO to your database. LogMiner does not generate SQL redo for temporary tables. In such a case, this column will contain the string "/* No SQL_REDO for temporary tables */".

Column	Datatype	Description
SQL_UNDO	VARCHAR2(4000)	Reconstructed SQL statement that can be used to undo the effect of the original statement that made the change. DDL statements have no corresponding SQL_UNDO. Refer to *Oracle Database Utilities* before executing SQL_UNDO to your database. LogMiner does not generate SQL undo for temporary tables. In such a case, this column will contain the string "/* No SQL_UNDO for temporary tables */".
RS_ID	VARCHAR2(32)	Record set ID. The tuple (RS_ID, SSN) together uniquely identifies a logical row change. This will usually mean one row from V$LOGMNR_CONTENTS, but could be more than one row if a single SQL statement for either the Redo or Undo would be too large to fit within the respective columns SQL_UNDO or SQL_REDO. RS_ID uniquely identifies the redo record that generated the row.
SSN	NUMBER	SQL sequence number. Used in conjunction with RS_ID, this uniquely identifies a logical row change, shown as one or more rows from the V$LOGMNR_CONTENTS view.
CSF	NUMBER	Continuation SQL flag. Possible values are: • 0 - Indicates SQL_REDO and SQL_UNDO is contained within the same row • 1 - Indicates that either SQL_REDO or SQL_UNDO is greater than 4000 bytes in size and is continued in the next row returned by the view
INFO	VARCHAR2(64)	Informational message about the row. For instance, the string "USER DDL" indicates that the DDL statement returned in the SQL_REDO column was the top-level DDL executed by the user and the string "INTERNAL DDL" indicates that the DDL statement returned in the SQL_REDO column was executed internally by the RDBMS.
STATUS	NUMBER	A value of 0 indicates that the reconstructed SQL statements as shown in the SQL_REDO and SQL_UNDO columns are valid executable SQL statements. Otherwise, the reconstructed SQL statements are not executable. This may be because no data dictionary was provided to LogMiner for the analysis, or that the data dictionary provided did not have the definition of the object being mined. A value of 5 indicates that this row is part of a change to an XMLType column or table and the XML document must be assembled before being applied.
REDO_VALUE	NUMBER	Used as input to the DBMS_LOGMNR.MINE_VALUE() and DBMS_LOGMNR.COLUMN_PRESENT() functions
UNDO_VALUE	NUMBER	Used as input to the DBMS_LOGMNR.MINE_VALUE() and DBMS_LOGMNR.COLUMN_PRESENT() functions
SAFE_RESUME_SCN	NUMBER	Reserved for future use
CSCN	NUMBER	This column is deprecated in favor of the COMMIT_SCN column
OBJECT_ID	RAW(16)	Object identifier for DMLs to XMLType tables. For changes to non-typed tables, this column is NULL.
EDITION_NAME	VARCHAR2(384)	Identifies the edition in which a DDL statement was executed
CLIENT_ID	VARCHAR2(64)	Client identifier in the session that performed the operation, if available.
SRC_CON_NAME	VARCHAR2(384)	Contains the pluggable database (PDB) name. This information will only be available when mining with a current LogMiner dictionary.
SRC_CON_ID	NUMBER	Contains the PDB ID (the PDB_ID column from the DBA_PDBS view). This information will be available only with a current LogMiner dictionary.

Column	Datatype	Description
SRC_CON_UID	NUMBER	Contains the PDB UID (the CON_UID column from the DBA_PDBS view). This information will be available with or without a current LogMiner dictionary.
SRC_CON_DBID	NUMBER	Contains the PDB identifier (the DBID column from the DBA_PDBS view). This information will only be available when mining with a current LogMiner dictionary.
SRC_CON_GUID	RAW(16)	Contains the GUID associated with the PDB (the GUID column from the DBA_PDBS view). This information will only be available when mining with a current LogMiner dictionary.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root

See Also:

- *Oracle Database PL/SQL Packages and Types Reference* for more information about the DBMS_LOGMNR package
- *Oracle Database PL/SQL Packages and Types Reference* for more information about the DBMS_LOB package

9.52 V$LOGMNR_DICTIONARY

V$LOGMNR_DICTIONARY contains log history information.

Column	Datatype	Description
DB_NAME	VARCHAR2(9)	Name of the database
DB_ID	NUMBER	Database ID
DB_CREATED	DATE	Creation date of the source database (corresponds to the CREATED column in the V$DATABASE view)
TIMESTAMP	DATE	Date when the dictionary was created
RESET_SCN	NUMBER	Reset log SCN when the dictionary was created
RESET_SCN_TIME	DATE	Timestamp of the reset log SCN when the dictionary was created
DB_VERSION_TIME	DATE	Version time for the source database (corresponds to the VERSION_TIME column in the V$DATABASE view)
DB_CHARACTER_SET	VARCHAR2(30)	Character set of the source database
DB_VERSION	VARCHAR2(64)	This column is deprecated.
DB_STATUS	VARCHAR2(64)	This column is deprecated.
DICTIONARY_SCN	NUMBER	Database checkpoint SCN at which the dictionary was created
ENABLED_THREAD_MAP	RAW(16)	This column is deprecated.
DB_TXN_SCN	NUMBER	SCN at which the dictionary was created

Column	Datatype	Description
FILENAME	VARCHAR2(512)	Dictionary file name
INFO	VARCHAR2(32)	Informational/Status message BAD_DATE indicates that the SCN of the dictionary file does not match the SCN range of the log files
STATUS	NUMBER	A NULL indicates a valid dictionary file for the list of log files. A non-NULL value indicates further information is contained in the INFO column as a text string.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.53 V$LOGMNR_DICTIONARY_LOAD

V$LOGMNR_DICTIONARY_LOAD displays information about LogMiner dictionaries for all active LogMiner sessions on the system.

Each query of this view will return one row for each attached LogMiner session. This view will not show valid information for LogMiner adhoc query clients.

Column	Datatype	Description
SESSION_ID	NUMBER	LogMiner session ID
LOGMNR_UID	NUMBER	LogMiner dictionary UID
ACTION#	NUMBER	Reserved for internal use
OPCODE	NUMBER	Reserved for internal use
COMMAND	VARCHAR2(161)	Current command being executed
CURRENT_STATE	VARCHAR2(32)	Summary information if LOADED=ACTIVE
COMPLETED_ACTIONS	NUMBER	The steps completed so far
TOTAL_ACTIONS	NUMBER	Total steps to complete
LOADED	VARCHAR2(7)	Status of the dictionary load: • YES - dictionary has been loaded • NO - dictionary has not been loaded • ACTIVE - dictionary is in the process of being loaded
PERCENT_DONE	NUMBER	Percentage of the dictionary that has been loaded
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.54 V$LOGMNR_LATCH

V$LOGMNR_LATCH can be joined with the V$LATCH and the V$LATCH_CHILDREN views to obtain statistics about different latches used by active LogMiner persistent sessions.

A persistent LogMiner session is created either by starting Data Guard SQL Apply on a logical standby database for the first time or by creating a Replication capture.

Column	Datatype	Description
SESSION_ID	NUMBER	Unique identifier of the LogMiner persistent session
NAME	VARCHAR2(32)	Name of the Latch: • LogMiner internal state - Identifies the latch that protects communications between the LogMiner READER, PREPARER, and BUILDER processes • LogMiner memory allocation - Identifies the latch that protects all memory allocation and deallocation inside LogMiner • LogMiner transaction list - Identifies the latch that protects interaction between the LogMiner layer and its clients during Data Guard SQL Apply on a logical standby database
CHILD_ADDR	RAW(4 | 8)	Address of the child latch object. This column matches the corresponding ADDR column in the V$LATCH_CHLDREN view
STATE	VARCHAR2(6)	State of the Latch: • UNINIT - The latch structure is uninitialized; that is, it is not currently assigned to any LogMiner persistent session • READY - The latch structure is being used by a persistent LogMiner session
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

See Also:

- "V$LATCH"
- "V$LATCH_CHILDREN"

9.55 V$LOGMNR_LOGS

V$LOGMNR_LOGS contains log information.

Column	Datatype	Description
LOG_ID	NUMBER	This column is deprecated.
FILENAME	VARCHAR2(512)	Name of the log file

Column	Datatype	Description
LOW_TIME	DATE	Oldest date of any records in the file
HIGH_TIME	DATE	Most recent date of any records in the file
DB_ID	NUMBER	Database ID
DB_NAME	VARCHAR2(8)	Name of the database
RESET_SCN	NUMBER	Resetlogs SCN of the database incarnation that generated the log file
RESET_SCN_TIME	DATE	Resetlogs timestamp of the database incarnation that generated the log file
COMPATIBLE	VARCHAR2(17)	The setting of the database COMPATIBLE initialization parameter at the time the log file was generated
THREAD_ID	NUMBER	Thread number
THREAD_SQN	NUMBER	Thread sequence number
LOW_SCN	NUMBER	SCN allocated when log switched into
NEXT_SCN	NUMBER	SCN after this log. Low SCN of the next log.
DICTIONARY_BEGIN	VARCHAR2(3)	Indicates whether dictionary dumped to redo logs starts in this redo log (YES) or not (NO)
DICTIONARY_END	VARCHAR2(3)	Indicates whether dictionary dumped to redo logs ends in this redo log (YES) or not (NO)
TYPE	VARCHAR2(7)	Redo log file type: • ARCHIVED • ONLINE
BLOCKSIZE	NUMBER	Database block size
FILESIZE	NUMBER	Size of the redo file (in bytes)
INFO	VARCHAR2(32)	Informational message. A value of MISSING_LOGFILE will be assigned to a row entry where a needed log file is missing from the list of log files.
STATUS	NUMBER	Status of the redo log file: • 0 - Will be read • 1 - First to be read • 2 - Not needed • 4 - Missing log file
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.56 V$LOGMNR_PARAMETERS

V$LOGMNR_PARAMETERS contains log information.

Column	Datatype	Description
START_DATE	DATE	Date to start the search
REQUIRED_START_DATE	DATE	Required date to start the search if DDL tracking is enabled

Column	Datatype	Description
END_DATE	DATE	Date to end the search
START_SCN	NUMBER	System change number to start the search
REQUIRED_START_SCN	NUMBER	Required system change number to start the search if DDL tracking is enabled
END_SCN	NUMBER	System change number to end the search
OPTIONS	NUMBER	Options specified for the current LogMiner session
INFO	VARCHAR2(32)	This column is always null.
STATUS	NUMBER	This column is always 0.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.57 V$LOGMNR_PROCESS

V$LOGMNR_PROCESS identifies all processes attached to an active LogMiner persistent session.

(A persistent LogMiner session is created either by starting Data Guard SQL Apply on a logical standby database for the first time or by creating Replication capture.) This view can be joined with either the V$SESSION view or the V$PROCESS view to gather process-specific information.

Column	Datatype	Description
SESSION_ID	NUMBER	Unique identifier for the LogMiner persistent session
PID	NUMBER	Oracle process identifier for the SQL Apply or Replication capture process (same as the V$PROCESS.PID)
SPID	VARCHAR2(24)	Operating system process identifier (same as the V$PROCESS.SPID)
ROLE	VARCHAR2(32)	Identifies the role of the active LogMiner process: READER, PREPARER, BUILDER, COORDINATOR, or APPLY SERVER
USERNAME	VARCHAR2(15)	Operating system process user name that is connected to the database
SID	NUMBER	Session identifier for the V$SESSION.SID process
SERIAL#	NUMBER	Session serial number associated with the V$SESSION.SERIAL process
LATCHWAIT	VARCHAR2(16)	Address of the latch the process is waiting for; NULL if none
LATCHSPIN	VARCHAR2(16)	This column is obsolete
WORK_MICROSEC	VARCHAR2(21)	Microseconds spent by the process doing useful work
OVERHEAD_MICROSEC	VARCHAR2(21)	Microseconds spent by the process doing overhead tasks or simply waiting/idling

Column	Datatype	Description
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

See Also:

- "V$SESSION"
- "V$PROCESS"

9.58 V$LOGMNR_SESSION

V$LOGMNR_SESSION displays information about active LogMiner persistent sessions.

(A LogMiner persistent session is created either by starting Data Guard SQL Apply on a logical standby database for the first time or by creating Replication capture.) Transient LogMiner sessions (those created as a result of querying the V$LOGMNR_CONTENTS view) do not show up in the V$LOGMNR_SESSION view. The statistics shown in this view correspond to snapshots of the system and are not cumulative in nature.

Column	Datatype	Description
SESSION_ID	NUMBER	Unique identifier for the LogMiner persistent session
SESSION_NAME	VARCHAR2(32)	Unique session name
SESSION_STATE	VARCHAR2(9)	Current state of the LogMiner persistent session: • READY - Client has created the LogMiner persistent session and added the relevant archived redo log files, but has not loaded the initial LogMiner dictionary • STARTED - The LogMiner dictionary has been loaded • ACTIVE - The LogMiner persistent session is mining the redo stream • DETACHED - The LogMiner persistent session is not currently active, and it is in the process of becoming inactive • DISCARDED - Client is about to destroy the LogMiner persistent session
DB_NAME	VARCHAR2(128)	Global database name for the source database
DB_ID	NUMBER	Database identifier of the source database
RESET_SCN	NUMBER	System change number (SCN) when the session started
RESET_TIMESTAMP	NUMBER	Time of the RESETLOGs when the LogMiner persistent session started
RESET_TIME	DATE	The time when the session started
NUM_PROCESS	NUMBER	Number of processes allocated to this session
CHUNK_SIZE	NUMBER	Amount of memory allocated for this chunk

Column	Datatype	Description
START_SCN	NUMBER	System change number (SCN) at start of the session
END_SCN	NUMBER	System change number (SCN) at end of the session
SPILL_SCN	NUMBER	In the event of a restart, redo records with an SCN lower than this will not be read from the archived redo log files
PROCESSED_SCN	NUMBER	The BUILDER process has successfully mined redo records up to this SCN
PROCESSED_TIME	DATE	The BUILDER process has successfully mined redo records up to this time
PREPARED_SCN	NUMBER	The PREPARER processes have successfully transformed all redo records below this SCN into logical change records (LCRs). However the LCRs may not have been grouped into transactions or merged in case they pertain to chained rows or LOB updates.
READ_SCN	NUMBER	The READER process has read all redo records below this SCN
LOW_MARK_SCN	NUMBER	LogMiner has delivered all transactions that committed below this SCN to the client
CONSUMED_SCN	NUMBER	Client has consumed and released all transactions that committed below this SCN.
MAX_MEMORY_SIZE	NUMBER	Maximum amount of shared memory (in bytes) that LogMiner is allowed to consume
USED_MEMORY_SIZE	NUMBER	Amount of shared memory (in bytes) actually consumed by LogMiner
BUILDER_WORK_SIZE	NUMBER	Amount of redo (in bytes) in the current work unit being processed by the BUILDER process.
PREPARED_WORK_SIZE	NUMBER	Amount of redo (in bytes) that has been prepared by LogMiner
AVAILABLE_WORK_SIZE	NUMBER	Amount of redo records (in bytes) that are ready, but are yet to be consumed by the client
AVAILABLE_TXN	NUMBER	Number of transaction chunks ready for consumption
AVAILABLE_COMMITTED_TXN	NUMBER	Number of committed transactions ready for consumption. This is less than, or equal to, AVAILABLE_TXN.
DELIVERED_TXN	NUMBER	Number of transaction chunks that the client currently has in its possession
DELIVERED_COMMITTED_TXN	NUMBER	Number of committed transaction chunks that the client is currently working on. This is less than, or equal to, the value of the DELIVERED_TXN column.
PINNED_TXN	NUMBER	Number of transactions pinned (the client is actively working on)
PINNED_COMMITTED_TXN	NUMBER	Number of committed transactions pinned (the client is actively working on)
CHECKPOINT_INTERVAL	NUMBER	Checkpoint interval
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

> **See Also:**
>
> "V$LOGMNR_CONTENTS"

9.59 V$LOGMNR_STATS

V$LOGMNR_STATS displays the activity currently being performed by the active LogMiner persistent sessions.

Column	Datatype	Description
SESSION_ID	NUMBER	Unique identifier for the LogMiner persistent sessions

Column	Datatype	Description
NAME	VARCHAR2(64)	Name of the LogMiner statistic, state, or status, including: • Bytes of Redo Processed - Cumulative count of bytes processed by SQL Apply • Redo Records Processed - Count of redo records processed by SQL Apply • Txns Delivered to Client - Count of SQL transactions processed by SQL Apply • DML txns delivered - Count of DML transactions processed by SQL Apply • DDL txns delivered - Count of DDL transactions processed by SQL Apply • CTAS txns delivered - Count of `CREATE TABLE AS SELECT` (CTAS) transactions processed by SQL Apply • Recursive txns delivered - Count of recursive transactions processed by SQL Apply • Rolled back txns seen • LCRs delivered to client - Number of logical change records (LCRs) processed by SQL Apply • Bytes paged out - Cumulative count of bytes that have been paged out. LogMiner pages out memory from the LCR cache to accommodate certain ill-behaved workloads or under-configured systems. The ratio of bytes paged out to bytes of redo processed should be low. If this ratio is high (10% or higher), try increasing the `MAX_SGA` allocated to SQL Apply. • Microsecs spent in pageout - Time spent by LogMiner paging out memory from the LCR cache • Bytes checkpointed - Keeps track of the amount of bytes checkpointed. The mining engine takes periodic checkpoints, whereby it writes out logical change records (LCRs) pertaining to long-running transactions. The ratio of `Bytes Checkpointed` to `Bytes of Redo Processed` should be low. A high ratio (10% or higher) indicates an ill-behaved workload. • Microsecs spent in checkpoint - Time spent by the mining engine taking checkpoints, whereby it writes out logical change records (LCRs) pertaining to long-running transactions. • Bytes rolled back - Cumulative value of the number of bytes rolled back by LogMiner. There are times that LogMiner needs to backtrack and reprocess a section of the redo stream. In this case, it will roll back work it has already done. The ratio of `Bytes Rolled Back` to `Bytes of Redo Processed` should be low. If this ratio is high (10% or higher), reduce the number of `PREPARER` processes allocated to SQL Apply. • Microsecs spent in rollback - Time spent rolling back transactions already applied to the logical standby database
VALUE	VARCHAR2(64)	The corresponding metric value
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • `0`: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • `1`: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.60 V$LOGSTDBY

V$LOGSTDBY is deprecated.

Column	Datatype	Description
SERIAL#	NUMBER	SQL Session serial number. This data is used when joining this view with the V$SESSION and V$PX_SESSION views.
LOGSTDBY_ID	NUMBER	Parallel query worker ID
PID	VARCHAR2(24)	Process ID of the SQL apply process
TYPE	VARCHAR2(128)	Indicates the task being performed by the process (COORDINATOR, APPLIER, ANALYZER, READER, PREPARER, or BUILDER)
STATUS_CODE	NUMBER	Status number (or Oracle error code) belonging to the STATUS message
STATUS	VARCHAR2(256)	Description of the current activity of the process
HIGH_SCN	NUMBER	Highest system change number (SCN) seen by the process. This column is used to confirm the progress of the individual process.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.61 V$LOGSTDBY_PROCESS

V$LOGSTDBY_PROCESS displays dynamic information about what is happening to the Data Guard log apply services.

This view is helpful when diagnosing performance problems during the logical application of archived redo logs to the standby database, and it can be helpful for other problems. This view is for logical standby databases only.

Column	Datatype	Description
SID	NUMBER	Session id of the associated session. This matches the SID column of the corresponding row in the V$SESSION view.
SERIAL#	NUMBER	Serial number of the associated session. Together, (SID,SERIAL#) uniquely identify the session in the current database instance.
LOGSTDBY_ID	NUMBER	Parallel query worker ID
SPID	VARCHAR2(24)	This corresponds to the SPID value of the row corresponding to this process in the V$PROCESS view

Column	Datatype	Description
TYPE	VARCHAR2(128)	Role that the process plays in the context of SQL Apply: • COORDINATOR • APPLIER • ANALYZER • READER • PREPARER • BUILDER
STATUS_CODE	NUMBER	Operation code identifying the current action of the process: • 16111 - SQL Apply process is initializing • 16112 - SQL Apply process is cleaning up as apply and mining processes are stopping based on a user command • 16116 - SQL Apply process is idle • 16117 - SQL Apply process is busy and is not waiting on any interesting event • 16110 - APPLIER process has invoked a user-provided stored procedure in order to inspect a DDL statement prior to it being processed • 16113 - APPLIER process is applying DML changes to some user object or to a sequence • 16114 - APPLIER process is applying a DDL change • 16115 - COORDINATOR process is loading the LogMiner dictionary from the redo stream • 16243 - BUILDER process is paging out memory to free up space in lcr cache • 16240 - READER process idle waiting for additional logfile to be available • 16241 - READER process is idle waiting for the logfile to fill the log sequence gap • 16242 - READER process is processing a logfile
STATUS	VARCHAR2(256)	Description of the current action of the process
HIGH_SCN	NUMBER	Identifies the highest redo record/LCR processed by this process
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.62 V$LOGSTDBY_PROGRESS

V$LOGSTDBY_PROGRESS displays the progress of log apply services on the logical standby database. This view is for logical standby databases only.

Column	Datatype	Description
APPLIED_SCN	NUMBER	All the transactions with COMMIT SCN lower than or equal to this SCN have been applied
APPLIED_TIME	DATE	The time and date of APPLIED_SCN

Column	Datatype	Description
RESTART_SCN	NUMBER	During an apply restart, LogMiner does not read any log file with a NEXT_CHANGE# lower than this SCN.
RESTART_TIME	DATE	The time and date of RESTART_SCN
LATEST_SCN	NUMBER	The highest SCN of all redo records that Logical Standby has encountered
LATEST_TIME	DATE	The time and date of LATEST_SCN
MINING_SCN	NUMBER	The SCN of the latest redo record processed by the builder process
MINING_TIME	DATE	The time and date of MINING_SCN
RESETLOGS_ID	NUMBER	A redo branch is identified by resetlogs SCN and resetlogs timestamp. The RESETLOGS_ID column contents are the same as resetlogs timestamp converted to a number.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.63 V$LOGSTDBY_STATE

V$LOGSTDBY_STATE provides consolidated information from V$LOGSTDBY and V$LOGSTDBY_STATS about the running state of Logical Standby.

Column	Datatype	Description
PRIMARY_DBID	NUMBER	Database ID (DBID) of the primary database
PRIMARY_CON_DBID	NUMBER	This column indicates the DBID of the source database or source PDB corresponding to the database (or CDB) from which this column is queried.For a non-CDB and the root of a CDB, this column matches the PRIMARY_DBID column.For a maintained PDB, this column indicates the DBID for the corresponding PDB at the source.For a skipped PDB or a local PDB, this column would be NULL.
SESSION_ID	NUMBER	LogMiner session ID allocated to SQL Apply.
REALTIME_APPLY	VARCHAR2(64)	Y indicates that SQL Apply is running in real-time apply mode. If a standby redo log is configured, SQL Apply applies changes as they are written to the standby redo log files. N indicates that SQL Apply applies changes as each archived redo log file is received.

Column	Datatype	Description
STATE	VARCHAR2(64)	• INITIALIZING: LogMiner session has been created and coordinator has attached to it • LOADING DICTIONARY: SQL Apply is loading the LogMiner dictionary • WAITING ON GAP: SQL Apply is waiting for a log file to be sent from the primary database • APPLYING: SQL Apply is actively mining or applying transactions • WAITING FOR DICTIONARY LOGS: SQL Apply is waiting for the archived logs containing the LogMiner dictionary to be shipped from the primary database • IDLE: SQL Apply has applied all changes available at the logical standby, and is caught up with the primary database
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

See Also:

- "V$LOGSTDBY"
- "V$LOGSTDBY_STATS"

9.64 V$LOGSTDBY_STATS

V$LOGSTDBY_STATS displays statistics, current state, and status information related to SQL Apply.

No rows are returned from this view when SQL Apply is not running. This view is only meaningful in the context of a logical standby database.

All statistics shown in this view are reinitialized at each SQL Apply start.

Column	Datatype	Description
NAME	VARCHAR2(64)	Name of the statistic, state, or status: **Note:** Many of the following statistics are subject to change or deletion; programmers should write application code to tolerate missing or extra statistics. • id of the logminer session used by SQL Apply to mine the redo logs. • number of preparers • number of appliers • server processes in use for SQL Apply • maximum SGA (in MBytes) for LCR cache • whether SQL Apply is preserving the commit order seen at the primary database while applying changes • maximum events recorded in the DBA_LOGSTDBY_EVENTS table • whether SQL Apply is logging errors that are skipped in the DBA_LOGSTDBY_EVENTS table • whether SQL Apply is logging DDLs that are skipped in the DBA_LOGSTDBY_EVENTS table • whether SQL Apply is logging DDLs that are applied in the DBA_LOGSTDBY_EVENTS table • whether SQL Apply is logging unsupported operations that are encountered in the DBA_LOGSTDBY_EVENTS table • whether or not real time apply is on • value of apply delay (in minutes) • coordinator state • coordinator uptime in seconds • time of the most recent start of SQL Apply • number of transactions mined and made available for apply • number of transactions applied • number of rolled back transactions mined • number of DDL txns mined • number of CTAS (Create Table as Select) txns mined • number of thread enable events encountered in the redo stream • number of thread disable events encountered in the redo stream • bytes of redo records mined • bytes paged out • seconds spent in pageout activity • bytes checkpointed • seconds spent in checkpointing activity • seconds SQL Apply is idle • number of times a complete standby redo logs are mined without having to mine the corresponding archived log • number of times SQL Apply had to switch from a standby redo log to the corresponding archived log • number of times SQL Apply mined redo from the archived logs • number of archived logs that arrived at the standby via gap fetch mechanism (gap fetched logs mined) • number of failed attempts to open a logfile • amount of time spent in waiting for the current gap to resolve if SQL Apply is running in real time mode (current logfile wait)[1] • time spent in waiting for gap to resolve if SQL Apply is running in real time mode (total logfile wait)[2]
VALUE	VARCHAR2(64)	Value of the statistic or state information

Column	Datatype	Description
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

1 In case SQL Apply is not running in real time mode, this may not reflect time spent in gap resolution, but simply the time spent waiting for the most recent archived log to show up at the logical standby.
2 In case SQL Apply is not running in real time mode, this will include time that SQL Apply spent every time it finished processing all archived logs registered with it, and waited for the next log to be archived.

9.65 V$LOGSTDBY_TRANSACTION

V$LOGSTDBY_TRANSACTION displays all transactions that are actively being processed by SQL Apply.

The transaction identifiers shown in this view are those mined from the redo stream and correspond to transaction identifiers assigned at the primary database, and do not correspond to the transactions that are active at the logical standby database. For information regarding transactions active in the logical standby database, including those created as part of SQL Apply, query the V$TRANSACTION view at the logical standby database.

Column	Datatype	Description
PRIMARY_XIDUSN	NUMBER	Undo segment number of the transaction
PRIMARY_XIDSLT	NUMBER	Slot number of the transaction
PRIMARY_XIDSQN	NUMBER	Sequence number of the transaction
PRIMARY_XID	RAW(8)	Transaction ID
PRIMARY_START_SCN	NUMBER	Start system change number (SCN) base
PRIMARY_START_TIME	DATE	Start time
PRIMARY_PARENT_XIDUSN	NUMBER	Undo segment number of the parent transaction
PRIMARY_PARENT_XIDSLT	NUMBER	Slot number of the parent transaction
PRIMARY_PARENT_XIDSQN	NUMBER	Sequence number of the parent transaction
PRIMARY_PARENT_XID	RAW(8)	Transaction ID of the parent transaction (PDML)
TYPE	VARCHAR2(32)	Type: • PL/SQL - Transaction was done as part of a supported PL/SQL procedure • Direct Path Load - Transaction is a direct path load • CTAS - Transaction contains at least one CREATE TABLE ... AS SELECT operation • DDL - Transaction contains one or more DDL operations • PDML Child - Transaction is a child transaction • DML - Transaction contains only DML operations

Column	Datatype	Description
MINING_STATUS	VARCHAR2(32)	Mining status: • ACTIVE - Transaction is still being mined by LogMiner. At least part of this transaction is ready to be applied or has already been applied. • COMPLETE - Transaction is complete and ready to be applied. LogMiner has finished mining.
SRC_CON_ID	NUMBER	Contains the PDB ID (the PDB_ID column from the DBA_PDBS view) of the source database that generated the change for this transaction.
APPLY_STATUS	VARCHAR2(6)	• ACTIVE - Transaction has been assigned to an apply server. It is in one of the following states: - The transaction is being actively applied - The transaction is being held by an apply server waiting for certain events to occur - The transaction is being held by an apply server waiting for subsequent parts of this transaction • NONE - Transaction has not yet been assigned to an apply server
SID	NUMBER	Session ID of the apply server's session; Null if APPLY_STATUS is NONE
SERIAL#	NUMBER	Serial number of the apply server's session; Null if APPLY_STATUS is NONE
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

See Also:

"V$TRANSACTION"

9.66 V$MANAGED_STANDBY

V$MANAGED_STANDBY displays current status information for some Oracle Database processes related to physical standby databases in the Data Guard environment. This view does not persist after an instance shutdown.

Column	Datatype	Description
PROCESS	VARCHAR2(9)	Type of the process whose information is being reported: • RFS - Remote file server • MRP0 - Detached recovery server process • MR(fg) - Foreground recovery session • ARCH - Archiver process • DGRD - Generic Oracle Data Guard process • FGRD • LGWR • RFS(FAL) • RFS(NEXP) • LNS - ASYNC Redo Transport process
PID	VARCHAR2(24)	Operating system process identifier of the process
STATUS	VARCHAR2(20)	Current process status: • ALLOCATED - Process is active but not currently connected to a primary database • ANNOUNCING - Process is announcing the existence of a potential dependent archived redo log • APPLYING_LOG - Process is actively applying the archived redo log to the standby database • ATTACHED - Process is actively attached and communicating to a primary database • CLOSING - Process has completed archival and is closing the archived redo log • CONNECTED - Network connection established to a primary database • ERROR - Process has failed • IDLE - Process is not performing any activities • OPENING - Process is opening the archived redo log • RECEIVING - Process is receiving network communication • REGISTERING - Process is registering the existence of a completed dependent archived redo log • UNUSED - No active process • WAIT_FOR_GAP - Process is waiting for the archive gap to be resolved • WAIT_FOR_LOG - Process is waiting for the archived redo log to be completed • WAIT_FOR_NEW_PRIMARY - Process is waiting for a new primary (source PDB) to register itself with this container database • WRITING - Process is actively writing redo data to the archived redo log
CLIENT_PROCESS	VARCHAR2(8)	Identifies the corresponding primary database process: • Archival - Foreground (manual) archival process (SQL) • ARCH - Background ARC*n* process • LGWR - Background LGWR process
CLIENT_PID	VARCHAR2(40)	Operating system process identifier of the client process
CLIENT_DBID	VARCHAR2(40)	Database identifier of the primary database
GROUP#	VARCHAR2(40)	Standby redo log group
RESETLOG_ID	NUMBER	Resetlogs identifier of the archived redo log
THREAD#	NUMBER	Archived redo log thread number
SEQUENCE#	NUMBER	Archived redo log sequence number

Column	Datatype	Description
BLOCK#	NUMBER	Last processed archived redo log block number
BLOCKS	NUMBER	Count (in 512-byte blocks) of the last write to a redo log, or for a recovery process, the expected final read count
DELAY_MINS	NUMBER	Archived redo log delay interval in minutes
KNOWN_AGENTS	NUMBER	Total number of standby database agents processing an archived redo log
ACTIVE_AGENTS	NUMBER	Number of standby database agents actively processing an archived redo log
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

Note:

This view is deprecated in Oracle Database 12*c* Release 2 (12.2.0.1) and may be desupported in a future release. The V$DATAGUARD_PROCESS view should be used, instead.

See Also:

"V$DATAGUARD_PROCESS"

9.67 V$MAP_COMP_LIST

V$MAP_COMP_LIST displays supplementary information for all element mapping structures.

Column	Datatype	Description
ELEM_IDX	NUMBER	Index corresponding to the element
NUM_COMP	NUMBER	Number of components (maximum is 5)
COMP1_NAME	VARCHAR2(256)	Name of the first component
COMP1_VAL	VARCHAR2(256)	Value of the first component
COMP2_NAME	VARCHAR2(256)	Name of the second component
COMP2_VAL	VARCHAR2(256)	Value of the second component
COMP3_NAME	VARCHAR2(256)	Name of the third component
COMP3_VAL	VARCHAR2(256)	Value of the third component
COMP4_NAME	VARCHAR2(256)	Name of the fourth component
COMP4_VAL	VARCHAR2(256)	Value of the fourth component

Column	Datatype	Description
COMP5_NAME	VARCHAR2(256)	Name of the fifth component
COMP5_VAL	VARCHAR2(256)	Value of the fifth component
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.68 V$MAP_ELEMENT

V$MAP_ELEMENT displays a list of all element mapping structures in the SGA of the instance.

Column	Datatype	Description
ELEM_NAME	VARCHAR2(256)	Element name
ELEM_IDX	NUMBER	Index corresponding to the element
ELEM_CFGID	VARCHAR2(256)	Configuration ID (N/A if configuration ID is not supported)
ELEM_TYPE	VARCHAR2(12)	Element type: • MIRROR • STRIPE • RAID5 • CONCATENATED • PARTITION • DISK • NONE
ELEM_SIZE	NUMBER	Element Size in HKB
ELEM_NSUBELEM	NUMBER	Number of Subelements
ELEM_DESCR	VARCHAR2(256)	Element Description
STRIPE_SIZE	NUMBER	Stripe Size in HKB for RAID-5 and STRIPE elements, 0 for the remaining types
LIB_IDX	NUMBER	Index of the library which claims ownership of the element
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.69 V$MAP_EXT_ELEMENT

V$MAP_EXT_ELEMENT displays supplementary information for all element mapping structures.

Column	Datatype	Description
ELEM_IDX	NUMBER	Index corresponding to the element
NUM_ATTRB	NUMBER	Number of Attributes (maximum is 5)
ATTRB1_NAME	VARCHAR2(256)	Name of the first Attribute
ATTRB1_VAL	VARCHAR2(256)	Value of the first attribute
ATTRB2_NAME	VARCHAR2(256)	Name of the second attribute
ATTRB2_VAL	VARCHAR2(256)	Value of the second attribute
ATTRB3_NAME	VARCHAR2(256)	Name of the third attribute
ATTRB3_VAL	VARCHAR2(256)	Value of the third attribute
ATTRB4_NAME	VARCHAR2(256)	Name of the fourth attribute
ATTRB4_VAL	VARCHAR2(256)	Value of the fourth attribute
ATTRB5_NAME	VARCHAR2(256)	Name of the fifth attribute
ATTRB5_VAL	VARCHAR2(256)	Value of the fifth attribute
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.70 V$MAP_FILE

V$MAP_FILE displays a list of all file mapping structures in the shared memory of the instance.

Column	Datatype	Description
FILE_MAP_IDX	NUMBER	Index corresponding to the file
FILE_CFGID	VARCHAR2(256)	Configuration ID (N/A if configuration ID is not supported)
FILE_STATUS	VARCHAR2(7)	Status of the mapping information: • VALID - File mapping information is latest • INVALID - Mapping must be refreshed
FILE_NAME	VARCHAR2(256)	Absolute file name
FILE_TYPE	VARCHAR2(11)	File type: • DATAFILE • SPFILE • TEMPFILE • CONTROLFILE • LOGFILE • ARCHIVEFILE
FILE_STRUCTURE	VARCHAR2(9)	File structure: • FILE • RAWVOLUME • RAWDEVICE • NONE
FILE_SIZE	NUMBER	File size in HKB (half KB)

Column	Datatype	Description
FILE_NEXTS	NUMBER	Number of file extents in the file (not necessarily the same as the number of file extents mapped)
LIB_IDX	NUMBER	Index of mapping library claiming ownership of the file
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.71 V$MAP_FILE_EXTENT

V$MAP_FILE_EXTENT displays a list of all file extent mapping structures in the shared memory of the instance.

Column	Datatype	Description
FILE_MAP_IDX	NUMBER	File index (corresponds to FILE_MAP_IDX in V$MAP_FILE)
EXT_NUM	NUMBER	File extent number
EXT_ELEM_OFF	NUMBER	Element offset in HKB
EXT_SIZE	NUMBER	File extent size in HKB
EXT_FILE_OFF	NUMBER	File Offset in HKB
EXT_TYPE	VARCHAR2(6)	File Extent Type: • DATA • PARITY • NONE
ELEM_IDX	NUMBER	Index in V$MAP_ELEMENT corresponding to the element where the file extent resides
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.72 V$MAP_FILE_IO_STACK

V$MAP_FILE_IO_STACK displays the hierarchical arrangement of storage containers for files. Each row in the view represents a level in the hierarchy.

Column	Datatype	Description
FILE_MAP_IDX	NUMBER	File index (corresponds to FILE_MAP_IDX in V$MAP_FILE)
DEPTH	NUMBER	Element depth within the I/O stack

Column	Datatype	Description
ELEM_IDX	NUMBER	Index corresponding to the element
CU_SIZE	NUMBER	Contiguous set of logical blocks of the file (in HKB units) that is resident contiguously on the element
STRIDE	NUMBER	Number of HKB between contiguous units (CU) in the file that are contiguous on this element. Used in RAID5 and striped files.
NUM_CU	NUMBER	Number of contiguous units that are adjacent to each other on this element that are separated by STRIDE HKB in the file. In RAID5, the number of contiguous units also include the parity stripes.
ELEM_OFFSET	NUMBER	Element offset in HKB units
FILE_OFFSET	NUMBER	Offset in HKB units from the start of the file to the first byte of the contiguous units
DATA_TYPE	VARCHAR2(15)	Datatype: • DATA • PARITY • DATA AND PARITY
PARITY_POS	NUMBER	Position of the parity. Only for RAID5. This column is needed to distinguish the parity from the data part.
PARITY_PERIOD	NUMBER	Parity period. Only for RAID5.
ID	NUMBER	Unique identifier
PARENT_ID	NUMBER	Parent identifier
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.73 V$MAP_LIBRARY

V$MAP_LIBRARY displays a list of all mapping libraries dynamically loaded by the external process.

Column	Datatype	Description
LIB_IDX	NUMBER	Index corresponding to the library
LIB_NAME	VARCHAR2(256)	Absolute library name
VENDOR_NAME	VARCHAR2(64)	Name of the vendor implementing the library
PROTOCOL_NUM	NUMBER	Mapping protocol that the library supports
VERSION_NUM	VARCHAR2(32)	Version number
PATH_NAME	VARCHAR2(4000)	Path name
MAP_FILE	VARCHAR2(1)	Indicates whether the library supports mapping files (Y) or not (N)
FILE_CFGID	VARCHAR2(13)	Type of configuration ID supported for files: • NONE - Not supported • PERSISTENT • NONPERSISTENT

Column	Datatype	Description
MAP_ELEM	VARCHAR2(1)	Indicates whether the library supports mapping elements (Y) or not (N)
ELEM_CFGID	VARCHAR2(13)	Type of configuration id supported for elements: • NONE - Not supported • PERSISTENT • NONPERSISTENT
MAP_SYNC	VARCHAR2(1)	Indicates whether the library needs to be explicitly synchronized so that future mappings reflect the most recent changes (Y) or not (N). Note that configuration IDs cannot be supported if the library needs to be explicitly synced.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.74 V$MAP_SUBELEMENT

V$MAP_SUBELEMENT displays a list of all subelement mapping structures in the shared memory of the instance.

Column	Datatype	Description
CHILD_IDX	NUMBER	Index in V$MAP_ELEMENT corresponding to the child element
PARENT_IDX	NUMBER	Index in V$MAP_ELEMENT corresponding to the parent element
SUB_NUM	NUMBER	Subelement number
SUB_SIZE	NUMBER	Subelement size in HKB
ELEM_OFFSET	NUMBER	Offset in HKB on the child element
SUB_FLAGS	NUMBER	Subelement flags (currently unused)
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.75 V$MAPPED_SQL

V$MAPPED_SQL lists the SQL statements that are translated and mapped in memory to a different SQL statement for execution.

Column	Datatype	Description
SQL_TEXT	VARCHAR2(1000)	First 1000 characters of the original SQL text

Column	Datatype	Description
SQL_FULLTEXT	CLOB	Full text for the original SQL statement
SQL_ID	VARCHAR2(13)	SQL identifier of the original SQL statement
HASH_VALUE	NUMBER	Hash value of the original SQL statement
MAPPED_SQL_TEXT	VARCHAR2(1000)	First 1000 characters of the mapped SQL text
MAPPED_SQL_FULLTEXT	CLOB	Full text for the mapped SQL statement
MAPPED_SQL_ID	VARCHAR2(13)	SQL identifier of the mapped SQL statement
MAPPED_HASH_VALUE	NUMBER	Hash value of the mapped SQL statement
SQL_TRANSLATION_PROFILE_ID	NUMBER	A non-NULL value specifies the object ID of the SQL translation profile used to translate the SQL statement into the mapped SQL statement
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data
TRANSLATION_TIMESTAMP	DATE	Time this SQL statement was translated
TRANSLATION_CPU_TIME	NUMBER	CPU time used to translate this SQL statement
TRANSLATION_ELAPSED_TIME	NUMBER	Elapsed time used to translate this SQL statement
TRANSLATION_METHOD	VARCHAR2(10)	Method used to translate this SQL statement
DICTIONARY_SQL_ID	VARCHAR2(13)	SQL ID of SQL text in custom translation dictionary used to translate this SQL statement
USE_COUNT	NUMBER	Number of times this translation has been used

9.76 V$MEMOPTIMIZE_WRITE_AREA

V$MEMOPTIMIZE_WRITE_AREA displays information about fast ingest data in the large pool.

Column	Datatype	Description
TOTAL_SIZE	NUMBER	Total amount of memory allocated for fast ingest data in the large pool (in bytes)
USED_SPACE	NUMBER	Total amount of memory currently used by fast ingest data in the large pool (in bytes)
FREE_SPACE	NUMBER	Total amount of memory currently free for storing fast ingest data in the large pool (in bytes)
NUM_WRITES	NUMBER	Number of fast ingest insert operations for which data is still in the large pool and is yet to be written to disk
NUM_WRITERS	NUMBER	Number of clients currently using fast ingest for inserting data into the database

Column	Datatype	Description
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.77 V$MEMORY_CURRENT_RESIZE_OPS

V$MEMORY_CURRENT_RESIZE_OPS displays information about memory resize operations (both automatic and manual) which are currently in progress.

An operation can be a grow or a shrink of a dynamic memory component. All sizes are expressed in bytes.

Column	Datatype	Description
COMPONENT	VARCHAR2(64)	Component name
OPER_TYPE	VARCHAR2(13)	Operation type: • STATIC • INITIALIZING • DISABLED • GROW • SHRINK • SHRINK_CANCEL
OPER_MODE	VARCHAR2(9)	Operation mode: • MANUAL • DEFERRED • IMMEDIATE
PARAMETER	VARCHAR2(80)	Name of the parameter for the resize operation
INITIAL_SIZE	NUMBER	Parameter value at the start of the operation
TARGET_SIZE	NUMBER	Desired value of the parameter after the resize
CURRENT_SIZE	NUMBER	Current value of the parameter
START_TIME	DATE	Start time of the operation
LAST_UPDATE_TIME	DATE	Last time progress was made for the operation
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.78 V$MEMORY_DYNAMIC_COMPONENTS

V$MEMORY_DYNAMIC_COMPONENTS displays information about the dynamic SGA components.

This view summarizes information based on all completed SGA resize operations since instance startup. All sizes are expressed in bytes.

Column	Datatype	Description
COMPONENT	VARCHAR2(64)	Component name
CURRENT_SIZE	NUMBER	Current size of the component
MIN_SIZE	NUMBER	Minimum size of the component since instance startup
MAX_SIZE	NUMBER	Maximum size of the component since instance startup
USER_SPECIFIED_SIZE	NUMBER	Value of the user parameter for the component
OPER_COUNT	NUMBER	Number of operations since instance startup
LAST_OPER_TYPE	VARCHAR2(13)	Last completed operation for the component: • STATIC • INITIALIZING • DISABLED • GROW • SHRINK • SHRINK_CANCEL
LAST_OPER_MODE	VARCHAR2(9)	Mode of the last completed operation: • MANUAL • DEFERRED • IMMEDIATE
LAST_OPER_TIME	DATE	Start time of the last completed operation
GRANULE_SIZE	NUMBER	Granularity of the GROW or SHRINK operation
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.79 V$MEMORY_RESIZE_OPS

V$MEMORY_RESIZE_OPS displays information about the last 800 completed memory resize operations (both automatic and manual). This does not include in-progress operations. All sizes are expressed in bytes.

Column	Datatype	Description
COMPONENT	VARCHAR2(64)	Component name

Column	Datatype	Description
OPER_TYPE	VARCHAR2(13)	Operation type: • STATIC • INITIALIZING • DISABLED • GROW • SHRINK • SHRINK_CANCEL
OPER_MODE	VARCHAR2(9)	Operation mode: • MANUAL • DEFERRED • IMMEDIATE
PARAMETER	VARCHAR2(80)	Name of the parameter for the resize operation
INITIAL_SIZE	NUMBER	Parameter value at the start of the operation
TARGET_SIZE	NUMBER	Requested value of the parameter after the resize
FINAL_SIZE	NUMBER	Real value of the parameter after the resize
RECEIVED_SIZE	NUMBER	Amount of memory that became available as a result of the resize
STATUS	VARCHAR2(9)	Completion status of the operation: • INACTIVE • PENDING • COMPLETE • CANCELLED • ERROR
START_TIMESTAMP	TIMESTAMP(6)	Timestamp for the start time of the operation
END_TIMESTAMP	TIMESTAMP(6)	Timestamp for the end time of the operation
START_TIME	DATE	Start time of the operation
END_TIME	DATE	End time of the operation
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.80 V$MEMORY_TARGET_ADVICE

V$MEMORY_TARGET_ADVICE provides information about how the MEMORY_TARGET parameter should be sized based on current sizing and satisfaction metrics.

Column	Datatype	Description
MEMORY_SIZE	NUMBER	If the MEMORY_SIZE_FACTOR column has a value of 1, then this column shows the current size of memory, as set by the MEMORY_TARGET initialization parameter. If the value of the MEMORY_SIZE_FACTOR column is less than or greater than 1, then this column shows a proposed memory size.

Column	Datatype	Description
MEMORY_SIZE_FACTOR	NUMBER	A multiplier for the current memory size. Possible values are 0.25, 0.5, 0.75, 1, 1.5, 1.75, and 2. This multiplier times the current memory size equals the value of the MEMORY_SIZE column.
ESTD_DB_TIME	NUMBER	For current memory size (MEMORY_SIZE_FACTOR = 1), the amount of database time required to complete the current workload. For a proposed memory size, the estimated amount of database time that would be required if the MEMORY_TARGET parameter were changed to the proposed size.
ESTD_DB_TIME_FACTOR	NUMBER	For a proposed memory size, ratio of estimated database time to current database time
VERSION	NUMBER	Version number of this recommendation (this snapshot of the V$MEMORY_TARGET_ADVICE view)
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

Table 9-2 shows how the information provided in V$MEMORY_TARGET_ADVICE could be used to improve performance. The data indicates that if current memory size is 380M, and you were to increase it to 760M (2x), the current workload would take 80525 units of DBtime as opposed to 115475 units of DBtime, which is a significant improvement in performance.

Table 9-2 Example of Using V$MEMORY_TARGET_ADVICE

MEMORY_SIZE	MEMORY_SIZE_FACTOR	ESTD_DB_TIME	ESTD_DB_TIME_FACTOR	VERSION
380	1	115475	1	3
95	.25	200500	1.7	3
190	.5	125600	1.1	3
760	2	80525	0.7	3

> **See Also:**
>
> "MEMORY_TARGET"

9.81 V$METRIC

V$METRIC displays the most recent statistic values for the complete set of metrics captured by the Automatic Workload Repository (AWR) infrastructure.

Column	Datatype	Description
BEGIN_TIME	DATE	Begin time of the interval

Column	Datatype	Description
END_TIME	DATE	End time of the interval
INTSIZE_CSEC	NUMBER	Interval size (in hundredths of a second)
GROUP_ID	NUMBER	Metric Group ID. Refer to the V$METRICNAME view for the name of the group.
ENTITY_ID	NUMBER	Entity ID for the metric in question. The value of the Entity ID depends upon the metric group. (See the following table for an explanation of possible values.)
ENTITY_SEQUENCE	NUMBER	Entity Sequence number for the metric in question. The value of the Entity Sequence depends upon the metric group. (See the following table for an explanation of possible values.)
METRIC_ID	NUMBER	Metric ID
METRIC_NAME	VARCHAR2(64)	Metric Name. This is the statistic that is captured for the entity.
VALUE	NUMBER	Value of the statistic between BEGIN_TIME and END_TIME
METRIC_UNIT	VARCHAR2(64)	Unit for the VALUE
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

The following table describes what the ENTITY_ID and ENTITY_SEQUENCE are for each metric group:

GID	Group NAME	Entity ID	Entity Sequence
0	Event Metrics	Event#	N/A
1	Event Class Metrics	Wait Class ID	N/A
2	System Metrics Long Duration	N/A	N/A
3	System Metrics Short Duration	N/A	N/A
4	Session Metrics Long Duration	Session ID	Serial#
5	Session Metrics Short Duration	Session ID	Serial#
6	Service Metrics	N/A	Service Hash
7	File Metrics Long Duration	File#	Creation Change#
9	Tablespace Metrics Long Duration	Tablespace#	N/A
10	Service Metrics (Short)	N/A	Service Hash

9.82 V$METRICGROUP

V$METRICGROUP displays information about the metric group for each of the four major Replication components: capture, propagation, apply, and queue.

Column	Datatype	Description
GROUP_ID	NUMBER	Internal ID associated with each group
NAME	VARCHAR2(64)	External name of the group
INTERVAL_SIZE	NUMBER	How often to collect statistics
MAX_INTERVAL	NUMBER	Total number of intervals over which statistics should be collected
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.83 V$METRIC_HISTORY

V$METRIC_HISTORY displays all the available statistic values for the complete set of metrics captured by the Automatic Workload Repository (AWR) infrastructure.

Column	Datatype	Description
BEGIN_TIME	DATE	Begin time of the interval
END_TIME	DATE	End time of the interval
INTSIZE_CSEC	NUMBER	Interval size (in hundredths of a second)
GROUP_ID	NUMBER	Metric Group ID. Refer to the V$METRICNAME view for the name of the group.
ENTITY_ID	NUMBER	Entity ID for the metric in question. The value of the Entity ID depends upon the metric group. (See the table in the description of V$METRIC for an explanation of possible values.)
ENTITY_SEQUENCE	NUMBER	Entity Sequence number for the metric in question. The value of the Entity Sequence depends upon the metric group. (See the table in the description of V$METRIC for an explanation of possible values.)
METRIC_ID	NUMBER	Metric ID
METRIC_NAME	VARCHAR2(64)	Metric Name. This is the statistic that is captured for the entity.
VALUE	NUMBER	Value of the statistic between BEGIN_TIME and END_TIME
METRIC_UNIT	VARCHAR2(64)	Unit for the VALUE
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.84 V$METRICNAME

V$METRICNAME displays the mapping of the name of metrics to their metric ID.

Column	Datatype	Description
GROUP_ID	NUMBER	Metric group ID
GROUP_NAME	VARCHAR2(64)	Metric group name
METRIC_ID	NUMBER	Metric ID
METRIC_NAME	VARCHAR2(64)	Metric name
METRIC_UNIT	VARCHAR2(64)	Unit of measurement
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.85 V$MTTR_TARGET_ADVICE

V$MTTR_TARGET_ADVICE displays rows that predict the number of physical I/Os for the MTTR corresponding to each row.

The rows also compute a physical I/O factor, which is the ratio of the number of estimated I/Os to the number of I/Os actually performed by the current MTTR setting during the measurement interval.

The content of the view is empty if MTTR advisory has not been turned on since database startup. Otherwise, it returns the advisory information collected. If advisory is currently off, then this information comes from the last time MTTR advisory was on. FAST_START_MTTR_TARGET must be set to a nonzero value if the STATISTICS_LEVEL parameter is dynamically modified to turn MTTR advisory on.

If the FAST_START_MTTR_TARGET parameter is changed while MTTR advisory is on, then MTTR advisory is temporarily turned off until the new FAST_START_MTTR_TARGET setting takes effect. During this transition period, the contents of V$MTTR_TARGET_ADVICE reflect the simulation result for the old MTTR setting.

Column	Datatype	Description
MTTR_TARGET_FOR_ESTIMATE	NUMBER	MTTR setting being simulated. Equal to the current MTTR setting if this is the first row of the view.
ADVICE_STATUS	VARCHAR2(5)	Current status of MTTR simulation: • ON • READY • OFF
DIRTY_LIMIT	NUMBER	Dirty buffer limit derived from the MTTR being simulated
ESTD_CACHE_WRITES	NUMBER	Estimated number of cache physical writes under this MTTR
ESTD_CACHE_WRITE_FACTOR	NUMBER	Estimated cache physical write ratio under this MTTR. It is the ratio of the estimated number of cache writes to the number of cache writes under the current MTTR setting.
ESTD_TOTAL_WRITES	NUMBER	Estimated total number of physical writes under this MTTR

Column	Datatype	Description
ESTD_TOTAL_WRITE_FACTOR	NUMBER	Estimated total physical write ratio under this MTTR. It is the ratio of the estimated total number of physical writes to the total number of physical writes under the current MTTR setting.
ESTD_TOTAL_IOS	NUMBER	Estimated total number of I/Os under this MTTR
ESTD_TOTAL_IO_FACTOR	NUMBER	Estimated total I/O ratio under this MTTR. It is the ratio of the estimated total number of I/Os to the total number of I/Os under the current MTTR setting.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

See Also:

- "FAST_START_MTTR_TARGET"
- "STATISTICS_LEVEL"

9.86 V$MUTEX_SLEEP

V$MUTEX_SLEEP shows the wait time, and the number of sleeps for each combination of mutex type and location.

Column	Datatype	Description
MUTEX_TYPE	VARCHAR2(32)	Type of action/object the mutex protects
LOCATION	VARCHAR2(40)	The code location where the waiter slept for the mutex
SLEEPS	NUMBER	Number of sleeps for this MUTEX_TYPE and LOCATION
WAIT_TIME	NUMBER	Wait time in microseconds
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.87 V$MUTEX_SLEEP_HISTORY

V$MUTEX_SLEEP_HISTORY displays time-series data.

Each row in this view is for a specific time, mutex type, location, requesting session and blocking session combination. That is, it shows data related to a specific session (requesting session) that slept while requesting a specific mutex type and location, because it was being held by a specific blocking session. The data in this view is contained within a circular buffer, with the most recent sleeps shown.

Column	Datatype	Description
MUTEX_IDENTIFIER	NUMBER	Mutex ID
SLEEP_TIMESTAMP	TIMESTAMP(6)	The last date/time this MUTEX_TYPE and LOCATION was slept for by the REQUESTING_SESSION, while being held by the BLOCKING_SESSION.
MUTEX_TYPE	VARCHAR2(32)	Type of action/object the mutex protects
GETS	NUMBER	The total number of gets since the mutex was created and up until the time of the wait (and from all sessions past and present)
SLEEPS	NUMBER	The number of times the requester had to sleep before obtaining the mutex
REQUESTING_SESSION	NUMBER	The SID of a session requesting the mutex
BLOCKING_SESSION	NUMBER	The SID of a session holding the mutex
LOCATION	VARCHAR2(40)	The code location where the waiter slept for the mutex
MUTEX_VALUE	RAW(4 | 8)	If the mutex is held in exclusive (X) mode, this column shows the SID of the blocking session, else it shows the number of sessions referencing the mutex in S mode.
P1	NUMBER	Reserved for internal use
P1RAW	RAW(4 | 8)	Reserved for internal use
P2	NUMBER	Reserved for internal use
P3	NUMBER	Reserved for internal use
P4	NUMBER	Reserved for internal use
P5	VARCHAR2(64)	Reserved for internal use
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.88 V$MVREFRESH

V$MVREFRESH displays information about the materialized views currently being refreshed.

Column	Datatype	Description
SID	NUMBER	Session identifier

Column	Datatype	Description
SERIAL#	NUMBER	Session serial number, which is used to uniquely identify a session's objects. Guarantees that session-level commands are applied to the correct session objects if the session ends with, and another session begins with, the same session ID.
CURRMVOWNER	VARCHAR2(31)	Owner of the materialized view currently being refreshed. The materialized view resides in this user's schema.
CURRMVNAME	VARCHAR2(31)	Name of the materialized view currently being refreshed
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.89 V$MYSTAT

V$MYSTAT contains statistics on the current session.

Column	Datatype	Description
SID	NUMBER	ID of the current session
STATISTIC#	NUMBER	Number of the statistic
VALUE	NUMBER	Value of the statistic
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.90 V$NFS_CLIENTS

V$NFS_CLIENTS displays information about NFS clients currently connected to the XML DB NFS Server.

Column	Datatype	Description
CLIENTID	NUMBER	A number identifying the client
PRINCIPAL	VARCHAR2(2000)	User string denoting the principal that set the client ID (SetClientId)
CLIENTOPAQUEIDENTIFIER	VARCHAR2(1000)	Opaque string presented as identification by the client to the NFS server
VERIFIER	RAW(8)	Verifier presented by the client
LEASEEXPIRY	NUMBER	Number of seconds in which the lease expires for the client
CLIENTADDR	VARCHAR2(2000)	Address of the client

Column	Datatype	Description
CONFIRMED	VARCHAR2(5)	TRUE if the client is confirmed; otherwise FALSE
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.91 V$NFS_LOCKS

V$NFS_LOCKS displays information about byte range locks held on different files by NFS clients.

Column	Datatype	Description
OPENSTATEID	RAW(16)	Open state ID of the open owner
OPENSEQUENCEID	NUMBER	Open Sequence ID of the open owner
LOCKSTATEID	RAW(16)	Lock state ID of the lock owner
LOCKSEQUENCEID	NUMBER	Lock sequence ID of the lock owner
LOCKOWNER	VARCHAR2(2000)	Opaque string presented as identification by the lock owner to the NFS server
OFFSET	NUMBER	Byte Offset from which lock starts
LENGTH	NUMBER	Length of the lock
LOCKTYPE	VARCHAR2(20)	Type of the lock
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.92 V$NFS_OPEN_FILES

V$NFS_OPEN_FILES displays information about all the files currently opened by clients at the NFS server.

Column	Datatype	Description
CLIENTID	NUMBER	Number identifying the client
OPENOWNEROPAQUE	VARCHAR2(2000)	All the files currently opened by clients at the NFS server
OPENSTATEID	RAW(16)	Open state ID of the open owner
FILEHANDLE	RAW(32)	FileHandle of the file that has been opened
OPENSEQUENCEID	NUMBER	Open sequence ID of open owner
OPENREAD	VARCHAR2(5)	TRUE if the file is open for READ operations; otherwise FALSE

Column	Datatype	Description
OPENWRITE	VARCHAR2(5)	TRUE if the file is open for WRITE operations; otherwise FALSE
SHAREACCESS	VARCHAR2(15)	Sharing mode of the file (SharedReadWrite, SharedRead, SharedWrite)
SHAREDENY	VARCHAR2(13)	Deny mode of the file (DenyReadWrite, DenyRead, DenyWrite)
CONFIRMED	VARCHAR2(5)	TRUE if open is confirmed; otherwise FALSE
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.93 V$NLS_PARAMETERS

V$NLS_PARAMETERS contains current values of NLS parameters.

Column	Datatype	Description
PARAMETER	VARCHAR2(64)	Parameter names are as follows: NLS_CALENDAR, NLS_CHARACTERSET, NLS_COMP, NLS_CURRENCY, NLS_DATE_FORMAT, NLS_DATE_LANGUAGE, NLS_DUAL_CURRENCY, NLS_ISO_CURRENCY, NLS_LANGUAGE, NLS_LENGTH_SEMANTICS, NLS_NCHAR_CHARACTERSET, NLS_NCHAR_CONV_EXCP, NLS_NUMERIC_CHARACTERS, NLS_SORT, NLS_TERRITORY, NLS_TIMESTAMP_FORMAT, and NLS_TIMESTAMP_TZ_FORMAT. Two additional parameters, NLS_TIME_FORMAT and NLS_TIME_TZ_FORMAT, are currently used for internal purposes only.
VALUE	VARCHAR2(64)	NLS parameter value
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.94 V$NLS_VALID_VALUES

V$NLS_VALID_VALUES lists all valid values for NLS parameters.

Column	Datatype	Description
PARAMETER	VARCHAR2(64)	Parameter name (LANGUAGE | SORT | TERRITORY | CHARACTERSET)
VALUE	VARCHAR2(64)	NLS parameter value
ISDEPRECATED	VARCHAR2(5)	Indicates whether the parameter has been deprecated (TRUE) or not (FALSE)

Column	Datatype	Description
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.95 V$NONLOGGED_BLOCK

V$NONLOGGED_BLOCK displays ranges of nonlogged datafile blocks recorded in the control file.

Prior to Oracle Database 12*c*, the presence of any nonlogged blocks in a data file was recorded in the file header via the FIRST_NONLOGGED_SCN column of the V$DATAFILE view. Now with 12*c*, in addition to the file header data, the ranges themselves are recorded in the control file. A control file range is a superset of the actual nonlogged blocks, meaning that small ranges can be merged to form larger ranges, even when there are some valid blocks between the smaller ranges.

The information in the view is maintained by RMAN VALIDATE, RMAN RESTORE, RMAN RECOVER, and Flashback Database and Media Recovery. A non RMAN-based restore will cause the data to become invalid, and it will be purged the next time any of those tasks are invoked and involve the file. As a result of space reuse, it is possible for ranges to no longer contain any nonlogged blocks. An RMAN VALIDATE command can be used to synchronize the ranges with the actual nonlogged blocks found from a scan of the data file.

Column	Datatype	Description
FILE#	NUMBER	Absolute file number of the data file that contains the nonlogged blocks
BLOCK#	NUMBER	Block number of the first nonlogged block in the range of nologged blocks
BLOCKS	NUMBER	Number of nonlogged blocks found starting with BLOCK#
NONLOGGED_START_CHANGE#	NUMBER	The smallest SCN on which any block in this block range became nonlogged. NULL if unknown.
NONLOGGED_START_TIME	DATE	The time that corresponds to NONLOGGED_START_CHANGE#. NULL if unknown.
NONLOGGED_END_CHANGE#	NUMBER	The largest SCN on which any block in this block range became nonlogged. NULL if unknown.
NONLOGGED_END_TIME	DATE	The time that corresponds to NONLOGGED_END_CHANGE#. NULL if unknown.
RESETLOGS_CHANGE#	NUMBER	The resetlogs SCN of the incarnation on which this block range was first marked as nonlogged. NULL if unknown.
RESETLOGS_TIME	DATE	The resetlogs time of the incarnation on which this block range was first marked as nologged. NULL if unknown.
OBJECT#	VARCHAR2(40)	The object ID this range belongs to. If this field is NULL, the object number is unknown.

Column	Datatype	Description
REASON	VARCHAR2(9)	The reason why this block range appears in this list, for example, primary file offline, could not talk to primary, non-standby recovery, and so on. For Oracle Database 12*c* and later releases, it is always UNKNOWN.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.96 V$OBJECT_DEPENDENCY

V$OBJECT_DEPENDENCY displays the objects depended on by a package, procedure, or cursor that is currently loaded in the shared pool. For example, together with V$SESSION and V$SQL, this view can be used to determine which tables are used in the SQL statement that a user is currently executing.

> **See Also:**
>
> "V$SESSION" and "V$SQL"

Column	Datatype	Description
FROM_ADDRESS	RAW(4 | 8)	Address of a procedure, package, or cursor that is currently loaded in the shared pool
FROM_HASH	NUMBER	Hash value of a procedure, package, or cursor that is currently loaded in the shared pool
TO_OWNER	VARCHAR2(64)	Owner of the object that is depended on
TO_NAME	VARCHAR2(1000)	Name of the object that is depended on
TO_ADDRESS	RAW(4 | 8)	Address of the object that is depended on. These can be used to look up more information on the object in V$DB_OBJECT_CACHE.
TO_HASH	NUMBER	Hash value of the object that is depended on. These can be used to look up more information on the object in V$DB_OBJECT_CACHE.
TO_TYPE	NUMBER	Type of the object that is depended on
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.97 V$OBJECT_PRIVILEGE

V$OBJECT_PRIVILEGE displays information about privileges associated with an object.

Column	Datatype	Description
OBJECT_TYPE_NAME	VARCHAR2(64)	Name of the object type
OBJECT_TYPE_ID	NUMBER	ID of the object type
PRIVILEGE_ID	NUMBER	ID of the privilege
PRIVILEGE_NAME	VARCHAR2(64)	Name of the privilege
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.98 V$OBSOLETE_BACKUP_FILES

V$OBSOLETE_BACKUP_FILES displays all obsolete backups, copies, and archived logs according to the current retention policy.

This view requires that the database is set using the DBMS_RCVMAN.SETDATABASE procedure.

Column	Datatype	Description
PKEY	NUMBER	Primary key for the backup
BACKUP_TYPE	VARCHAR2(32)	Type of the backup: • BACKUP SET • COPY • PROXY COPY
FILE_TYPE	VARCHAR2(32)	Type of the file: • DATAFILE • CONTROLFILE • SPFILE • REDO LOG • PIECE
KEEP	VARCHAR2(3)	Indicates whether the backup has a retention policy different from the value for CONFIGURE RETENTION POLICY (YES) or not (NO)
KEEP_UNTIL	DATE	If the KEEP UNTIL TIME clause of the BACKUP command was specified, then this column shows the date after which the backup becomes obsolete. If the column is null and KEEP_OPTIONS is not null, the backup never becomes obsolete.
KEEP_OPTIONS	VARCHAR2(13)	KEEP options for the backup: • LOGS - RMAN keeps the logs needed to recover the backup • NOLOGS - RMAN does not keep the logs needed to recover the backup If this column is null, then the backup has no KEEP options and will be made obsolete based on the retention policy.

Column	Datatype	Description
STATUS	VARCHAR2(16)	Status of the backup: • AVAILABLE • UNAVAILABLE • EXPIRED • OTHER
FNAME	VARCHAR2(1024)	Name of the file
TAG	VARCHAR2(32)	Tag of the piece, copy, or proxy copy
MEDIA	VARCHAR2(80)	Media ID of the piece or proxy copy
RECID	NUMBER	Recid of the record in the controlfile
STAMP	NUMBER	Stamp of the record in the controlfile
DEVICE_TYPE	VARCHAR2(255)	Type of media device that stores the backup
BLOCK_SIZE	NUMBER	Block size for the backup (in bytes)
COMPLETION_TIME	DATE	Time when the backup completed
BS_KEY	NUMBER	Primary key of the backup set (valid only when BACKUP_TYPE is BACKUP SET)
BS_COUNT	NUMBER	Count of the backup set from the controlfile record (valid only when BACKUP_TYPE is BACKUP SET)
BS_STAMP	NUMBER	Stamp of the backup set from the controlfile record (valid only when BACKUP_TYPE is BACKUP SET)
BS_TYPE	VARCHAR2(32)	Type of the backup set (valid only when BACKUP_TYPE is BACKUP SET): • DATAFILE • ARCHIVED LOG
BS_INCR_TYPE	VARCHAR2(32)	Incremental level of the backup set (valid only when BACKUP_TYPE is BACKUP SET)
BS_PIECES	NUMBER	Number of backup pieces in the backup set (valid only when BACKUP_TYPE is BACKUP SET)
BS_COMPLETION_TIME	DATE	Completion time of the backup set (valid only when BACKUP_TYPE is BACKUP SET)
BP_PIECE#	NUMBER	Number of the backup piece (valid only when FILE_TYPE is PIECE and BACKUP_TYPE is BACKUP SET)
BP_COPY#	NUMBER	Copy number of the backup piece (valid only when FILE_TYPE is PIECE and BACKUP_TYPE is BACKUP SET)
DF_FILE#	NUMBER	Absolute file number of the datafile (valid only when FILE_TYPE is DATAFILE)
DF_RESETLOGS_CHANGE#	NUMBER	System change number (SCN) of the most recent RESETLOGS when the control file or datafile was created (valid only when FILE_TYPE is DATAFILE)
DF_CREATION_CHANGE#	NUMBER	Creation SCN of the control file or datafile (valid only when FILE_TYPE is DATAFILE)
DF_CHECKPOINT_CHANGE#	NUMBER	System change number (SCN) of the most recent control file or datafile checkpoint (valid only when FILE_TYPE is DATAFILE)
DF_CKP_MOD_TIME	DATE	Modification time in case of SPFILE, otherwise time when the control file or datafile was checkpointed (valid only when FILE_TYPE is DATAFILE)
RL_THREAD#	NUMBER	Number of the redo thread (valid only when FILE_TYPE is REDO LOG)

Column	Datatype	Description
RL_SEQUENCE#	NUMBER	Log sequence number (valid only when FILE_TYPE is REDO LOG)
RL_RESETLOGS_CHANGE#	NUMBER	System change number (SCN) of the most recent RESETLOGS when the record was created (valid only when FILE_TYPE is REDO LOG)
RL_FIRST_CHANGE#	NUMBER	First SCN of the redo log (valid only when FILE_TYPE is REDO LOG)
RL_FIRST_TIME	DATE	Time when Oracle switched into the redo log (valid only when FILE_TYPE is REDO LOG)
RL_NEXT_CHANGE#	NUMBER	First SCN of the next redo log in the thread (valid only when FILE_TYPE is REDO LOG)
RL_NEXT_TIME	DATE	First timestamp of the next redo log in the thread (valid only when FILE_TYPE is REDO LOG)

See Also:

Oracle Database Backup and Recovery User's Guide for more information about the DBMS_RCVMAN.SETDATABASE procedure

9.99 V$OBSOLETE_PARAMETER

V$OBSOLETE_PARAMETER displays information about obsolete initialization parameters. If any row of the view contains TRUE in the ISSPECIFIED column, then you should examine why.

Column	Datatype	Description
NAME	VARCHAR2(64)	Name of the parameter
ISSPECIFIED	VARCHAR2(5)	Indicates whether the parameter was specified in the parameter file (TRUE) or not (FALSE)
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.100 V$OFFLINE_RANGE

V$OFFLINE_RANGE displays datafile offline information from the control file.

Note that the last offline range of each datafile is kept in the DATAFILE record.

An offline range is created for a datafile when its tablespace is first altered to be OFFLINE NORMAL or READ ONLY, and then subsequently altered to be ONLINE or read/write. Note that no offline range is created if the datafile itself is altered to be OFFLINE or if the tablespace is altered to be OFFLINE IMMEDIATE.

> **See Also:**
>
> "V$DATAFILE"

Column	Datatype	Description
RECID	NUMBER	Record ID
STAMP	NUMBER	Record stamp
FILE#	NUMBER	Datafile number
OFFLINE_CHANGE#	NUMBER	SCN at which offlined
ONLINE_CHANGE#	NUMBER	SCN at which onlined
ONLINE_TIME	DATE	Time of offline SCN
RESETLOGS_CHANGE#	NUMBER	Resetlogs change number of the record
RESETLOGS_TIME	DATE	Resetlogs timestamp of the record
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.101 V$OFS_STATS

V$OFS_STATS displays performance statistics for various Oracle File System operations. These statistics are maintained at per mount level.

Column	Datatype	Description
OFS_MNTPNT	VARCHAR2(4096)	Mount point
OFS_LOOKUP	NUMBER	Number of lookup operations performed
OFS_FORGET	NUMBER	Number of forget operations performed
OFS_GETATTR	NUMBER	Number of getattr operations
OFS_SETATTR	NUMBER	Number of setattr operations
OFS_READLINK	NUMBER	Number of readlink operations
OFS_SYMLINK	NUMBER	Number of symlink operations
OFS_MKNOD	NUMBER	Number of mknod operations
OFS_MKDIR	NUMBER	Number of mkdir operations
OFS_UNLINK	NUMBER	Number of remove file operations
OFS_RMDIR	NUMBER	Number of remove directory operations
OFS_RENAME	NUMBER	Number of file rename operations
OFS_LINK	NUMBER	Number of hard link operations
OFS_OPEN	NUMBER	Number of file open operations
OFS_READ	NUMBER	Number of file read operations

Column	Datatype	Description
OFS_WRITE	NUMBER	Number of file write operations
OFS_STATFS	NUMBER	Number of statfs operations performed
OFS_RELEASE	NUMBER	Number of release operations
OFS_FSYNC	NUMBER	Number of file sync operations
OFS_SETXATTR	NUMBER	Number of set extended attributes operations
OFS_GETXATTR	NUMBER	Number of get extended attributes
OFS_LISTXATTR	NUMBER	Number of list extended attributes
OFS_REMOVEXATTR	NUMBER	Number of remove extended attributes
OFS_FLUSH	NUMBER	Number of flush operations
OFS_INIT	NUMBER	Number of init operations
OFS_OPENDIR	NUMBER	Number of opendir operations
OFS_READDIR	NUMBER	Number of readdir operations
OFS_RELEASEDIR	NUMBER	Number of releasedir operations
OFS_FSYNCDIR	NUMBER	Number of directory sync operations
OFS_GETLK	NUMBER	Number of file get lock operations
OFS_SETLK	NUMBER	Number of file lock operations
OFS_SETLKW	NUMBER	Number of file lock operation with wait option
OFS_ACCESS	NUMBER	Number of access operations
OFS_CREATE	NUMBER	Number of create operations
OFS_INTERRUPT	NUMBER	Number of interrupt operations received on the mount
OFS_BMAP	NUMBER	Number of block map operations received
OFS_DESTROY	NUMBER	Number of destroy operations
OFS_BYTES_READ	NUMBER	Total number of bytes read on the mount point
OFS_BYTES_WRITTEN	NUMBER	Total number of bytes written to the mount point
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

Note:

This database view is supported only on the Linux operating system.

See Also:

"V$OFSMOUNT"

9.102 V$OFS_THREADS

V$OFS_THREADS displays information about Oracle file system (OFS) threads.

Column	Datatype	Description
PDBID	NUMBER	ID of the PDB in which the thread was created
FSID	NUMBER	ID of the mounted file system on which the thread is working
FSNAME	VARCHAR2(1024)	Name of the mounted file system on which the thread is working
FILE_NAME	VARCHAR2(1024)	File, directory, or filesystem on which the thread is working The value of this column depends on the value of the REQ_TYPE column: • If REQ_TYPE = SYSREQ, this column displays the path to the mount point of the filesystem on which the thread is working. • If REQ_TYPE = FILEREQ, this column displays the name of the file or directory on which the thread is working. • If REQ_TYPE = NONE, the value of this column is null.
THREADID	VARCHAR2(24)	ID of the thread
THREAD_TYPE	VARCHAR2(24)	Type of thread: • WORKER - The thread is used for executing file, directory, or filesystem requests. • RECEIVER - The thread receives requests from all OFS mounted file systems.
THREAD_NAME	VARCHAR2(24)	Name of the thread
REQ_TYPE	VARCHAR2(1024)	Type of request: • SYSREQ • FILEREQ • NONE
OPCODE	VARCHAR2(1024)	Type of operation on which the thread is working The value of this column depends on the value of the REQ_TYPE column: • If REQ_TYPE = SYSREQ, this column displays a DBMS_FS operation. • If REQ_TYPE = FILEREQ, this column displays any type of file operation. • If REQ_TYPE = NONE, the value of this column is null.
ELAPSED_TIME_MICRO	NUMBER	If REQ_TYPE = SYSREQ, the amount of time (in microseconds) that has elapsed since the current request was queued
SYSRQ_COUNT	NUMBER	Total number of REQ_TYPE = SYSREQ requests completed since the instance was started
FREQ_COUNT	NUMBER	Total number of REQ_TYPE = FILEREQ requests completed since the instance was started
WAIT_EVENT	VARCHAR2(64)	Event on which the thread is waiting
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

> **Note:**
>
> This view is available starting with Oracle Database 23ai.

9.103 V$OFSMOUNT

`V$OFSMOUNT` provides information about the file systems that are mounted by Oracle File System.

Column	Datatype	Description
`OFS_MNTPATH`	`VARCHAR2(1024)`	Mount path where the file system is mounted
`OFS_FSPATH`	`VARCHAR2(1024)`	File system path
`OFS_MNTOPTS`	`VARCHAR2(1024)`	Mount options used to mount the file system
`OFS_MNTFLAGS`	`VARCHAR2(7)`	Flags to specify if the file system is mounted. A value of 1 indicates that the file system is mounted.
`CON_ID`	`NUMBER`	The ID of the container to which the data pertains. Possible values include: • `0`: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • `1`: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data
`OFS_NODENM`	`VARCHAR2(255)`	Node name
`OFS_FSID`	`NUMBER`	Unique ID that identifies the mounted file system
`OFS_FSTYPE`	`VARCHAR2(255)`	Oracle file system type. This is the value that is passed to the `dbms_fs.mount_oracle_fs()` procedure. Some of the expected values are `dbfs` and `ofs`.

> **Note:**
>
> This database view is supported only on the Linux operating system.

> **See Also:**
>
> - "V$OFS_STATS"
> - *Oracle Database PL/SQL Packages and Types Reference* for more information about the `DBMS_FS.MOUNT_ORACLE_FS` procedure

9.104 V$ONLINE_REDEF

`V$ONLINE_REDEF` provides information about the status of currently running online redefinitions.

Column	Datatype	Description
SID	NUMBER	Session identifier
SERIAL#	NUMBER	Session serial number, which is used to uniquely identify a session's objects
REDEFINITION_ID	NUMBER	Redefinition identifier
TABLE_OWNER	VARCHAR2(129)	Owner of the table currently being redefined. The table resides in this user's schema.
ORIGINAL_TABLE_NAME	VARCHAR2(129)	Name of the original table
INTERIM_TABLE_NAME	VARCHAR2(1024)	Interim table currently being redefined
PARTITION_NAME	VARCHAR2(1024)	Partition of the table currently being redefined. The table resides in this user's schema.
OPERATION	VARCHAR2(128)	Operations during the redefining process: • start_redef_table • sync_interim_table • copy_table_dependents • finish_redef_table • abort_redef_table
SUBOPERATION	VARCHAR2(128)	Sub operation during the redefining process
DETAILED_MESSAGE	VARCHAR2(1024)	Details of operations during redefining process. This can include details such as the number of DML being processed, execution start_time, and partition_name
PROGRESS	VARCHAR2(128)	Percentage of completion of each operation
REFRESH_STATEMENT_SQL_ID	VARCHAR2(128)	The SQL ID for the statement in the REFRESH_STATEMENT column
REFRESH_STATEMENT	VARCHAR2(4000)	A refresh statement executed during a refresh operation in some online redefinition procedures. When refresh is a sub operation in the SUBOPERATION column, there will be a refresh statement in this REFRESH_STATEMENT column.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.105 V$OPEN_CURSOR

V$OPEN_CURSOR lists cursors that each user session currently has opened and parsed, or cached.

Column	Datatype	Description
SADDR	RAW(4 | 8)	Session address
SID	NUMBER	Session identifier
USER_NAME	VARCHAR2(128)	User that is logged in to the session
ADDRESS	RAW(4 | 8)	Used with HASH_VALUE to uniquely identify the SQL statement being executed in the session

Column	Datatype	Description
HASH_VALUE	NUMBER	Used with ADDRESS to uniquely identify the SQL statement being executed in the session
SQL_ID	VARCHAR2(13)	SQL identifier of the SQL statement being executed in the session
SQL_TEXT	VARCHAR2(60)	First 60 characters of the SQL statement that is parsed into the open cursor
LAST_SQL_ACTIVE_TIME	DATE	Time when this cursor was last executed
SQL_EXEC_ID	NUMBER	If the open cursor is executing, then the SQL execution identifier for that execution (see V$SQL_MONITOR)
CURSOR_TYPE	VARCHAR2(64)	Type of cursor: • OPEN PL/SQL - Open PL/SQL cursors • OPEN - Other open cursors • SESSION CURSOR CACHED - Cursors cached in the generic session cursor cache • OPEN-RECURSIVE - Open recursive cursors • DICTIONARY LOOKUP CURSOR CACHED - Cursors cached in the dictionary lookup cursor cache • BUNDLE DICTIONARY LOOKUP CACHED - Cursors cached in the bundled dictionary lookup cursor cache • JAVA NAME TRANSLATION CURSOR CACHED - Cursors cached in the Java name translation cursor cache • REPLICATION TRIGGER CURSOR CACHED - Cursors cached in the replication trigger cursor cache • CONSTRAINTS CURSOR CACHED - Cursors cached in the constraints cursor cache • PL/SQL CURSOR CACHED - Cursors cached in the PL/SQL cursor cache
CHILD_ADDRESS	RAW(4 | 8)	Address of the child cursor
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.106 V$OPTIMIZER_PROCESSING_RATE

V$OPTIMIZER_PROCESSING_RATE displays the processing rates used by the optimizer to compute degree of parallelism.

> **Note:**
>
> You can manipulate these rates using these procedures for the DBMS_STATS package:
>
> - SET_PROCESSING_RATE
> - DELETE_PROCESSING_RATE
> - GATHER_PROCESSING_RATE
>
> See *Oracle Database PL/SQL Packages and Types Reference* for more information about the DBMS_STATS package.

Column	Datatype	Description
ADDRESS	RAW(4| 8)	Address of the handle to the parent for this cursor
OPERATION_NAME	VARCHAR2(64)	Name of the operation. The possible values are AGGR, ALL, CPU, CPU_ACCESS, CPU_AGGR, CPU_BYTES_PER_SEC, CPU_FILTER, CPU_GBY, CPU_HASH_JOIN, CPU_JOIN, CPU_NL_JOIN, CPU_RANDOM_ACCESS, CPU_ROWS_PER_SEC, CPU_SEQUENTIAL_ACCESS, CPU_SM_JOIN, CPU_SORT, HASH, IO, IO_ACCESS, IO_BYTES_PER_SEC, IO_RANDOM_ACCESS, IO_ROWS_PER_SEC, IO_SEQUENTIAL_ACCESS, MEMCMP, MEMCPY
MANUAL_VALUE	VARCHAR2(10)	Value of the operation set manually by the user
CALIBRATION_VALUE	VARCHAR2(10)	Value of the operation obtained from calibration (by running the GATHER_PROCESSING_RATE procedure)
DEFAULT_VALUE	VARCHAR2(10)	Default value of the operation
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.107 V$OPTION

V$OPTION displays Oracle Database options and features.

Typically, although not always, options must be separately licensed, whereas features come with the product and are enabled based on the product that is running (for example, Enterprise Edition).

Column	Datatype	Description
PARAMETER	VARCHAR2(64)	Name of the option (or feature)

Column	Datatype	Description
VALUE	VARCHAR2(64)	Indicates whether the option (or feature) is installed (TRUE) or not (FALSE)
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

> **See Also:**
>
> *Oracle Database Licensing Information User Manual* for more information about Oracle Database options and features

9.108 V$OSSTAT

V$OSSTAT displays system utilization statistics from the operating system. One row is returned for each system statistic.

Column	Datatype	Description
STAT_NAME	VARCHAR2(64)	Name of the statistic (see Table 9-3)
VALUE	NUMBER	Instantaneous statistic value
OSSTAT_ID	NUMBER	Statistic ID
COMMENTS	VARCHAR2(64)	Any additional operating system-specific clarifications for the statistic
CUMULATIVE	VARCHAR2(3)	Indicates whether the statistic is cumulative (that is, accumulates over time) (YES) or not (NO)
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

Table 9-3 V$OSSTAT Statistics

Statistic Name	Description
NUM_CPUS	Number of CPUs or processors available
IDLE_TIME	Number of hundredths of a second that a processor has been idle, totalled over all processors
BUSY_TIME	Number of hundredths of a second that a processor has been busy executing user or kernel code, totalled over all processors

Table 9-3 (Cont.) V$OSSTAT Statistics

Statistic Name	Description
USER_TIME	Number of hundredths of a second that a processor has been busy executing user code, totalled over all processors
SYS_TIME	Number of hundredths of a second that a processor has been busy executing kernel code, totalled over all processors
IOWAIT_TIME	Number of hundredths of a second that a processor has been waiting for I/O to complete, totalled over all processors
NICE_TIME	Number of hundredths of a second that a processor has been busy executing low-priority user code, totalled over all processors
AVG_IDLE_TIME	Number of hundredths of a second that a processor has been idle, averaged over all processors
AVG_BUSY_TIME	Number of hundredths of a second that a processor has been busy executing user or kernel code, averaged over all processors
AVG_USER_TIME	Number of hundredths of a second that a processor has been busy executing user code, averaged over all processors
AVG_SYS_TIME	Number of hundredths of a second that a processor has been busy executing kernel code, averaged over all processors
AVG_IOWAIT_TIME	Number of hundredths of a second that a processor has been waiting for I/O to complete, averaged over all processors
AVG_NICE_TIME	Number of hundredths of a second that a processor has been busy executing low-priority user code, averaged over all processors
OS_CPU_WAIT_TIME	Total number of hundredths of a second that processes have been in a ready state, waiting to be selected by the operating system scheduler to run
RSRC_MGR_CPU_WAIT_TIME	Total number of hundredths of a second that Oracle processes have been in a ready state, waiting for CPU to be available for their consumer group in the currently active resource plan
VM_IN_BYTES	Total number of bytes of data that have been paged in due to virtual memory paging
VM_OUT_BYTES	Total number of bytes of data that have been paged out due to virtual memory paging
PHYSICAL_MEMORY_BYTES	Total number of bytes of physical memory
LOAD	Current number of processes that are either running or in the ready state, waiting to be selected by the operating-system scheduler to run. On many platforms, this statistic reflects the average load over the past minute.
NUM_CPU_CORES	Number of CPU cores available (includes subcores of multicore CPUs as well as single-core CPUs)
NUM_CPU_SOCKETS	Number of CPU sockets available (represents an absolute count of CPU chips on the system, regardless of multithreading or multi-core architectures)
NUM_VCPUS	Number of virtual CPUs available
NUM_LCPUS	Number of logical CPUs available (includes hardware threads if hardware threading is turned on)
TCP_SEND_SIZE_MIN	Minimum size of the TCP send buffer
TCP_SEND_SIZE_DEFAULT	Default size of the TCP send buffer
TCP_SEND_SIZE_MAX	Maximum size of the TCP send buffer
TCP_RECEIVE_SIZE_MIN	Minimum size of the TCP receive buffer
TCP_RECEIVE_SIZE_DEFAULT	Default size of the TCP receive buffer
TCP_RECEIVE_SIZE_MAX	Maximum size of the TCP receive buffer

Table 9-3 (Cont.) V$OSSTAT Statistics

Statistic Name	Description
GLOBAL_SEND_SIZE_MAX	Maximum size of the global send buffer
GLOBAL_RECEIVE_SIZE_MAX	Maximum size of the global receive buffer

> **Note:**
>
> The availability of all statistics except for `NUM_CPUS` and `RSRC_MGR_CPU_WAIT_TIME` is subject to the operating system platform on which the Oracle Database is running.

9.109 V$PARALLEL_DEGREE_LIMIT_MTH

`V$PARALLEL_DEGREE_LIMIT_MTH` displays all available parallel degree limit resource allocation methods.

Column	Datatype	Description
`NAME`	`VARCHAR2(40)`	Name of the parallel degree limit resource allocation method
`CON_ID`	`NUMBER`	The ID of the container to which the data pertains. Possible values include: • `0`: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • `1`: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.110 V$PARAMETER

`V$PARAMETER` displays information about the initialization parameters that are currently in effect for the session. A new session inherits parameter values from the instance-wide values displayed by the `V$SYSTEM_PARAMETER` view.

Column	Datatype	Description
`NUM`	`NUMBER`	Parameter number
`NAME`	`VARCHAR2(80)`	Name of the parameter
`TYPE`	`NUMBER`	Parameter type: • `1` - Boolean • `2` - String • `3` - Integer • `4` - Parameter file • `5` - Reserved • `6` - Big integer
`VALUE`	`VARCHAR2(4000)`	Parameter value for the session (if modified within the session); otherwise, the instance-wide parameter value

Column	Datatype	Description
DISPLAY_VALUE	VARCHAR2(4000)	Parameter value in a user-friendly format. For example, if the VALUE column shows the value 262144 for a big integer parameter, then the DISPLAY_VALUE column will show the value 256K.
DEFAULT_VALUE	VARCHAR2(255)	The default value for this parameter. This is the value of the parameter if a value is not explicitly specified for the parameter.
ISDEFAULT	VARCHAR2(9)	Indicates whether the parameter is set to the default value (TRUE) or the parameter value was specified in the parameter file (FALSE). The database sets the value of the ISDEFAULT column to TRUE for parameters that are not specified in the init.ora or server parameter file (SPFILE).
ISSES_MODIFIABLE	VARCHAR2(5)	Indicates whether the parameter can be changed with ALTER SESSION (TRUE) or not (FALSE)
ISSYS_MODIFIABLE	VARCHAR2(9)	Indicates whether the parameter can be changed with ALTER SYSTEM and when the change takes effect: • IMMEDIATE - Parameter can be changed with ALTER SYSTEM regardless of the type of parameter file used to start the instance. The change takes effect immediately. • DEFERRED - Parameter can be changed with ALTER SYSTEM regardless of the type of parameter file used to start the instance. The change takes effect in subsequent sessions. • FALSE - Parameter cannot be changed with ALTER SYSTEM unless a server parameter file was used to start the instance. The change takes effect in subsequent instances.
ISPDB_MODIFIABLE	VARCHAR2(5)	Indicates whether the parameter can be modified inside a PDB (TRUE) or not (FALSE)
ISINSTANCE_MODIFIABLE	VARCHAR2(5)	For parameters that can be changed with ALTER SYSTEM, indicates whether the value of the parameter can be different for every instance (TRUE) or whether the parameter must have the same value for all Real Application Clusters instances (FALSE). If the ISSYS_MODIFIABLE column is FALSE, then this column is always FALSE.
ISMODIFIED	VARCHAR2(10)	Indicates whether the parameter has been modified after instance startup: • MODIFIED - Parameter has been modified with ALTER SESSION • SYSTEM_MOD - Parameter has been modified with ALTER SYSTEM (which causes all the currently logged in sessions' values to be modified) • FALSE - Parameter has not been modified after instance startup
ISADJUSTED	VARCHAR2(5)	Indicates whether Oracle adjusted the input value to a more suitable value (for example, the parameter value should be prime, but the user input a non-prime number, so Oracle adjusted the value to the next prime number)
ISDEPRECATED	VARCHAR2(5)	Indicates whether the parameter has been deprecated (TRUE) or not (FALSE)
ISBASIC	VARCHAR2(5)	Indicates whether the parameter is a basic parameter (TRUE) or not (FALSE)
DESCRIPTION	VARCHAR2(255)	Description of the parameter
UPDATE_COMMENT	VARCHAR2(255)	Comments associated with the most recent update
HASH	NUMBER	Hash value for the parameter name

Column	Datatype	Description
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

Examples

The following query returns the default value for the ALLOW_GLOBAL_DBLINKS initialization parameter:

```
SQL> SELECT name, default_value FROM v$parameter
  2  WHERE name = 'allow_global_dblinks';

NAME
--------------------------------------------------------------
DEFAULT_VALUE
--------------------------------------------------------------
allow_global_dblinks
FALSE


SQL>
```

The following query shows that the ALLOW_GLOBAL_DBLINKS initialization parameter is not modifiable in a PDB:

```
SQL> SELECT name, ispdb_modifiable FROM v$parameter
  2  WHERE name = 'allow_global_dblinks';

NAME
--------------------------------------------------------------
ISPDB
-----
allow_global_dblinks
FALSE


SQL>
```

> **See Also:**
>
> "V$SYSTEM_PARAMETER"

9.111 V$PARAMETER_VALID_VALUES

V$PARAMETER_VALID_VALUES displays a list of valid values for list parameters.

Column	Datatype	Description
NUM	NUMBER	Parameter number
NAME	VARCHAR2(64)	Parameter name
ORDINAL	NUMBER	Ordinal number in the list (1-based)
VALUE	VARCHAR2(255)	Parameter value at ordinal
ISDEFAULT	VARCHAR2(64)	Indicates whether the given ordinal value is the default value for the parameter
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.112 V$PARAMETER2

V$PARAMETER2 displays information about the initialization parameters that are currently in effect for the session, with each list parameter value appearing as a row in the view. A new session inherits parameter values from the instance-wide values displayed in the V$SYSTEM_PARAMETER2 view.

Presenting the list parameter values in this format enables you to quickly determine the values for a list parameter. For example, if a parameter value is a, b, then the V$PARAMETER view does not tell you if the parameter has two values (both a and b) or one value (a, b). V$PARAMETER2 makes the distinction between the list parameter values clear.

Column	Datatype	Description
NUM	NUMBER	Parameter number
NAME	VARCHAR2(80)	Name of the parameter
TYPE	NUMBER	Parameter type: • 1 - Boolean • 2 - String • 3 - Integer • 4 - Parameter file • 5 - Reserved • 6 - Big integer
VALUE	VARCHAR2(4000)	Parameter value for the session (if modified within the session); otherwise, the instance-wide parameter value
DISPLAY_VALUE	VARCHAR2(4000)	Parameter value in a user-friendly format. For example, if the VALUE column shows the value 262144 for a big integer parameter, then the DISPLAY_VALUE column will show the value 256K.
ISDEFAULT	VARCHAR2(6)	Indicates whether the parameter is set to the default value (TRUE) or the parameter value was specified in the parameter file (FALSE)

Column	Datatype	Description
ISSES_MODIFIABLE	VARCHAR2(5)	Indicates whether the parameter can be changed with ALTER SESSION (TRUE) or not (FALSE)
ISSYS_MODIFIABLE	VARCHAR2(9)	Indicates whether the parameter can be changed with ALTER SYSTEM and when the change takes effect: • IMMEDIATE - Parameter can be changed with ALTER SYSTEM regardless of the type of parameter file used to start the instance. The change takes effect immediately. • DEFERRED - Parameter can be changed with ALTER SYSTEM regardless of the type of parameter file used to start the instance. The change takes effect in subsequent sessions. • FALSE - Parameter cannot be changed with ALTER SYSTEM unless a server parameter file was used to start the instance. The change takes effect in subsequent instances.
ISPDB_MODIFIABLE	VARCHAR2(5)	Indicates whether the parameter can be modified inside a PDB (TRUE) or not (FALSE). In a non-CDB, the value of this column is NULL.
ISINSTANCE_MODIFIABLE	VARCHAR2(5)	For parameters that can be changed with ALTER SYSTEM, indicates whether the value of the parameter can be different for every instance (TRUE) or whether the parameter must have the same value for all Real Application Clusters instances (FALSE). If the ISSYS_MODIFIABLE column is FALSE, then this column is always FALSE.
ISMODIFIED	VARCHAR2(10)	Indicates whether the parameter has been modified after instance startup: • MODIFIED - Parameter has been modified with ALTER SESSION • SYSTEM_MOD - Parameter has been modified with ALTER SYSTEM (which causes all the currently logged in sessions' values to be modified) • FALSE - Parameter has not been modified after instance startup
ISADJUSTED	VARCHAR2(5)	Indicates whether Oracle adjusted the input value to a more suitable value (for example, the parameter value should be prime, but the user input a non-prime number, so Oracle adjusted the value to the next prime number)
ISDEPRECATED	VARCHAR2(5)	Indicates whether the parameter has been deprecated (TRUE) or not (FALSE)
ISBASIC	VARCHAR2(5)	Indicates whether the parameter is a basic parameter (TRUE) or not (FALSE)
DESCRIPTION	VARCHAR2(255)	Description of the parameter
ORDINAL	NUMBER	Position (ordinal number) of the parameter value. Useful only for parameters whose values are lists of strings.
UPDATE_COMMENT	VARCHAR2(255)	Comments associated with the most recent update
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

> **See Also:**
>
> - "V$SYSTEM_PARAMETER"
> - "V$PARAMETER"

9.113 V$PASSWORDFILE_INFO

V$PASSWORDFILE_INFO provides information about the database password file.

This view can be queried from the root or from a pluggable database (PDB) in a multitenant container database (CDB). When queried, this view always returns one row.

Column	Datatype	Description
FILE_NAME	VARCHAR2(513)	Fully qualified password file name/location.
FORMAT	VARCHAR2(6)	Shows the format of the password file: • LEGACY • 12 • 12.2
IS_ASM	VARCHAR2(5)	Indicates whether the password file is stored in Oracle ASM. Possible values: • TRUE: The password file is stored in Oracle ASM. • FALSE: The password file is stored in Oracle Exascale or in the operating system file system.
IS_EXC	VARCHAR2(5)	Indicates whether the password file is stored in Oracle Exascale. Possible values: • TRUE: The password file is stored in Oracle Exascale. • FALSE: The password file is stored in Oracle ASM or in the operating system file system.
CON_ID	NUMBER	The ID of the container to which the data pertains. Because there is only one password file for a CDB that is common for the entire CDB, the only value possible for CON_ID is 0 for this view.

> **Note:**
>
> If the database password file name or location was recently changed, and you do not see the change reflected in this view, you can run the following SQL statement:
>
> ```
> SQL> ALTER SYSTEM FLUSH PASSWORDFILE_METADATA_CACHE;
> ```
>
> This statement flushes the metadata cache and updates the database to use the new password file. It also updates this view with the current password file information.

9.114 V$PATCHES

V$PATCHES shows the patches applied on an Oracle ASM instance and the list of patches applied to an Oracle Grid infrastructure home directory.

> **Note:**
>
> If this view is queried in an RDBMS instance, no patch information is returned.

Column	Datatype	Description
PATCH_ID	NUMBER	Patch identifier
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.115 V$PDB_INCARNATION

V$PDB_INCARNATION displays information about all PDB incarnations. Oracle creates a new PDB incarnation whenever a PDB is opened with the RESETLOGS option.

Column	Datatype	Description
DB_INCARNATION#	NUMBER	Database incarnation number
PDB_INCARNATION#	NUMBER	PDB incarnation number
STATUS	VARCHAR2(7)	Incarnation status: • ORPHAN: Orphan incarnation • CURRENT: Current incarnation of the PDB • PARENT: Parent of the current incarnation
INCARNATION_SCN	NUMBER	The SCN to flashback or recover to for this PDB incarnation
INCARNATION_TIME	DATE	The point in time recovered to for this PDB incarnation
BEGIN_RESETLOGS_SCN	NUMBER	The SCN at the beginning of PDB resetlogs
BEGIN_RESETLOGS_TIME	DATE	The time at the beginning of PDB resetlogs
END_RESETLOGS_SCN	NUMBER	The SCN at the end of PDB resetlogs
END_RESETLOGS_TIME	DATE	The time at the end of PDB resetlogs
PRIOR_DB_INCARNATION#	NUMBER	Parent database incarnation number
PRIOR_PDB_INCARNATION#	VARCHAR2(40)	Parent PDB incarnation number
FLASHBACK_DATABASE_ALLOWED	VARCHAR2(3)	Indicates whether the PDB can be flashbacked to this incarnation as part of a flashback database operation for the CDB

Column	Datatype	Description
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.116 V$PDBS

V$PDBS displays information about PDBs associated with the current instance.

Column	Datatype	Description
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data
DBID	NUMBER	PDB identifier calculated when the PDB is created and stored in all file headers associated with the PDB
CON_UID	NUMBER	Unique identifier associated with the PDB
GUID	RAW(16)	Globally unique identifier (GUID) of this PDB
NAME	VARCHAR2(128)	Name of the PDB
OPEN_MODE	VARCHAR2(10)	Open mode information. Possible values: • MOUNTED • READ WRITE • READ ONLY • MIGRATE
RESTRICTED	VARCHAR2(3)	Indicates whether only users possessing RESTRICTED SESSION privilege can connect to the PDB. Possible values: • YES • NO • NULL
OPEN_TIME	TIMESTAMP(3) WITH TIME ZONE	Date and time when the database was last opened
CREATE_SCN	NUMBER	System change number (SCN) for the creation of this PDB
TOTAL_SIZE	NUMBER	Shows the disk space (in bytes) used by the PDB, including both data and temp files.
BLOCK_SIZE	NUMBER	The current block size for the PDB

Column	Datatype	Description
RECOVERY_STATUS	VARCHAR2(26)	Shows the recovery status for the PDB. Possible values: • DISABLED: The PDB is offline. This is similar to all files belonging to this PDB being offline, except that all files to be added to this PDB in the future will also be offline to begin with. • DISABLED AUTOMATIC RECOVER: Same as DISABLED, except that standby Active Data Guard recovery will automatically attempt to spawn PDB recovery isolation for the PDB and transition the PDB into ENABLED. • ENABLED: The PDB is not offline at the PDB level. Individual files in the PDB may be online or offline at the file level.
SNAPSHOT_PARENT_CON_ID	NUMBER	This column shows the container ID of the PDB that this PDB is a snapshot clone of. This column shows a nonzero value only if the PDB is a snapshot clone. For all other cases, it shows a value of 0.
APPLICATION_ROOT	VARCHAR2(3)	Indicates whether the PDB is an application root
APPLICATION_PDB	VARCHAR2(3)	Indicates whether the PDB is an application PDB
APPLICATION_SEED	VARCHAR2(3)	Indicates whether the PDB is an application seed (an application seed is also an application PDB)
APPLICATION_ROOT_CON_ID	NUMBER	If this PDB is an application PDB, the container ID of an application root to which this application PDB belongs. If this PDB is an application root clone, the container ID of an application root to which this application root clone belongs. Otherwise, NULL.
APPLICATION_ROOT_CLONE	VARCHAR2(3)	Indicates whether this PDB is an application root clone (YES) or not (NO)
PROXY_PDB	VARCHAR2(3)	Indicates whether this PDB is a proxy PDB (YES) or not (NO)
LOCAL_UNDO	NUMBER	Shows whether the PDB is in local undo. Possible values: • 1 – PDB is in local undo mode • 0 – PDB is in shared undo mode This column is not relevant for CDB$ROOT.
UNDO_SCN	NUMBER	System change number (SCN) at which the PDB was last converted from shared to local undo, or from local to shared undo. This column is not relevant for CDB$ROOT.
UNDO_TIMESTAMP	DATE	Date and time at which the PDB was last converted from shared to local undo, or from local to shared undo. This column is not relevant for CDB$ROOT.
CREATION_TIME	DATE	Date and time at which the PDB was created.
DIAGNOSTICS_SIZE	NUMBER	Shows the current disk space usage (in bytes) of the diagnostic traces generated in the PDB, which is represented by the CON_ID column of the row
PDB_COUNT	NUMBER	The number of user-created PDBs belonging to a given application root or CDB$ROOT. For all other containers, its value is 0.
AUDIT_FILES_SIZE	NUMBER	Shows the current disk space usage (in bytes) by Unified Audit files (.bin format) in the current PDB.
MAX_SIZE	NUMBER	Shows the maximum amount of disk space (in bytes) that can be used by data and temp files in the PDB
MAX_DIAGNOSTICS_SIZE	NUMBER	Shows the maximum amount of disk space (in bytes) that can be used by diagnostic traces generated in the PDB
MAX_AUDIT_SIZE	NUMBER	Shows the maximum amount of disk space (in bytes) that can be used by Unified Audit files (.bin format) in the PDB

Column	Datatype	Description
LAST_CHANGED_BY	VARCHAR2(11)	Indicates what type of user last changed the PDB. Possible values: • COMMON USER • LOCAL USER
TEMPLATE	VARCHAR2(3)	For internal use only
TENANT_ID	VARCHAR2(256)	Pluggable database tenant key
UPGRADE_LEVEL	NUMBER	For internal use only
GUID_BASE64	VARCHAR2(30)	The GUID of the PDB, encoded in base64
RECOVERY_TARGET_PDB_INCARNATION#	NUMBER	The incarnation number to which all datafiles belonging to the PDB would be recovered by the RECOVER PLUGGABLE DATABASE command
CLOUD_IDENTITY	VARCHAR2(32767)	Cloud identifier for the PDB
CLOSE_TIME	TIMESTAMP(3) WITH TIME ZONE	Date and time when the database was last closed
PRIORITY	NUMBER	The priority of the PDB when performing an OPEN or SAVE STATE operation on multiple PDBs with a single ALTER PLUGGABLE DATABASE or ALTER DATABASE statement If a priority is not assigned to the PDB, then the value of this column is null. An operation is performed first on the PDB with the lowest PRIORITY number, then on the PDB with the second lowest PRIORITY number, and so on. The operation is then performed on PDBs with no assigned priority, if any, in no particular order.
BACKUP_SIZE	NUMBER	Reserved for future use
BACKUP_STATUS	VARCHAR2(8)	Reserved for future use

9.117 V$PERSISTENT_PUBLISHERS

V$PERSISTENT_PUBLISHERS displays information about all active publishers of the persistent queues in the database. There is one row per instance per queue per publisher. The rows are deleted when the database (or instance in an Oracle RAC environment) restarts.

Column	Datatype	Description
QUEUE_ID	NUMBER	Identifier for the queue
QUEUE_SCHEMA	VARCHAR2(128)	Owner of the queue
QUEUE_NAME	VARCHAR2(128)	Name of the queue
PUBLISHER_NAME	VARCHAR2(128)	Name of the agent enqueuing the message
PUBLISHER_ADDRESS	VARCHAR2(1024)	Address of the publisher agent
PROTOCOL	NUMBER	Protocol used by the publisher's address
ENQUEUED_MSGS	NUMBER	Number of messages that have been enqueued
ELAPSED_ENQUEUE_TIME	NUMBER	Total time spent doing enqueue (in hundredths of a second)
ENQUEUE_CPU_TIME	NUMBER	Total CPU time for enqueue (in hundredths of a second)
LAST_ENQUEUE_TIME	TIMESTAMP(6)	Last enqueue message timestamp
ENQUEUE_TRANSACTIONS	NUMBER	Number of enqueue transactions

Column	Datatype	Description
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.118 V$PERSISTENT_QMN_CACHE

V$PERSISTENT_QMN_CACHE displays detailed information and statistics about the background activities for all queue tables in the system. There is one row per queue table. The rows are deleted when the database (or instance in an Oracle RAC environment) restarts.

Column	Datatype	Description
QUEUE_TABLE_ID	NUMBER	Queue table object ID
TYPE	VARCHAR2(32)	Type of the queue table's queue monitor cache
STATUS	NUMBER	Status of the queue table's queue monitor cache
NEXT_SERVICE_TIME	TIMESTAMP(3) WITH TIME ZONE	Time when the queue table should be serviced by QMON servers
WINDOW_END_TIME	TIMESTAMP(3) WITH TIME ZONE	Time manager activity period for non-owner queue table operations
TOTAL_RUNS	NUMBER	Total number of times this queue table is served
TOTAL_LATENCY	NUMBER	Cumulative latency in serving the queue table (in hundredths of a second)
TOTAL_ELAPSED_TIME	NUMBER	Total time spent in processing this queue table (in seconds)
TOTAL_CPU_TIME	NUMBER	Cumulative CPU time for serving the queue table (in hundredths of a second)
TMGR_ROWS_PROCESSED	NUMBER	Number of time manager entries processed
TMGR_ELAPSED_TIME	NUMBER	Cumulative time for time management activities (in hundredths of a second)
TMGR_CPU_TIME	NUMBER	Cumulative CPU time for time management activities (in hundredths of a second)
LAST_TMGR_PROCESSING_TIME	TIMESTAMP(3) WITH TIME ZONE	Last timer manager processing time
DEQLOG_ROWS_PROCESSED	NUMBER	Number of dequeue log entries processed
DEQLOG_PROCESSING_ELAPSED_TIME	NUMBER	Total time for processing dequeue log entries (in hundredths of a second)
DEQLOG_PROCESSING_CPU_TIME	NUMBER	Total CPU time for processing dequeue log entries (in hundredths of a second)
LAST_DEQLOG_PROCESSING_TIME	TIMESTAMP(3) WITH TIME ZONE	Last dequeue log processing time
DEQUEUE_INDEX_BLOCKS_FREED	NUMBER	Number of dequeue index blocks freed
HISTORY_INDEX_BLOCKS_FREED	NUMBER	Number of history index blocks freed

Column	Datatype	Description
TIME_INDEX_BLOCKS_FREED	NUMBER	Number of time manager index blocks freed
INDEX_CLEANUP_COUNT	NUMBER	Number of times index block cleanup was attempted
INDEX_CLEANUP_ELAPSED_TIME	NUMBER	Total time for index block cleanup (in hundredths of a second)
INDEX_CLEANUP_CPU_TIME	NUMBER	Total CPU time for index block cleanup (in hundredths of a second)
LAST_INDEX_CLEANUP_TIME	TIMESTAMP(3) WITH TIME ZONE	Last index block cleanup time
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.119 V$PERSISTENT_QUEUES

V$PERSISTENT_QUEUES displays information about all active persistent queues in the database since the queues' first activity time. There is one row per queue. The rows are deleted when the database (or instance in an Oracle RAC environment) restarts.

Column	Datatype	Description
QUEUE_ID	NUMBER	Identifier for the queue
QUEUE_TABLE_ID	NUMBER	Queue table identifier
QUEUE_SCHEMA	VARCHAR2(128)	Owner of the queue
QUEUE_NAME	VARCHAR2(128)	Name of the queue
FIRST_ACTIVITY_TIME	TIMESTAMP(6)	First queue activity time since database startup
ENQUEUED_MSGS	NUMBER	Number of messages enqueued
DEQUEUED_MSGS	NUMBER	Number of messages dequeued **Note:** This column will not be incremented until all the subscribers of the message have dequeued the message and its retention time has elapsed.
BROWSED_MSGS	NUMBER	Number of messages that have been browsed
ELAPSED_ENQUEUE_TIME	NUMBER	Total time (in hundredths of a second) spent doing enqueue
ELAPSED_DEQUEUE_TIME	NUMBER	Total time (in hundredths of a second) spent doing dequeue
ENQUEUE_CPU_TIME	NUMBER	Total CPU time for enqueue (in hundredths of a second)
DEQUEUE_CPU_TIME	NUMBER	Total CPU time for dequeue (in hundredths of a second)
AVG_MSG_AGE	NUMBER	Average age of messages in the queue
DEQUEUED_MSG_LATENCY	NUMBER	Last dequeued message latency (in seconds)
ELAPSED_TRANSFORMATION_TIME	NUMBER	Total time (in hundredths of a second) spent doing transformation
ELAPSED_RULE_EVALUATION_TIME	NUMBER	Total time (in hundredths of a second) spent doing rule evaluation
ENQUEUED_EXPIRY_MSGS	NUMBER	Number of messages enqueued with expiry

Column	Datatype	Description
ENQUEUED_DELAY_MSGS	NUMBER	Number of messages enqueued with delay
MSGS_MADE_EXPIRED	NUMBER	Number of messages expired by time manager
MSGS_MADE_READY	NUMBER	Number of messages made ready by time manager
LAST_ENQUEUE_TIME	TIMESTAMP(6)	Last message enqueue time
LAST_DEQUEUE_TIME	TIMESTAMP(6)	Last message dequeue time
LAST_TM_EXPIRY_TIME	TIMESTAMP(6)	Last time message was expired by time manager
LAST_TM_READY_TIME	TIMESTAMP(6)	Last time message was made ready by time manager
ENQUEUE_TRANSACTIONS	NUMBER	Number of enqueue transactions
DEQUEUE_TRANSACTIONS	NUMBER	Number of dequeue transactions
EXECUTION_COUNT	NUMBER	Number of executions of the dequeue cursor
OLDEST_MSGID	RAW(16)	Message ID of the oldest message in the queue
OLDEST_MSG_ENQTM	TIMESTAMP(6)	Enqueue time of the oldest message in the queue
MANDATORY_AFF_SWITCHES_OUT	NUMBER	An affinity switch is a change in dequeue instance for a shard-subscriber pair. A mandatory affinity switch is when there are local enqueues in the queue at the instance but no local dequeues present, so the dequeue affinity is switched to another instance for that shard-subscriber pair. This column shows the number of times mandatory affinity switches were needed from this instance to another for this queue.
OPTIONAL_AFF_SWITCHES_OUT	NUMBER	Optional affinity switches are affinity switches that are not mandatory. Optional affinity switches are done for global load balancing across the Oracle Real Application Clusters (Oracle RAC) database. This column shows the number of times optional affinity switches were needed from this instance to another for this queue.
AFF_SWITCHES_BACK_IN	NUMBER	The number of times dequeue affinities have come back from other instances to this instance. (MANDATORY_AFF_SWITCHES_OUT + OPTIONAL_AFF_SWITCHES_OUT - AFF_SWITCHES_BACK_IN) is the number of cross instance affinities present across all shard-subscriber pair for shards owned by this instance for this queue.
CROSS_STREAM_JOBS	NUMBER	The number of times a shard is being forwarded to another instance due to cross instance dequeues for this queue
RESTORE_BITMAP_JOBS	NUMBER	The number of times subscribers used existing shard forwarding to have cross instance dequeues for this queue
SHADOW_AFF_SWITCHES_IN	NUMBER	The number of affinity switches for this queue where this instance is the dequeue instance for a shard-subscriber pair where the shard is being enqueued at another instance
SHADOW_AFF_SWITCHES_OUT	NUMBER	The number of affinity switches for this queue where shadow affinity is switched back to source instance of the shard. (SHADOW_AFF_SWITCHES_IN - SHADOW_AFF_SWITCHES_OUT) is the number of dequeue affinities which are performing cross instance dequeues from non-local shards.
SHADOW_SHARDS_RECEIVED	NUMBER	The number of times a shard is being forwarded from another instance to this instance due to cross instance dequeues for this queue
SHADOW_SHARDS_FREED	NUMBER	The number of times a forwarded shard to this instance was stopped due to removal of cross instance dequeues for this queue
SHADOW_EVENT_STREAMS_RECEIVED	NUMBER	The number of times an event stream is being forwarded from another instance to this instance due to cross instance dequeues for this queue

Column	Datatype	Description
SHADOW_EVENT_STREAMS_FREED	NUMBER	The number of times a forwarded event stream to this instance was stopped due to removal of cross instance dequeues for this queue
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

> **Note:**
>
> For sharded queues, only the following columns in this view contain accurate information: FIRST_ACTIVITY_TIME, BROWSED_MSGS, LAST_ENQUEUE_TIME, LAST_DEQUEUE_TIME, ENQUEUE_TRANSACTIONS, and DEQUEUE_TRANSACTIONS. The rest of the columns in this view should be ignored when querying information about sharded queues.

9.120 V$PERSISTENT_SUBSCRIBERS

V$PERSISTENT_SUBSCRIBERS displays information about all active subscribers of the persistent queues in the database.

There is one row per instance per queue per subscriber. The rows are deleted when the database (or instance in an Oracle RAC environment) restarts.

> **Note:**
>
> This view does not display information about sharded queues. For information about sharded queues, refer to the "V$AQ_SHARDED_SUBSCRIBER_STAT" view.

Column	Datatype	Description
QUEUE_ID	NUMBER	Identifier for the queue
QUEUE_SCHEMA	VARCHAR2(128)	Owner of the queue
QUEUE_NAME	VARCHAR2(128)	Name of the queue
SUBSCRIBER_ID	NUMBER	Internal subscriber number
SUBSCRIBER_NAME	VARCHAR2(512)	Name of the subscriber
SUBSCRIBER_ADDRESS	VARCHAR2(1024)	Address of the subscribing agent
PROTOCOL	NUMBER	Protocol of the subscribing agent
SUBSCRIBER_TYPE	VARCHAR2(128)	Type of the subscriber: • PROXY - Propagation subscriber • SUBSCRIBER - Normal subscriber • RECIPIENT - Recipient
FIRST_ACTIVITY_TIME	TIMESTAMP(6)	First subscriber activity time since database startup

Column	Datatype	Description
ENQUEUED_MSGS	NUMBER	Number of messages enqueued since FIRST_ACTIVITY_TIME
DEQUEUED_MSGS	NUMBER	Number of messages dequeued since FIRST_ACTIVITY_TIME
AVG_MSG_AGE	NUMBER	Average age of messages in the queue
BROWSED_MSGS	NUMBER	Number of messages that have been browsed
EXPIRED_MSGS	NUMBER	Number of messages expired since FIRST_ACTIVITY_TIME
DEQUEUED_MSG_LATENCY	NUMBER	Last dequeued message latency (in seconds)
LAST_ENQUEUE_TIME	TIMESTAMP(6)	Timestamp of the last enqueued message
LAST_DEQUEUE_TIME	TIMESTAMP(6)	Timestamp of the last dequeued message
ELAPSED_DEQUEUE_TIME	NUMBER	Total time spent in dequeue (in hundredths of a second)
DEQUEUE_CPU_TIME	NUMBER	Total CPU time for dequeue (in hundredths of a second)
DEQUEUE_TRANSACTIONS	NUMBER	Number of dequeue transactions
EXECUTION_COUNT	NUMBER	Number of executions of the dequeue index cursor
DEQUEUE_MEMORY_LOCKS	NUMBER	Number of dequeue transactions that obtained memory locks
DEQUEUE_DISK_LOCKS	NUMBER	Number of dequeue transactions that obtained disk locks
DEQUEUE_DISK_DELETES	NUMBER	Number of dequeue transactions that deleted index-organized table entries
OLDEST_MSGID	RAW(16)	Message ID of the oldest message
OLDEST_MSG_ENQTM	TIMESTAMP(6)	Enqueue time of the oldest message
PARENT_SUBSCRIBER_ID	NUMBER	Subscriber ID of the parent durable subscriber
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.121 V$PGA_TARGET_ADVICE

V$PGA_TARGET_ADVICE predicts how the cache hit percentage and over allocation count statistics displayed by the V$PGASTAT performance view would be impacted if the value of the PGA_AGGREGATE_TARGET parameter is changed.

The prediction is performed for various values of the PGA_AGGREGATE_TARGET parameter, selected around its current value. The advice statistic is generated by simulating the past workload run by the instance.

The content of the view is empty if PGA_AGGREGATE_TARGET is not set. In addition, the content of this view is not updated if the STATISTICS_LEVEL parameter is set to BASIC. Base statistics for this view are reset at instance startup and when the value of the PGA_AGGREGATE_TARGET initialization parameter is dynamically modified.

Column	Datatype	Description
PGA_TARGET_FOR_ESTIMATE	NUMBER	Value of PGA_AGGREGATE_TARGET for this prediction (in bytes)

Column	Datatype	Description
PGA_TARGET_FACTOR	NUMBER	PGA_TARGET_FOR_ESTIMATE / the current value of the PGA_AGGREGATE_TARGET parameter
ADVICE_STATUS	VARCHAR2(3)	Indicates whether the advice is enabled (ON) or disabled (OFF) depending on the value of the STATISTICS_LEVEL parameter
BYTES_PROCESSED	NUMBER	Total bytes processed by all the work areas considered by this advice (in bytes)
ESTD_TIME	NUMBER	Time (in seconds) required to process the bytes
ESTD_EXTRA_BYTES_RW	NUMBER	Estimated number of extra bytes which would be read or written if PGA_AGGREGATE_TARGET was set to the value of the PGA_TARGET_FOR_ESTIMATE column. This number is derived from the estimated number and size of work areas which would run in one-pass (or multi-pass) for that value of PGA_AGGREGATE_TARGET.
ESTD_PGA_CACHE_HIT_PERCENTAGE	NUMBER	Estimated value of the cache hit percentage statistic when PGA_AGGREGATE_TARGET equals PGA_TARGET_FOR_ESTIMATE. This column is derived from the above two columns and is equal to BYTES_PROCESSED / (BYTES_PROCESSED + ESTD_EXTRA_BYTES_RW)
ESTD_OVERALLOC_COUNT	NUMBER	Estimated number of PGA memory over-allocations if the value of PGA_AGGREGATE_TARGET is set to PGA_TARGET_FOR_ESTIMATE. A nonzero value means that PGA_TARGET_FOR_ESTIMATE is not large enough to run the work area workload. Hence, the DBA should not set PGA_AGGREGATE_TARGET to PGA_TARGET_FOR_ESTIMATE since Oracle will not be able to honor that target.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

See Also:

- "V$PGASTAT"
- "PGA_AGGREGATE_TARGET"
- "STATISTICS_LEVEL"
- *Oracle Database Performance Tuning Guide* for information on tuning the PGA_AGGREGATE_TARGET initialization parameter using the PGA advice views

9.122 V$PGA_TARGET_ADVICE_HISTOGRAM

V$PGA_TARGET_ADVICE_HISTOGRAM predicts how statistics displayed by the V$SQL_WORKAREA_HISTOGRAM dynamic view would be impacted if the value of the PGA_AGGREGATE_TARGET parameter is changed.

This prediction is performed for various values of the PGA_AGGREGATE_TARGET parameter, selected around its current value. The advice statistic is generated by simulating the past workload run by the instance.

The content of the view is empty if PGA_AGGREGATE_TARGET is not set. In addition, the content of this view is not updated when the STATISTICS_LEVEL initialization parameter is set to BASIC. Base statistics for this view are reset at instance startup or when the value of the PGA_AGGREGATE_TARGET initialization parameter is dynamically modified.

Column	Datatype	Description
PGA_TARGET_FOR_ESTIMATE	NUMBER	Value of PGA_AGGREGATE_TARGET for this prediction (in bytes)
PGA_TARGET_FACTOR	NUMBER	PGA_TARGET_FOR_ESTIMATE / the current value of the PGA_AGGREGATE_TARGET parameter
ADVICE_STATUS	VARCHAR2(3)	Indicates whether the advice is enabled (ON) or disabled (OFF) depending on the value of the STATISTICS_LEVEL parameter
LOW_OPTIMAL_SIZE	NUMBER	Lower bound for the optimal memory requirement of work areas included in this row (in bytes)
HIGH_OPTIMAL_SIZE	NUMBER	Upper bound for the optimal memory requirement of work areas included in this row (in bytes)
ESTD_OPTIMAL_EXECUTIONS	NUMBER	Number of work areas with an optimal memory requirement comprised between LOW_OPTIMAL_SIZE and HIGH_OPTIMAL_SIZE which are predicted to run optimal given a value of PGA_AGGREGATE_TARGET equal to PGA_TARGET_FOR_ESTIMATE
ESTD_ONEPASS_EXECUTIONS	NUMBER	Number of work areas with an optimal memory requirement comprised between LOW_OPTIMAL_SIZE and HIGH_OPTIMAL_SIZE which are predicted to run one-pass given a value of PGA_AGGREGATE_TARGET equal to PGA_TARGET_FOR_ESTIMATE
ESTD_MULTIPASSES_EXECUTIONS	NUMBER	Number of work areas with an optimal memory requirement comprised between LOW_OPTIMAL_SIZE and HIGH_OPTIMAL_SIZE which are predicted to run multi-pass given a value of PGA_AGGREGATE_TARGET equal to PGA_TARGET_FOR_ESTIMATE
ESTD_TOTAL_EXECUTIONS	NUMBER	Sum of ESTD_OPTIMAL_EXECUTIONS, ESTD_ONEPASS_EXECUTIONS, and ESTD_MULTIPASSES_EXECUTIONS
IGNORED_WORKAREAS_COUNT	NUMBER	Number of work areas with optimal memory requirement between LOW_OPTIMAL_SIZE and HIGH_OPTIMAL_SIZE ignored in the advice generation due to memory and CPU constraints
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

> **See Also:**
>
> - "V$SQL_WORKAREA_HISTOGRAM"
> - "PGA_AGGREGATE_TARGET"
> - "STATISTICS_LEVEL"
> - *Oracle Database Performance Tuning Guide* for information on tuning the PGA_AGGREGATE_TARGET initialization parameter using the PGA advice views

9.123 V$PGASTAT

V$PGASTAT displays PGA memory usage statistics as well as statistics about the automatic PGA memory manager when it is enabled (that is, when PGA_AGGREGATE_TARGET is set). Cumulative values in V$PGASTAT are accumulated since instance startup.

Column	Datatype	Description
NAME	VARCHAR2(64)	Name of the statistic (see Table 9-4)
VALUE	NUMBER	Statistic value
UNIT	VARCHAR2(12)	Unit for the value: • bytes • microseconds • percent
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

Table 9-4 V$PGASTAT Statistics

Statistic Name	Description
aggregate PGA auto target	Amount of PGA memory the Oracle Database can use for work areas running in automatic mode. This amount is dynamically derived from the value of the PGA_AGGREGATE_TARGET initialization parameter and the current work area workload, and continuously adjusted by the Oracle Database. If this value is small compared to the value of PGA_AGGREGATE_TARGET, then a large amount of PGA memory is used by other components of the system (for example, PL/SQL or Java memory) and little is left for work areas. The DBA must ensure that enough PGA memory is left for work areas running in automatic mode.
aggregate PGA target parameter	Current value of the PGA_AGGREGATE_TARGET initialization parameter. If this parameter is not set, then its value is 0 and automatic management of PGA memory is disabled.
bytes processed	Number of bytes processed by memory intensive SQL operators, cumulated since instance startup.

Table 9-4 (Cont.) V$PGASTAT Statistics

Statistic Name	Description
cache hit percentage	A metric computed by the Oracle Database to reflect the performance of the PGA memory component, cumulative since instance startup. A value of 100% means that all work areas executed by the system since instance startup have used an optimal amount of PGA memory. When a work area cannot run optimal, one or more extra passes is performed over the input data. This will reduce the cache hit percentage in proportion to the size of the input data and the number of extra passes performed.
extra bytes read/written	Number of bytes processed during extra passes of the input data, cumulated since instance startup. When a work area cannot run optimal, one or more of these extra passes is performed.
global memory bound	Maximum size of a work area executed in automatic mode. This value is continuously adjusted by the Oracle Database to reflect the current state of the work area workload. The global memory bound generally decreases when the number of active work areas is increasing in the system. If the value of the global bound decreases below 1 MB, then the value of `PGA_AGGREGATE_TARGET` should be increased.
max processes count	Maximum number of processes active at any one time since instance startup.
maximum PGA allocated	Maximum number of bytes of PGA memory allocated at one time since instance startup.
maximum PGA used for auto workareas	Maximum amount of PGA memory consumed at one time by work areas running under the automatic memory management mode since instance startup.
maximum PGA used for manual workareas	Maximum amount of PGA memory consumed at one time by work areas running under the manual memory management mode since instance startup.
over allocation count	This statistic is cumulative since instance startup. Over allocating PGA memory can happen if the value of `PGA_AGGREGATE_TARGET` is too small. When this happens, the Oracle Database cannot honor the value of `PGA_AGGREGATE_TARGET` and extra PGA memory needs to be allocated. If over allocation occurs, then increase the value of `PGA_AGGREGATE_TARGET` using the information provided by the `V$PGA_TARGET_ADVICE` view.
PGA memory freed back to OS	Number of bytes of PGA memory freed back to the operating system, cumulated since instance startup.
process count	Number of processes active within up to the last 3 seconds.
recompute count (total)	Number of times the instance bound, which is a cap on the maximum size of each active work area, has been recomputed since instance startup. Generally, the instance bound is recomputed in the background every 3 seconds, but it could be recomputed by a foreground process when the number of work areas changes rapidly in a short period of time.
total freeable PGA memory	Number of bytes of PGA memory in all processes that could be freed back to the operating system.
total PGA allocated	Current amount of PGA memory allocated by the instance. The Oracle Database attempts to keep this number below the value of the `PGA_AGGREGATE_TARGET` initialization parameter. However, it is possible for the PGA allocated to exceed that value by a small percentage and for a short period of time when the work area workload is increasing very rapidly or when `PGA_AGGREGATE_TARGET` is set to a small value.
total PGA inuse	Indicates how much PGA memory is currently consumed by work areas. This number can be used to determine how much memory is consumed by other consumers of the PGA memory (for example, PL/SQL or Java).

Table 9-4 (Cont.) V$PGASTAT Statistics

Statistic Name	Description
total PGA used for auto workareas	Indicates how much PGA memory is currently consumed by work areas running under the automatic memory management mode. This number can be used to determine how much memory is consumed by other consumers of the PGA memory (for example, PL/SQL or Java).
total PGA used for manual workareas	Indicates how much PGA memory is currently consumed by work areas running under the manual memory management mode. This number can be used to determine how much memory is consumed by other consumers of the PGA memory (for example, PL/SQL or Java).

> **See Also:**
>
> "PGA_AGGREGATE_TARGET"

9.124 V$PKCS11_PATH

`V$PKCS11_PATH` displays all processes that are using a PKCS#11 library for Transparent Data Encryption (TDE).

The SQL command `ADMINISTER KEY MANAGEMENT SWITCHOVER TO LIBRARY` instructs the database to switch over from the current PKCS#11 library to a new PKCS#11 library, without incurring system downtime. After you issue this command, foreground and background processes are gradually switched over to the new library. You can use this view to monitor the PKCS#11 library being used by each process. When you determine that all processes have switched over to the new library, you can safely remove the old library from disk.

Column	Datatype	Description
`PID`	`NUMBER`	ID of the process
`PNAME`	`VARCHAR2(5)`	Name of the process
`PKCS11_LIB_PATH`	`VARCHAR2(4000)`	Full operating system path for the PKCS#11 library being used by the process
`CON_ID`	`NUMBER`	The ID of the container to which the data pertains. Possible values include: • `0`: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • `1`: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.125 V$PLSQL_DEBUGGABLE_SESSIONS

`V$PLSQL_DEBUGGABLE_SESSIONS` shows the current sessions of all users that the current user has privileges to debug with a PL/SQL debugger.

Column	Datatype	Description
SID	NUMBER	Session identifier
SERIAL#	NUMBER	Session serial number
LOGON_TIME	DATE	Time of logon
USER#	NUMBER	Oracle user identifier
USERNAME	VARCHAR2(128)	Oracle user name
OSUSER	VARCHAR2(128)	Operating system client user name
PROCESS	VARCHAR2(24)	Operating system client process ID
MACHINE	VARCHAR2(64)	Operating system client machine name
PORT	NUMBER	Operating system client port number
TERMINAL	VARCHAR2(30)	Operating system client terminal name
PROGRAM	VARCHAR2(84)	Operating system client program name
TYPE	VARCHAR2(10)	Session type
SERVICE_NAME	VARCHAR2(64)	Service name of the session
PLSQL_DEBUGGER_CONNECTED	VARCHAR2(5)	Indicates whether the session is connected to a PL/SQL debugger. Possible values: • TRUE • FALSE
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.126 V$PMEM_FILESTORE

V$PMEM_FILESTORE displays information about Persistent Memory Filestores (PMEM Filestores).

Column	Datatype	Description
NAME	VARCHAR2(65)	PMEM Filestore name
MOUNT_POINT	VARCHAR2(513)	Operating system directory path for the mount point of the PMEM Filestore
BACKING_FILE	VARCHAR2(513)	Operating system file path for the backing file of the PMEM Filestore
BLOCK_SIZE	NUMBER	Block size of the PMEM Filestore (in bytes)
TOTAL_BYTES	NUMBER	Current size of the PMEM Filestore (in bytes)
AUTOEXTENSIBLE	VARCHAR2(4)	Indicates whether the PMEM Filestore is autoextensible (YES) or not (NO)
INCREMENT_BY	NUMBER	If the PMEM Filestore is autoextensible, then this column displays the size of an autoextension increment (in bytes) If the PMEM Filestore is not autoextensible, then the value of this column is 0

Column	Datatype	Description
MAX_BYTES	NUMBER	Maximum autoextend size (in bytes)
AVAILABLE_BYTES	NUMBER	Amount of free space currently available in the PMEM Filestore (in bytes)
USED_BYTES	NUMBER	Amount of space currently allocated to data and metadata in the PMEM Filestore (in bytes)
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.127 V$PQ_SESSTAT

V$PQ_SESSTAT lists session statistics for parallel queries.

After you have run a query or DML operation, you can use the information derived from V$PQ_SESSTAT to view the number of PX (Parallel Execution) servers used, and other information for the session and system.

Column	Datatype	Description
STATISTIC	VARCHAR2(30)	Name of the statistic: • Allocation Height - Requested number of PX servers per instance • Allocation Width - Requested number of instances • DDL Parallelized - Number of DDL operations run in parallel • DFO Trees - Number of executed DFO trees • DML Parallelized - Number of DML operations run in parallel • DOP - Degree of parallelism used for the last statement • Distr Msgs Recv'd - Number of remote (inter-instance) messages received • Distr Msgs Sent - Number of remote (inter-instance) messages sent • Local Msgs Recv'd - Number of local (intra-instance) messages received • Local Msgs Sent - Number of local (intra-instance) messages sent • Queries Parallelized - Number of queries run in parallel • Server Sets - Maximum number of PX server sets used at the same time for the last statement • Server Sets Released Early - Total number of PX server sets released proactively • Servers - Maximum number of PX servers used at the same time • Servers Released Early - Total number of PX servers released proactively • Total Servers Released - Total number of PX servers acquired and released
LAST_QUERY	NUMBER	Value of the statistic for the last operation

Column	Datatype	Description
SESSION_TOTAL	NUMBER	Value of the statistic for the entire session to this point in time The value of the statistic for the entire session to this point in time is not maintained for the DOP, Servers, Allocation Height, Allocation Width, Server Sets, Total Servers Released, Servers Released Early, and Server Sets Released Early statistics.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.128 V$PQ_SLAVE

V$PQ_SLAVE lists statistics for each of the active PX (Parallel Execution) servers on an instance.

> **Note:**
>
> This view is deprecated. Oracle recommends that you instead use the V$PX_SERVER view.

> **See Also:**
>
> "V$PX_SERVER"

9.129 V$PQ_SYSSTAT

V$PQ_SYSSTAT lists system statistics for parallel queries.

After you have run a query or DML operation, you can use the information derived from V$PQ_SYSSTAT to view the number of PX (Parallel Execution) servers used, and other information for the system.

Column	Datatype	Description
STATISTIC	VARCHAR2(30)	Name of the statistic: • DDL Initiated - Total number of parallel DDL operations that were initiated • DDL Initiated (IPQ) - Total number of parallel DDL operations that were initiated for Internode Parallel Query • DFO Trees - Total number of DFO trees executed on this instance • DML Initiated - Total number of parallel DML operations that were initiated • DML Initiated (IPQ) - Total number of parallel DML operations that were initiated for Internode Parallel Query • Distr Msgs Recv'd - Total number of remote (inter-instance) messages received on this instance • Distr Msgs Sent - Total number of remote (inter-instance) messages sent on this instance • Local Msgs Recv'd - Total number of remote (inter-instance) messages received on this instance • Local Msgs Sent - Total number of local (intra-instance) messages sent on this instance • Queries Initiated - Total number of parallel queries initiated on this instance • Queries Initiated (IPQ) - Total number of parallel queries initiated on this instance for Internode Parallel Query • Queries Queued - Total number of queued queries on this instance • Server Sessions - Total number of operations executed in all PX servers on this instance • Servers Busy - Number of currently busy PX servers on this instance • Servers Cleaned Up - Total number of PX servers on this instance cleaned up due to process death • Servers Highwater - Number of active PX servers on this instance that have partaken in >= 1 operation so far • Servers Idle - Number of currently idle PX servers on this instance • Servers Shutdown - Total number of PX servers shutdown on this instance • Servers Started - Total number of PX servers started on this instance • Sessions Active - Total number of active sessions on this instance
VALUE	NUMBER	Value of the statistic
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.130 V$PQ_TQSTAT

`V$PQ_TQSTAT` contains statistics on parallel execution operations.

The statistics are compiled after the query completes and only remain for the duration of the session. It displays the number of rows processed through each PX (Parallel Execution) server at each stage of the execution tree. This view can help determine skew problems in a query's execution. (Note that for PDML, information from `V$PQ_TQSTAT` is available only after a commit or rollback operation.)

Column	Datatype	Description
DFO_NUMBER	NUMBER	Data flow operator (DFO) tree number to differentiate queries
TQ_ID	NUMBER	Table queue ID within the query, which represents the connection between two DFO nodes in the query execution tree
SERVER_TYPE	VARCHAR2(10)	The role in table queue - producer/consumer/ranger
NUM_ROWS	NUMBER	The number of rows produced/consumed
BYTES	NUMBER	The number of bytes produced/consumed
OPEN_TIME	NUMBER	Time (seconds) the table queue remained open
AVG_LATENCY	NUMBER	Time (minutes) for a message to be dequeued after it enters the queue
WAITS	NUMBER	The number of waits encountered during dequeue
TIMEOUTS	NUMBER	The number of timeouts when waiting for a message
PROCESS	VARCHAR2(6)	Process ID
INSTANCE	NUMBER	Instance ID
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.131 V$PROCESS

`V$PROCESS` displays information about the currently active processes.

Column	Datatype	Description
ADDR	RAW(4 | 8)	Address of the process state object
PID	NUMBER	Oracle process identifier
SOSID	VARCHAR2(24)	Operating system (process, thread) identifier. This identifier is unique whether the Oracle multiprocess/multithread feature is enabled or not.

Column	Datatype	Description
SPID	VARCHAR2(24)	Operating system process identifier. The Oracle multiprocess/multithread feature is available for UNIX systems. When the Oracle multiprocess/multithread feature is enabled, RDBMS processes are mapped to threads running in operating system processes, and the SPID identifier is not unique for RDBMS processes. When the Oracle multiprocess/multithread feature is not enabled on UNIX systems, the SPID identifier is unique for RDBMS processes.
STID	VARCHAR2(24)	Operating system thread identifier. The Oracle multiprocess/multithread feature is available for UNIX systems. When the Oracle multiprocess/multithread feature is enabled, RDBMS processes are mapped to threads running in operating system processes, and the SPID and STID together uniquely identify an RDBMS process. The STID is not unique on Solaris. The STID is unique on Linux, AIX, and Microsoft Windows.
EXECUTION_TYPE	VARCHAR2(10)	Operating system execution type
PNAME	VARCHAR2(5)	Name of this process
USERNAME	VARCHAR2(15)	Operating system process username
SERIAL#	NUMBER	Process serial number
TERMINAL	VARCHAR2(30)	Operating system terminal identifier
PROGRAM	VARCHAR2(84)	Program in progress
TRACEID	VARCHAR2(255)	Trace file identifier
TRACEFILE	VARCHAR2(513)	Trace file name of the process
BACKGROUND	VARCHAR2(1)	1 for a SYSTEM background process; NULL for foreground processes or non-SYSTEM background processes
ATTRIBUTES	VARCHAR2(512)	Attributes of the process, in JSON format
LATCHWAIT	VARCHAR2(16)	Address of the latch the process is waiting for; NULL if none
LATCHSPIN	VARCHAR2(16)	This column is obsolete
PGA_USED_MEM	NUMBER	PGA memory currently used by the process (in bytes)
PGA_ALLOC_MEM	NUMBER	PGA memory currently allocated by the process (including free PGA memory not yet released to the operating system by the server process), in bytes
PGA_FREEABLE_MEM	NUMBER	Allocated PGA memory which can be freed (in bytes)
PGA_MAX_MEM	NUMBER	Maximum PGA memory ever allocated by the process (in bytes)
NUMA_DEFAULT	NUMBER	The NUMA processor group of this process at initialization time
NUMA_CURR	NUMBER	The NUMA processor group of this process currently
CPU_USED	NUMBER	Tracks the CPU used by this process (in microseconds) from the time it was spawned

Column	Datatype	Description
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.132 V$PROCESS_MEMORY

V$PROCESS_MEMORY displays dynamic PGA memory usage by named component categories for each process.

Column	Datatype	Description
PID	NUMBER	Oracle process identifier
SERIAL#	NUMBER	Oracle process serial number
CATEGORY	VARCHAR2(15)	Category name. Categories include "SQL", "PL/SQL", "OLAP" and "JAVA". Special categories are "Freeable" and "Other". Freeable memory has been allocated to the process by the operating system, but has not been allocated to a category. "Other" memory has been allocated to a category, but not to one of the named categories.
ALLOCATED	NUMBER	Bytes of PGA memory allocated by the process for the category. For the "Freeable" category, it is the amount of free PGA memory eligible to be released to the operating system.
USED	NUMBER	Bytes of PGA memory used by the process for the category. For "Freeable", the value is zero. For "Other", the value is NULL for performance reasons.
MAX_ALLOCATED	NUMBER	Maximum bytes of PGA memory ever allocated by the process for the category.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.133 V$PROCESS_MEMORY_DETAIL

V$PROCESS_MEMORY_DETAIL provides detailed information on dynamic PGA memory usage for each automatically captured snapshot.

Column	Datatype	Description
PID	NUMBER	Oracle process identifier
SERIAL#	NUMBER	Oracle process serial number
TIMESTAMP	DATE	Time at which the snapshot was taken

Column	Datatype	Description
SQL_ID	VARCHAR2(13)	Identifier for the relevant SQL statement
TOTAL_MB	NUMBER	Total number of megabytes of PGA memory allocated at the time of the snapshot
CATEGORY	VARCHAR2(15)	Category name. Categories include: • SQL • PL/SQL • OLAP • JAVA • Freeable • Other
NAME	VARCHAR2(26)	PGA memory allocation comment. Small allocations may be grouped together with NAME set to Miscellaneous for performance reasons.
HEAP_NAME	VARCHAR2(15)	Name of heap or heaps (if same name) containing the allocations
BYTES	NUMBER	Bytes of PGA memory allocated in the process from heaps with the given heap name and with the given allocation comment
ALLOCATION_COUNT	NUMBER	Number of allocations with the comment found in the process inside heaps with the given heap name
HEAP_DESCRIPTOR	RAW(4 | 8)	If all the allocations are from one heap, then this is the address of the heap descriptor for that heap. Otherwise, this column is NULL.
PARENT_HEAP_DESCRIPTOR	RAW(4 | 8)	If all the allocations are from one heap, then this is the address of the parent heap descriptor for that heap. Otherwise, this column is NULL. If the heap has no parent, the value is zero.
MOST_RECENT	NUMBER	Oracle process identifier (PID) for most recent snapshot
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.134 V$PROCESS_POOL

V$PROCESS_POOL provides information about process pools.

Column	Datatype	Description
POOL_NAME	VARCHAR2(512)	Pool name
ENABLED	VARCHAR2(5)	Indicates whether the pool is active (TRUE) or not (FALSE)
MIN_COUNT	NUMBER	The default or configured minimum value
BATCH_COUNT	NUMBER	The default or configured batch count
INIT_COUNT	NUMBER	The default or configured initial count
CUR_COUNT	NUMBER	The number of spawned processes available in this pool

Column	Datatype	Description
MAX_COUNT	NUMBER	When INIT_COUNT is set, MAX_COUNT shows the number of processes yet to be spawned. Note that the current process count (CUR_COUNT) cannot reach INIT_COUNT in cases where processes are consumed during the spawn.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This row is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.135 V$PROPAGATION_RECEIVER

V$PROPAGATION_RECEIVER displays information about buffer queue propagation schedules on the receiving (destination) side. The values are reset to zero when the database (or instance in an Oracle Real Application Clusters (Oracle RAC) environment) restarts, when propagation migrates to another instance, or when an unscheduled propagation is attempted.

Column	Datatype	Description
SRC_QUEUE_SCHEMA	VARCHAR2(128)	Name of the source schema
SRC_QUEUE_NAME	VARCHAR2(128)	Name of the source queue
SRC_DBNAME	VARCHAR2(395)	Name of the source database
DST_QUEUE_SCHEMA	VARCHAR2(128)	Name of the destination schema
DST_QUEUE_NAME	VARCHAR2(128)	Name of the destination queue
STARTUP_TIME	DATE	Startup time of this schedule. This time changes when the source/ destination database gets restarted.
HIGH_WATER_MARK	NUMBER	High watermark of the messages received
ACKNOWLEDGEMENT	NUMBER	Acknowledgement of the messages received by the receiver
LAST_RECEIVED_MSG	NUMBER	Last received message
TOTAL_MSGS	NUMBER	Total number of messages
ELAPSED_UNPICKLE_TIME	NUMBER	Elapsed unpickle time
ELAPSED_RULE_TIME	NUMBER	Elapsed rule time
ELAPSED_ENQUEUE_TIME	NUMBER	Elapsed enqueue time
SESSION_ID	NUMBER	Session ID of the propagation receiver
SERIAL#	NUMBER	Serial number of the propagation receiver
SPID	VARCHAR2(24)	Process identification number of the propagation receiver
PROPAGATION_NAME	VARCHAR2(128)	Name of the propagation on the source database

Column	Datatype	Description
STATE	VARCHAR2(43)	State of the propagation receiver: • Initializing • Sending unapplied txns • Waiting for message from client • Receiving LCRs • Evaluating rules • Enqueueing LCRS • Waiting for memory • Waiting for apply to read • Waiting for message from propagation sender When the propagation schedule is not optimized, the state is Normal.
LAST_RECEIVED_MSG_POSITION	RAW(64)	Last received message position. Corresponds to LAST_RECEIVED_MSG, except the value is in position rather than SCN. Position is used by XStream to determine ordering.
ACKNOWLEDGEMENT_POSITION	RAW(64)	Acknowledgement position of the messages received by the receiver. Corresponds to ACKNOWLEDGEMENT, except the value is in position rather than SCN. Position is used by XStream to determine ordering.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.136 V$PROPAGATION_SENDER

V$PROPAGATION_SENDER displays information about buffer queue propagation schedules on the sending (source) side. The values are reset to zero when the database (or instance in an Oracle Real Application Clusters (Oracle RAC) environment) restarts, when propagation migrates to another instance, or when an unscheduled propagation is attempted.

Column	Datatype	Description
QUEUE_ID	NUMBER	Queue identifier of the queue
QUEUE_SCHEMA	VARCHAR2(128)	Schema of the queue
QUEUE_NAME	VARCHAR2(128)	Name of the queue
DST_QUEUE_SCHEMA	VARCHAR2(128)	Destination schema of the queue
DST_QUEUE_NAME	VARCHAR2(128)	Name of the destination queue
STARTUP_TIME	DATE	Time at which the propagation started
DBLINK	VARCHAR2(395)	Name of the destination database link
HIGH_WATER_MARK	NUMBER	High watermark of the messages sent
ACKNOWLEDGEMENT	NUMBER	Acknowledgement of the messages received by the receiver
SCHEDULE_STATUS	VARCHAR2(128)	Status of the propagation schedule
TOTAL_MSGS	NUMBER	Total messages propagated
TOTAL_BYTES	NUMBER	Total bytes propagated

Column	Datatype	Description
ELAPSED_DEQUEUE_TIME	NUMBER	Elapsed dequeue time (in hundredths of a second)
ELAPSED_PICKLE_TIME	NUMBER	Elapsed pickle time (time taken to linearize a logical change record (LCR) into a stream of bytes that can be sent over the network) (in hundredths of a second)
ELAPSED_PROPAGATION_TIME	NUMBER	Elapsed propagation time (in hundredths of a second)
ELAPSED_RULE_TIME	NUMBER	Elapsed rule time (in hundredths of a second)
MAX_NUM_PER_WIN	NUMBER	Maximum bytes per window
MAX_SIZE	NUMBER	Maximum bytes sent per window
LAST_MSG_LATENCY	NUMBER	Last propagated message latency
LAST_MSG_ENQUEUE_TIME	TIMESTAMP(6)	Last propagated message enqueue time
LAST_MSG_PROPAGATION_TIME	TIMESTAMP(6)	Last time when the message was propagated
LAST_LCR_LATENCY	NUMBER	Last propagated LCR latency
LAST_LCR_CREATION_TIME	DATE	Last propagated LCR timestamp
LAST_LCR_PROPAGATION_TIME	DATE	Last time when the LCR was propagated
DST_DATABASE_NAME	VARCHAR2(395)	Global name of the destination database
SESSION_ID	NUMBER	Session ID of the propagation sender process
SERIAL#	NUMBER	Serial number of the propagation sender process
SPID	VARCHAR2(24)	Process identification number of the propagation sender process
PROPAGATION_NAME	VARCHAR2(128)	Name of the propagation
STATE	VARCHAR2(128)	State of the propagation sender process: • Initializing • Initializing propagation receiver • Browsing LCRs • Evaluating rules • Dequeueing LCRs • Sending LCRs • Waiting for apply to be enabled • Waiting for apply database to start • Waiting for propagation to be enabled • Waiting for capture to terminate • Waiting for a subscriber to be added • Suspended due to a dropped subscriber • Suspended for auto split/merge • Waiting on empty queue When the SCHEDULE_STATUS column is not SCHEDULE OPTIMIZED, the state is the value of the SCHEDULE_STATUS column.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.137 V$PROXY_ARCHIVEDLOG

V$PROXY_ARCHIVEDLOG contains descriptions of archived log backups that were taken using the proxy copy functionality.

In a proxy copy, the media manager takes over the operations of backing up and restoring data. Each row represents a backup of one control file.

Column	Datatype	Description
RECID	NUMBER	Proxy copy record identifier
STAMP	NUMBER	Proxy copy stamp
DEVICE_TYPE	VARCHAR2(17)	Type of media device that stores the proxy copy
HANDLE	VARCHAR2(513)	Name or "handle" for the proxy copy
COMMENTS	VARCHAR2(81)	Comments about the proxy copy
MEDIA	VARCHAR2(65)	A comment that contains further information about the media manager that created this backup
MEDIA_POOL	NUMBER	Number of the media pool in which the proxy copy is stored
TAG	VARCHAR2(32)	Tag for the proxy copy
STATUS	VARCHAR2(1)	Status of the backup set: • A - Available • U - Unavailable • X - Expired • D - Deleted
DELETED	VARCHAR2(3)	Indicates whether this record has been deleted (YES) or not (NO)
THREAD#	NUMBER	Number of the redo thread
SEQUENCE#	NUMBER	Log sequence number
RESETLOGS_CHANGE#	NUMBER	RESETLOGS SCN of the database incarnation to which this archived log belongs
RESETLOGS_TIME	DATE	RESETLOGS time stamp of the database incarnation to which this archived log belongs
FIRST_CHANGE#	NUMBER	First SCN of this redo log
FIRST_TIME	DATE	Time when Oracle switched into the redo log
NEXT_CHANGE#	NUMBER	First SCN of the next redo log in the thread
NEXT_TIME	DATE	First time stamp of the next redo log in the thread
BLOCKS	NUMBER	Size of this archived redo log (in operating system blocks)
BLOCK_SIZE	NUMBER	Block size for the copy (in bytes)
START_TIME	DATE	Time when the proxy copy was initiated
COMPLETION_TIME	DATE	Time when the proxy copy was completed
ELAPSED_SECONDS	NUMBER	Duration of the proxy copy
RMAN_STATUS_RECID	NUMBER	Owning V$RMAN_STATUS record ID
RMAN_STATUS_STAMP	NUMBER	Owning V$RMAN_STATUS stamp
TERMINAL	VARCHAR2(3)	Indicates whether this record corresponds to a terminal archived redo log, as defined in V$ARCHIVED_LOG (YES) or not (NO)
KEEP	VARCHAR2(3)	Indicates whether this backup set has a retention policy that is different than the value for the configure retention policy (YES) or not (NO)

Column	Datatype	Description
KEEP_UNTIL	DATE	If specified, then this is the date after which the backup becomes obsolete. If this column is NULL, then the backup never expires.
KEEP_OPTIONS	VARCHAR2(11)	Additional retention options for this backup set: • LOGS - Indicates a long-term backup made with the LOGS keyword, which is now deprecated • BACKUP_LOGS - Indicates that the backup was made in open mode, so archived log backups must be applied to make it consistent • NOLOGS - Indicates a consistent backup made when the database was mounted • NULL - Indicates that this backup has no KEEP options and becomes obsolete based on the retention policy
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.138 V$PROXY_ARCHIVELOG_DETAILS

V$PROXY_ARCHIVELOG_DETAILS contains information about all available archive log proxy copies.

Column	Datatype	Description
SESSION_KEY	NUMBER	Session identifier
SESSION_RECID	NUMBER	Session recid
SESSION_STAMP	NUMBER	Session stamp
COPY_KEY	NUMBER	Copy identifier
THREAD#	NUMBER	Redo thread number
SEQUENCE#	NUMBER	Redo log sequence number
RESETLOGS_CHANGE#	NUMBER	Resetlogs change number of the database when this log was written
RESETLOGS_TIME	DATE	Resetlogs time of the database when this log was written
HANDLE	VARCHAR2(513)	Proxy copy handle identifies the copy for restore
MEDIA	VARCHAR2(65)	Name of the media on which the copy resides. This value is informational only. It is not needed for restore.
MEDIA_POOL	NUMBER	Media pool in which the copy resides. This is the same value that was entered in the POOL operand of the Recovery Manager BACKUP command.
TAG	VARCHAR2(32)	Proxy copy tag
FIRST_CHANGE#	NUMBER	First change number in the archived log
NEXT_CHANGE#	NUMBER	First change number in the next log
FIRST_TIME	DATE	Timestamp of the first change
NEXT_TIME	DATE	Timestamp of the next change
OUTPUT_BYTES	NUMBER	Total output bytes written

Column	Datatype	Description
COMPLETION_TIME	DATE	Completion time
OUTPUT_BYTES_DISPLAY	VARCHAR2(3)	Displayable format for output bytes
KEEP	DATE	Indicates whether this backup set has a retention policy that is different than the value for the configure retention policy (YES) or not (NO)
KEEP_UNTIL	VARCHAR2(11)	If specified, then this is the date after which the backup becomes obsolete. If this column is NULL, then the backup never expires.
KEEP_OPTIONS	VARCHAR2(4000)	Additional retention options for this backup set: • LOGS - Indicates a long-term backup made with the LOGS keyword, which is now deprecated • BACKUP_LOGS - Indicates that the backup was made in open mode, so archived log backups must be applied to make it consistent • NOLOGS - Indicates a consistent backup made when the database was mounted • NULL - Indicates that this backup has no KEEP options and becomes obsolete based on the retention policy
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.139 V$PROXY_ARCHIVELOG_SUMMARY

V$PROXY_ARCHIVELOG_SUMMARY provides summary information about the output proxy archive log file.

Column	Datatype	Description
NUM_FILES_BACKED	NUMBER	Number of archived log files backed up
NUM_DISTINCT_FILES_BACKED	NUMBER	Number of distinct archived log files backed up
MIN_FIRST_CHANGE#	NUMBER	Minimum first change SCN
MAX_NEXT_CHANGE#	NUMBER	Maximum first change SCN
MIN_FIRST_TIME	DATE	Minimum first change time. Forms the redo range, along with MAX_NEXT_TIME.
MAX_NEXT_TIME	DATE	Maximum next change time
OUTPUT_BYTES	NUMBER	Total output size, in bytes
OUTPUT_BYTES_DISPLAY	VARCHAR2(4000)	Displayable format for output bytes
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.140 V$PROXY_COPY_DETAILS

V$PROXY_COPY_DETAILS contains information about all available control file and datafile proxy copies.

Column	Datatype	Description
SESSION_KEY	NUMBER	Session identifier
SESSION_RECID	NUMBER	Session recid
SESSION_STAMP	NUMBER	Session stamp
COPY_KEY	NUMBER	Copy identifier
FILE#	NUMBER	Absolute datafile number, or 0 if this is a control file backup
HANDLE	VARCHAR2(513)	Proxy copy handle identifies the copy for restore
MEDIA	VARCHAR2(65)	Name of the media on which the copy resides. This value is informational only. It is not needed for restore.
MEDIA_POOL	NUMBER	Media pool in which the copy resides. This is the same value that was entered in the POOL operand of the Recovery Manager BACKUP command.
TAG	VARCHAR2(32)	Proxy copy tag
CREATION_CHANGE#	NUMBER	Datafile creation change number
CREATION_TIME	DATE	Datafile creation timestamp
CHECKPOINT_CHANGE#	NUMBER	Checkpoint change number of the datafile when the copy was made
CHECKPOINT_TIME	DATE	Checkpoint timestamp of the datafile when the copy was made
OUTPUT_BYTES	NUMBER	Total output bytes written
COMPLETION_TIME	DATE	Completion time
CONTROLFILE_TYPE	VARCHAR2(1)	Type of control file: • B - Normal control file • S - Standby control file
KEEP	VARCHAR2(3)	Indicates whether this backup set has a retention policy that is different than the value for the configure retention policy (YES) or not (NO)
KEEP_UNTIL	DATE	If specified, then this is the date after which the backup becomes obsolete. If this column is NULL, then the backup never expires.
KEEP_OPTIONS	VARCHAR2(11)	Additional retention options for this backup set: • LOGS - Indicates a long-term backup made with the LOGS keyword, which is now deprecated • BACKUP_LOGS - Indicates that the backup was made in open mode, so archived log backups must be applied to make it consistent • NOLOGS - Indicates a consistent backup made when the database was mounted. • NULL - Indicates that this backup has no KEEP options and becomes obsolete based on the retention policy
OUTPUT_BYTES_DISPLAY	VARCHAR2(4000)	Displayable format for output bytes

Column	Datatype	Description
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.141 V$PROXY_COPY_SUMMARY

V$PROXY_COPY_SUMMARY provides summary information about the output proxy datafile and control file.

Column	Datatype	Description
NUM_COPIES	NUMBER	Number of copies created
NUM_DISTINCT_COPIES	NUMBER	Number of distinct copies (that contain datafiles with different checkpoints)
MIN_CHECKPOINT_CHANGE#	NUMBER	Minimum checkpoint change SCN
MAX_CHECKPOINT_CHANGE#	NUMBER	Maximum checkpoint change SCN
MIN_CHECKPOINT_TIME	DATE	Minimum checkpoint change time
MAX_CHECKPOINT_TIME	DATE	Maximum checkpoint change time
OUTPUT_BYTES	NUMBER	Total output bytes
OUTPUT_BYTES_DISPLAY	VARCHAR2(4000)	Displayable format for output bytes
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.142 V$PROXY_DATAFILE

V$PROXY_DATAFILE contains descriptions of datafile and control file backups that are taken with Proxy Copy. Each row represents a backup of one database file.

Column	Datatype	Description
RECID	NUMBER	Proxy copy record ID
STAMP	NUMBER	Proxy copy record stamp
DEVICE_TYPE	VARCHAR2(17)	Type of the device on which the copy resides
HANDLE	VARCHAR2(513)	Proxy copy handle identifies the copy for restore
COMMENTS	VARCHAR2(81)	Comment returned by the operating system or storage subsystem. This value is informational only. It is not needed for restore.

Column	Datatype	Description
MEDIA	VARCHAR2(65)	Name of the media on which the copy resides. This value is informational only. It is not needed for restore.
MEDIA_POOL	NUMBER	Media pool in which the copy resides. This is the same value that was entered in the POOL operand of the Recovery Manager BACKUP command
TAG	VARCHAR2(32)	Proxy copy tag
STATUS	VARCHAR2(1)	Status of the backup set: • A - Available • U - Unavailable • X - Expired • D - Deleted
DELETED	VARCHAR2(3)	Indicates whether this record has been deleted (YES) or not (NO)
FILE#	NUMBER	Absolute datafile number, or 0 if this is a control file backup
CREATION_CHANGE#	NUMBER	Datafile creation change number
CREATION_TIME	DATE	Datafile creation Timestamp
RESETLOGS_CHANGE#	NUMBER	Resetlogs change number of the datafile when the copy was made
RESETLOGS_TIME	DATE	Resetlogs timestamp of the datafile when the copy was made
CHECKPOINT_CHANGE#	NUMBER	Checkpoint change number of the datafile when the copy was made
CHECKPOINT_TIME	DATE	Checkpoint timestamp of the datafile when the copy was made
ABSOLUTE_FUZZY_CHANGE#	NUMBER	Highest change in any block of the file, if known
RECOVERY_FUZZY_CHANGE#	NUMBER	Highest change written to the file by media recovery
RECOVERY_FUZZY_TIME	DATE	Timestamp of the highest change written to the file by media recovery
INCREMENTAL_LEVEL	NUMBER	If this backup is part of an incremental backup strategy, then 0. Otherwise null.
ONLINE_FUZZY	VARCHAR2(3)	Indicates whether this copy was made after a crash or offline immediate (or is a copy of a copy which was taken improperly while the database was open) (YES) or not (NO). Recovery will need to apply all redo up to the next crash recovery marker to make the file consistent.
BACKUP_FUZZY	VARCHAR2(3)	Indicates whether this is a copy taken using the BEGIN BACKUP | END BACKUP technique (YES) or not (NO). The BEGIN BACKUP | END BACKUP technique is used internally when proxy copies of open files are created. Recovery will need to apply all redo up to the end backup marker to make this copy consistent.
BLOCKS	NUMBER	Size of the copy (in blocks). Also the size of the datafile when the copy was made.
BLOCK_SIZE	NUMBER	Block size of the datafile
OLDEST_OFFLINE_RANGE	NUMBER	If the file number is 0 (that is, this is a control file backup), the RECID of the oldest offline range record in this control file copy. 0 for datafile copies.
START_TIME	DATE	Starting time
COMPLETION_TIME	DATE	Completion time
ELAPSED_SECONDS	NUMBER	Number of elapsed seconds
CONTROLFILE_TYPE	VARCHAR2(1)	Type of control file: • B - Normal control file • S - Standby control file

Column	Datatype	Description
KEEP	VARCHAR2(3)	Indicates whether this backup set has a retention policy that is different than the value for the configure retention policy (YES) or not (NO)
KEEP_UNTIL	DATE	If specified, then this is the date after which the backup becomes obsolete. If this column is NULL, then the backup never expires.
KEEP_OPTIONS	VARCHAR2(11)	Additional retention options for this backup set: • LOGS - Indicates a long-term backup made with the LOGS keyword, which is now deprecated • BACKUP_LOGS - Indicates that the backup was made in open mode, so archived log backups must be applied to make it consistent • NOLOGS - Indicates a consistent backup made when the database was mounted. • NULL - Indicates that this backup has no KEEP options and becomes obsolete based on the retention policy
RMAN_STATUS_RECID	NUMBER	Owning V$RMAN_STATUS record ID
RMAN_STATUS_STAMP	NUMBER	Owning V$RMAN_STATUS stamp
FOREIGN_DBID	NUMBER	Foreign DBID of the database from which this datafile was transported. The value is 0 if the file backed up is not a foreign database file.
PLUGGED_READONLY	VARCHAR2(3)	Indicates whether this is a proxy copy of a transported read-only foreign file (YES) or not (NO)
PLUGIN_CHANGE#	NUMBER	SCN at which the foreign datafile was transported into the database. The value is 0 if this file is not a foreign database file.
PLUGIN_RESETLOGS_CHANGE#	NUMBER	SCN of the RESETLOGS operation for the incarnation into which this foreign file was transported. The value is 0 if this file is not a foreign database file.
PLUGIN_RESETLOGS_TIME	DATE	Time of the RESETLOGS operation for the incarnation into which this foreign file was transported. The value is 0 if this file is not a foreign database file.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data
BACKED_BY_PDB	VARCHAR2(3)	Recovery Manager (RMAN) allows a PDB to be backed up in two ways. The value in this column indicates how the PDB backup was taken: • YES: The backup was taken when connected to the PDB • NO: The backup was taken when connected to the root container
GUID	RAW(16)	The GUID of the PDB to which the backup belongs. This is useful after the PDB is dropped to identify which PDB the backup belongs to.

9.143 V$PROXY_PDB_TARGETS

V$PROXY_PDB_TARGETS provides information about the target of a proxy PDB.

Column	Datatype	Description
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: *n*: Where *n* is the container ID of a proxy PDB
TARGET_PORT	NUMBER	Port number that the target of the proxy PDB has registered with the listener, and which will be used by the proxy PDB to connect to the target
TARGET_HOST	VARCHAR2(128)	Host name where the target of the proxy PDB is running, and which will be used by the proxy PDB to connect to the target
TARGET_SERVICE	VARCHAR2(128)	Service name that the target of the proxy PDB has registered with the listener, and which will be used by the proxy PDB to connect to the target
TARGET_USER	VARCHAR2(128)	User name used by the proxy PDB to connect to the target of the proxy PDB. If null, the name of the connected user will be used.

See Also:

Oracle Multitenant Administrator's Guide for information about creating proxy PDBs

9.144 V$PWFILE_USERS

V$PWFILE_USERS lists all users in the password file, and indicates whether the user has been granted the SYSDBA, SYSOPER, SYSASM, SYSBACKUP, SYSDG, and SYSKM privileges.

Column	Datatype	Description
USERNAME	VARCHAR2(128)	Name of the user that is contained in the password file
SYSDBA	VARCHAR2(5)	Indicates whether the user can connect with SYSDBA privileges (TRUE) or not (FALSE)
SYSOPER	VARCHAR2(5)	Indicates whether the user can connect with SYSOPER privileges (TRUE) or not (FALSE)
SYSASM	VARCHAR2(5)	Indicates whether the user can connect with SYSASM privileges (TRUE) or not (FALSE)
SYSBACKUP	VARCHAR2(5)	Indicates whether the user can connect with SYSBACKUP privileges (TRUE) or not (FALSE)
SYSDG	VARCHAR2(5)	Indicates whether the user can connect with SYSDG privileges (TRUE) or not (FALSE)
SYSKM	VARCHAR2(5)	Indicates whether the user can connect with SYSKM privileges (TRUE) or not (FALSE)

Column	Datatype	Description
ACCOUNT_STATUS	VARCHAR2(30)	Account status: • OPEN • EXPIRED • EXPIRED (GRACE) • LOCKED (TIMED) • LOCKED • EXPIRED & LOCKED (TIMED) • EXPIRED & LOCKED • EXPIRED (GRACE) & LOCKED
PASSWORD_PROFILE	VARCHAR2(128)	Password profile name
LAST_LOGIN	TIMESTAMP(9) WITH TIME ZONE	The time of the last user login This column is not populated when a user connects to the database with administrative privileges, that is, AS { SYSASM | SYSBACKUP | SYSDBA | SYSDG | SYSOPER | SYSRAC | SYSKM }. The exception is as follows: If a user connects to the database with administrative privileges from a *remote* session, and the database password file format is 12.2 (V$PASSWORDFILE_INFO.FORMAT = 12.2), then this column is populated.
LOCK_DATE	DATE	Date the account was locked if account status was LOCKED
EXPIRY_DATE	DATE	Date of expiration of the account
EXTERNAL_NAME	VARCHAR2(1024)	Shows Certificate DN or Principal Name of externally authenticated users
AUTHENTICATION_TYPE	VARCHAR2(8)	Indicates the authentication mechanism for the user: • EXTERNAL - CREATE USER *user1* IDENTIFIED EXTERNALLY; • GLOBAL - CREATE USER *user2* IDENTIFIED GLOBALLY; • PASSWORD - CREATE USER *user3* IDENTIFIED BY *password*;
COMMON	VARCHAR2(3)	This column has a value of YES if an administrative privilege (for example, SYSDBA) was granted with CONTAINER=ALL. Otherwise, the column has a value of NO.
PASSWORD_VERSIONS	VARCHAR2(12)	Shows the list of versions of the password hashes (also known as "verifiers") existing for the account. The values for this column can include: • 11G: If a SHA-1 hash exists • 12C: If a de-optimized PBKDF2-based hash exists For more information about the 12C verifier, see *Oracle Database Concepts*. Note that any combination of these verifiers can exist for any given account.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.145 V$PX_INSTANCE_GROUP

V$PX_INSTANCE_GROUP provides information about the instance groups being used for parallel operations by the current session.

Column	Datatype	Description
QC_INSTANCE_GROUP	VARCHAR2(64)	The instance group being used by this session for parallel operations. This value repeats for every row returned.
WHY	VARCHAR2(23)	Where the current instance group name comes from, as follows: SERVICE - the instance group being used is from the session's service name PARALLEL_INSTANCE_GROUP - the instance group being used is being used because the PARALLEL_INSTANCE_GROUP initialization parameter has been set. The value repeats for every row returned.
INSTANCE_NUMBER	NUMBER	Instance number of the instance providing this instance group. There will be one row for each instance in the instance group that the Query Coordinator is using.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

See Also:

"PARALLEL_INSTANCE_GROUP"

9.146 V$PX_PROCESS

V$PX_PROCESS contains information about the sessions running parallel execution.

Column	Datatype	Description
SERVER_NAME	VARCHAR2(4)	The name of the PX server (P000, P001, and so on)
STATUS	VARCHAR2(9)	The state of the PX server (IN USE or AVAILABLE)
PID	NUMBER	The process identifier
SPID	VARCHAR2(24)	Operating system process ID
SID	NUMBER	The session ID of the PX server, if in use
SERIAL#	NUMBER	The session serial number of the PX server, if in use

Column	Datatype	Description
IS_GV	VARCHAR2(5)	Indicates whether a PX server process in parallel is a normal process or a GV$ process. Possible values: • FALSE: The PX server process in parallel is a normal process • TRUE: The PX server process in parallel is a GV$ process
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.147 V$PX_PROCESS_DETAIL

V$PX_PROCESS_DETAIL lists statistics for each of the active PX (Parallel Execution) servers on an instance.

Column	Datatype	Description
SPID	VARCHAR2(24)	Operating system process ID
SERVER_NAME	VARCHAR2(7)	Name of the PX server (P000, P001, and so on)
STATISTIC	VARCHAR2(30)	Name of the statistic: • Start Time • Cleanup Start Time • Cleanup End Time
VALUE	VARCHAR2(64)	String representation of the value the statistic For example, if the value of the statistic is a timestamp, this column displays a string containing the timestamp information.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

> **Note:**
>
> This view is available starting with Oracle Database 23ai.

9.148 V$PX_PROCESS_SYSSTAT

V$PX_PROCESS_SYSSTAT contains information about the sessions running parallel execution.

Column	Datatype	Description
STATISTIC	VARCHAR2(30)	Name of the statistic: • Buffers Allocated - Number of times a message buffer has been allocated • Buffers Current - Number of message buffers currently being used • Buffers Freed - Number of times a message buffer has been freed • Buffers HWM - Maximum number of concurrently allocated message buffers • Chunk Lists - Number of memory chunk lists • Memory Chunks Allocated - Number of large memory chunks allocated by PX servers • Memory Chunks Current - Number of large memory chunks currently being used • Memory Chunks Freed - Number of large memory chunks freed • Memory Chunks HWM - Maximum number of concurrently allocated chunks • Memory Chunks Permanent - Number of large memory chunks allocated permanently • Server Sessions - Total number of sessions created by all PX servers • Servers Available - Number of PX servers available to perform parallel operations • Servers Cleaned Up - Number of times PMON had to clean up a PX server. This should only happen during nonstandard termination of a parallel operation. If this number is large, then you should determine the cause. • Servers Highwater - Maximum number of concurrent PX server processes If this number is equal to the PARALLEL_MAX_SERVERS initialization parameter, then consider increasing the parameter. This could allow you to increase your throughput, especially if your system is under-utilized and the V$SYSSTAT statistic "Parallel operations downgraded to serial" is large. • Servers In Use - Number of PX servers currently performing parallel operations • Servers Shutdown - Number of times a PX server process has been shut down. A PX server process will be shut down if it has not been used recently. If this value is large, then consider increasing the parameter. This will improve performance by avoiding the latency of PX server process creation. • Servers Started - Number of times the system has had to create a PX server process
VALUE	NUMBER	Value of the statistic
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.149 V$PX_SERVER

V$PX_SERVER lists statistics for each of the active PX (Parallel Execution) servers on an instance.

Column	Datatype	Description
SERVER_NAME	VARCHAR2(4)	Name of the PX server
STATUS	VARCHAR2(4)	Current status of the PX server: • BUSY • IDLE
SESSIONS	NUMBER	Number of sessions that have used this PX server
IDLE_TIME_CUR	NUMBER	Amount of time spent idle while processing statements in the current session
BUSY_TIME_CUR	NUMBER	Amount of time spent busy while processing statements in the current session
CPU_SECS_CUR	NUMBER	Amount of CPU time (in seconds) spent on the current session
MSGS_SENT_CUR	NUMBER	Number of messages sent while processing statements for the current session
MSGS_RCVD_CUR	NUMBER	Number of messages received while processing statements for the current session
IDLE_TIME_TOTAL	NUMBER	Total amount of time (in minutes) this PX server has been idle
BUSY_TIME_TOTAL	NUMBER	Total amount of time (in minutes) this PX server has been active
CPU_SECS_TOTAL	NUMBER	Total amount of CPU time (in seconds) this PX server has used to process statements
MSGS_SENT_TOTAL	NUMBER	Total number of messages this PX server has sent
MSGS_RCVD_TOTAL	NUMBER	Total number of messages this PX server has received
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

Note:

This view is available starting with Oracle Database 23ai.

9.150 V$PX_SESSION

V$PX_SESSION contains information about the sessions running parallel execution.

Column	Datatype	Description
SADDR	RAW(4 | 8)	Session address
SID	NUMBER	Session identifier
SERIAL#	NUMBER	Session serial number
QCSID	NUMBER	Session identifier of the parallel coordinator
QCSERIAL#	NUMBER	Session serial number of the parallel coordinator
QCINST_ID	NUMBER	Instance number on which the parallel coordinator is running
SERVER_GROUP	NUMBER	The logical group of PX servers to which this cluster database process belongs
SERVER_SET	NUMBER	The logical set of PX servers to which this cluster database process belongs. A single PX server group will have at most two PX server sets.
SERVER#	NUMBER	The logical number of the cluster database process within a PX server set
DEGREE	NUMBER	The degree of parallelism being used by the PX server set
REQ_DEGREE	NUMBER	The degree of parallelism that was requested by the user when the statement was issued and before any resource, multiuser, or load balancing reductions
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.151 V$PX_SESSTAT

V$PX_SESSTAT contains information about the sessions running parallel execution.

Column	Datatype	Description
SADDR	RAW(4 | 8)	Session address
SID	NUMBER	Session identifier
SERIAL#	NUMBER	Session serial number
QCSID	NUMBER	Session identifier of the parallel coordinator
QCSERIAL#	NUMBER	Session serial number of the parallel coordinator
QCINST_ID	NUMBER	Instance number on which the parallel coordinator is running
SERVER_GROUP	NUMBER	The logical group of PX servers to which this cluster database process belongs
SERVER_SET	NUMBER	The logical set of PX servers that this cluster database process belongs to. A single PX server group will have at most two PX server sets.
SERVER#	NUMBER	The logical number of the cluster database process within a PX server set
DEGREE	NUMBER	The degree of parallelism being used by the PX server set

Column	Datatype	Description
REQ_DEGREE	NUMBER	The degree of parallelism that was requested by the user when the statement was issued and before any resource, multiuser, or load balancing reductions
STATISTIC#	NUMBER	Statistic number
VALUE	NUMBER	Statistic value
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.152 V$QMON_COORDINATOR_STATS

V$QMON_COORDINATOR_STATS displays statistics of the non-sharded queue primary process. There is one row per instance. The rows are deleted when the database (or instance in an Oracle RAC environment) restarts.

Column	Datatype	Description
QMNC_PID	VARCHAR2(24)	Non-sharded queue primary process ID
STATUS	VARCHAR2(24)	Current status of the coordinator: • DEAD • INITIALIZING • RUNNING TASK COORDINATOR • WAITING • ADDING SERVER
NUM_SERVERS	NUMBER	Number of QMON servers currently running
LAST_SERVER_START_TIME	TIMESTAMP(3) WITH TIME ZONE	Last server startup time
LAST_SERVER_PID	VARCHAR2(24)	Process ID of the last server process (Qnnn) created
NEXT_WAKEUP_TIME	TIMESTAMP(3) WITH TIME ZONE	Next wakeup time of the coordinator
NEXT_READY_TIME	TIMESTAMP(3) WITH TIME ZONE	Ready time of the first delayed task
NEXT_EXPIRY_TIME	TIMESTAMP(3) WITH TIME ZONE	Expiry time of the next ready task
LAST_WAIT_TIME	TIMESTAMP(3) WITH TIME ZONE	Time when the coordinator went to sleep
LAST_FAILURE	VARCHAR2(32)	Last failure encountered
LAST_FAILURE_TIME	TIMESTAMP(3) WITH TIME ZONE	Last failure time
MAX_TASK_LATENCY	VARCHAR2(40)	Maximum task latency across all the servers (in seconds)
MIN_TASK_LATENCY	VARCHAR2(40)	Minimum task latency across all the servers (in seconds)
TOTAL_TASK_LATENCY	NUMBER	Cumulative latency across all the tasks (in seconds)
TOTAL_TASKS_EXECUTED	NUMBER	Cumulative number of tasks serviced by all the servers

Column	Datatype	Description
MAX_SERVERS	NUMBER	Maximum number of servers present at any point of time
CON_ID	NUMBER	The ID of the container to which the data pertains. The CON_ID value in this view is always 0. The rows pertain to the entire CDB or to the non-CDB.

9.153 V$QMON_SERVER_STATS

V$QMON_SERVER_STATS displays information and statistics about the active queue monitor server processes. There is one row per live queue monitor server process. The rows are deleted when the database (or instance in an Oracle RAC environment) restarts.

Column	Datatype	Description
QMNC_PID	VARCHAR2(24)	Non-sharded queue primary process ID
SERVER_PID	VARCHAR2(24)	Process ID of the server
SERVER_NAME	VARCHAR2(48)	Name of the server
STATUS	VARCHAR2(40)	Current state of the server: • UNUSED • RUNNING • IDLE WAIT • EXITING • NOT ACTIVE
SERVER_START_TIME	TIMESTAMP(3) WITH TIME ZONE	Start time of the server
TASK_NAME	VARCHAR2(32)	Current executing task
TASK_NUMBER	NUMBER	Unique task number of the running task
TASK_START_TIME	TIMESTAMP(3) WITH TIME ZONE	Start time of the running task
LAST_WAIT_TIME	TIMESTAMP(3) WITH TIME ZONE	Time when the server last waited
MAX_LATENCY	NUMBER	Maximum task latency for this server (in seconds)
MIN_LATENCY	NUMBER	Minimum task latency for this server (in seconds)
TOTAL_LATENCY	NUMBER	Cumulative task latency for this server (in seconds)
NUM_TASKS	NUMBER	Number of tasks processed by the server
LAST_FAILURE	VARCHAR2(128)	Last failure encountered by the server
LAST_FAILURE_TIME	TIMESTAMP(3) WITH TIME ZONE	Last failure time
LAST_FAILURE_TASK	VARCHAR2(32)	Task being run at the time of the last failure
LAST_FAILURE_TASKNUM	NUMBER	Unique task number of the failed task
CON_ID	NUMBER	The ID of the container to which the data pertains. The CON_ID value in this view is always 0. The rows pertain to the entire CDB or to the non-CDB.

9.154 V$QMON_TASK_STATS

V$QMON_TASK_STATS displays information and statistics based on different queue monitor tasks in the system (spilling, time manager activity, and so on). There is one row per kind of task. The rows are deleted when the database (or instance in an Oracle RAC environment) restarts.

Column	Datatype	Description
TASK_NAME	VARCHAR2(32)	Name of the task
TASK_TYPE	VARCHAR2(40)	Type of the task
LAST_CREATED_TASKNUM	NUMBER	Unique task number last created for this task
NUM_TASKS	NUMBER	Number of tasks currently present
TOTAL_TASK_RUN_TIME	NUMBER	Cumulative task run time
TOTAL_TASK_RUNS	NUMBER	Cumulative task runs
TOTAL_TASK_FAILURES	NUMBER	Cumulative failures
METRIC_TYPE	VARCHAR2(50)	Type of metric gathered for this task type
METRIC_VALUE	NUMBER	Value of this metric
LAST_FAILURE	VARCHAR2(32)	Last failure encountered while executing this type of task
LAST_FAILURE_TIME	TIMESTAMP(3) WITH TIME ZONE	Time when the last failure occurred
LAST_FAILURE_TASKNUM	NUMBER	Task number of the last failed task for this task
REMARK	VARCHAR2(64)	Remarks about the task
CON_ID	NUMBER	The ID of the container to which the data pertains. The CON_ID value in this view is always 0. The rows pertain to the entire CDB or to the non-CDB.

9.155 V$QMON_TASKS

V$QMON_TASKS displays information and statistics about all queue background tasks in the system, which would be served by queue monitor servers. There is one row per task. The rows are deleted when the database (or instance in an Oracle RAC environment) restarts.

Column	Datatype	Description
TASK_NAME	VARCHAR2(32)	Task name
TASK_NUMBER	NUMBER	Unique task number
TASK_TYPE	VARCHAR2(40)	Task type
TASK_SUBMIT_TIME	TIMESTAMP(3) WITH TIME ZONE	Task submit time
TASK_READY_TIME	TIMESTAMP(3) WITH TIME ZONE	Task ready time
TASK_EXPIRY_TIME	TIMESTAMP(3) WITH TIME ZONE	Time when this task expires
TASK_START_TIME	TIMESTAMP(3) WITH TIME ZONE	Last actual start time for the task
TASK_STATUS	VARCHAR2(32)	Status of the task
SERVER_NAME	VARCHAR2(48)	Name of the QMON server running this task

Column	Datatype	Description
MAX_RETRIES	NUMBER	Maximum retry count for the task
NUM_RUNS	NUMBER	Number of runs of the task if repeatable
NUM_FAILURES	NUMBER	Number of failures encountered while running the task
CON_ID	NUMBER	The ID of the container to which the data pertains. The CON_ID value in this view is always 0. The rows pertain to the entire CDB or to the non-CDB.

9.156 V$QUARANTINE

V$QUARANTINE provides information about quarantined objects.

Column	Datatype	Description
OBJECT	VARCHAR2(64)	Type of quarantined object (description of the memory)
ADDRESS	RAW(8)	Address of the object
BYTES	NUMBER	Amount of memory used by the object (this is the minimum amount; the actual amount may be unknown)
ERROR	VARCHAR2(128)	Oracle error causing quarantine
TIMESTAMP	TIMESTAMP(6) WITH TIME ZONE	Time the object was quarantined
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.157 V$QUARANTINE_SUMMARY

V$QUARANTINE_SUMMARY provides a summary of quarantine for each pluggable database (PDB).

Column	Datatype	Description
OBJECT_COUNT	NUMBER	Number of objects in quarantine
OBJECT_LIMIT	NUMBER	Limit on the number of objects that can be in quarantine before a termination will occur
MEMORY_TOTAL	NUMBER	Number of bytes in quarantine
MEMORY_LIMIT	NUMBER	Limit on the amount of memory that can be in quarantine before a termination will occur
RECENT_COUNT	NUMBER	Number of objects recently placed in quarantine (within the last hour)
RECENT_LIMIT	NUMBER	Limit on the number of recently quarantined objects before a termination will occur

Column	Datatype	Description
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

> **See Also:**
>
> "V$QUARANTINE" for additional details on objects recently placed into quarantine

9.158 V$QUEUE

V$QUEUE contains information on the shared server message queues.

Column	Datatype	Description
PADDR	RAW(4 | 8)	Address of the process that owns the queue
TYPE	VARCHAR2(10)	Type of queue: • COMMON - Processed by servers • DISPATCHER
QUEUED	NUMBER	Number of items in the queue
WAIT	NUMBER	Total time that all items in this queue have waited (in hundredths of a second). Divide by TOTALQ for average wait per item.
TOTALQ	NUMBER	Total number of items that have ever been in the queue
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.159 V$QUEUEING_MTH

V$QUEUEING_MTH displays all available queuing resource allocation methods.

Column	Datatype	Description
NAME	VARCHAR2(40)	Name of the queuing resource allocation method

Column	Datatype	Description
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.160 V$RAC_TWO_STAGE_ROLLING_UPDATES

V$RAC_TWO_STAGE_ROLLING_UPDATES displays information about Oracle RAC two-stage rolling updates.

Column	Datatype	Description
BUG_NUMBER	VARCHAR2(11)	Bug number for the two-stage rolling update
ENABLED	VARCHAR2(1)	Indicates whether the two-stage rolling update is enabled (Y) or disabled (N)
ONLINE_DISABLE	VARCHAR2(1)	Indicates whether the two-stage rolling update can be disabled while the database instance is running (Y) or not (N)
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

> **Note:**
>
> This view is available starting with Oracle Database 23ai.

9.161 V$RECOVER_FILE

V$RECOVER_FILE displays the status of files needing media recovery.

Column	Datatype	Description
FILE#	NUMBER	File identifier number
ONLINE	VARCHAR2(7)	This column is obsolete and maintained for backward compatibility. The value of this column is always equal to the value in ONLINE_STATUS.
ONLINE_STATUS	VARCHAR2(7)	Online status (ONLINE, OFFLINE)
ERROR	VARCHAR2(18)	Why the file must be recovered: NULL if reason unknown, or OFFLINE NORMAL if recovery not needed

Column	Datatype	Description
CHANGE#	NUMBER	SCN where recovery must start
TIME	DATE	Time of SCN when recovery must start
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.162 V$RECOVERY_AREA_USAGE

V$RECOVERY_AREA_USAGE displays usage information about recovery areas.

Column	Datatype	Description
FILE_TYPE	VARCHAR2(23)	File type: • CONTROL FILE • REDO LOG • ARCHIVED LOG • BACKUP PIECE • IMAGE COPY • FLASHBACK LOG • REMOTE ARCHIVED LOG
PERCENT_SPACE_USED	NUMBER	Percent of the recovery area that is in use
PERCENT_SPACE_RECLAIMABLE	NUMBER	Percent of the recovery area that is reclaimable
NUMBER_OF_FILES	NUMBER	Number of files in the recovery area
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.163 V$RECOVERY_FILE_DEST

V$RECOVERY_FILE_DEST displays information about the disk quota and current disk usage in the fast recovery area.

Column	Datatype	Description
NAME	VARCHAR2(513)	Location name. This is the value specified in the DB_RECOVERY_FILE_DEST initialization parameter.

Column	Datatype	Description
SPACE_LIMIT	NUMBER	Maximum amount of disk space (in bytes) that the database can use for the fast recovery area. This is the value specified in the DB_RECOVERY_FILE_DEST_SIZE initialization parameter.
SPACE_USED	NUMBER	Amount of disk space (in bytes) used by fast recovery area files created in current and all previous fast recovery areas. Changing fast recovery areas does not reset SPACE_USED to 0.
SPACE_RECLAIMABLE	NUMBER	Total amount of disk space (in bytes) that can be created by deleting obsolete, redundant, and other low priority files from the fast recovery area
NUMBER_OF_FILES	NUMBER	Number of files in the fast recovery area
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

See Also:

- "DB_RECOVERY_FILE_DEST"
- "DB_RECOVERY_FILE_DEST_SIZE"

9.164 V$RECOVERY_FILE_STATUS

V$RECOVERY_FILE_STATUS contains one row for each datafile for each RECOVER statement. This view contains useful information only for the Oracle process doing the recovery. When Recovery Manager directs a server process to perform recovery, only Recovery Manager can view the relevant information in this view. V$RECOVERY_FILE_STATUS will be empty to all other Oracle users.

Column	Datatype	Description
FILENUM	NUMBER	Number of the file being recovered
FILENAME	VARCHAR2(513)	Filename of the datafile being recovered
STATUS	VARCHAR2(13)	Status of the recovery: • IN RECOVERY • CURRENT • NOT RECOVERED

Column	Datatype	Description
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

> **See Also:**
>
> *Oracle Database Backup and Recovery User's Guide*

9.165 V$RECOVERY_LOG

V$RECOVERY_LOG lists information about archived logs that are needed to complete media recovery. This information is derived from the log history view, V$LOG_HISTORY.

V$RECOVERY_LOG contains useful information only for the Oracle process doing the recovery. When Recovery Manager directs a server process to perform recovery, only Recovery Manager can view the relevant information in this view. V$RECOVERY_LOG will be empty to all other Oracle users.

Column	Datatype	Description
THREAD#	NUMBER	Thread number of the archived log
SEQUENCE#	NUMBER	Sequence number of the archived log
TIME	DATE	Time of the first entry (lowest SCN) in the log
ARCHIVE_NAME	VARCHAR2(513)	Name of the file when archived, using the naming convention specified by the LOG_ARCHIVE_FORMAT initialization parameter **See Also:** "LOG_ARCHIVE_FORMAT"
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

> **See Also:**
>
> "V$LOG_HISTORY" and *Oracle Database Backup and Recovery User's Guide*

9.166 V$RECOVERY_PROGRESS

V$RECOVERY_PROGRESS can be used to track database recovery operations to ensure that they are not stalled, and also to estimate the time required to complete the operation in progress.

On non-coordinator instances, V$RECOVERY_PROGRESS is not populated.

On the coordinator instance (the instance where MRP0 was started to start recovery), V$RECOVERY_PROGRESS has the same set of rows as before, except the following rows in the ITEM column are always 0 (not used) with Multi-Instance Redo Apply:

- Active Apply
- Maximum Apply Rate
- Apply Time per Log
- Checkpoint Time per Log
- Recovery ID

V$RECOVERY_PROGRESS is a subview of V$SESSION_LONGOPS.

> **Note:**
>
> This view is populated on the instance where the MRP0 process is started if recovery is running in Multi-Instance Redo Apply mode. Not all the columns will be populated.

Column	Datatype	Description
START_TIME	DATE	Start time of the recovery operation
TYPE	VARCHAR2(64)	Type of recovery operation being performed: • CRASH RECOVERY • INSTANCE RECOVERY • MEDIA RECOVERY
ITEM	VARCHAR2(32)	Item being measured. When TYPE is CRASH RECOVERY or INSTANCE RECOVERY, the possible values are: • Log Files • Redo Blocks When TYPE is MEDIA RECOVERY, the possible values are: • Active Apply Rate • Average Apply Rate • Maximum Apply Rate • Redo Applied • Log Files • Last Applied Redo • Active Time • Elapsed Time • Apply Time per Log • Checkpoint Time per Log • Standby Apply Lag • Recovery ID

Column	Datatype	Description
UNITS	VARCHAR2(32)	The units of measurement for each item
SOFAR	NUMBER	Amount of work done so far
TOTAL	NUMBER	Total amount of work expected
TIMESTAMP	DATE	Timestamp of the last redo record applied
COMMENTS	VARCHAR2(248)	Miscellaneous notes; currently displays the SCN for the last applied redo
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

See Also:

- "V$SESSION_LONGOPS"
- "Background Processes" for more information about the MRP0 process
- *Oracle Database Backup and Recovery User's Guide* for more information about performing database recovery

9.167 V$RECOVERY_SLAVE

V$RECOVERY_SLAVE is used to track database media recovery processes to monitor their performance statistics and analyze a media recovery session.

Note:

This view is deprecated. Oracle recommends that you instead use the V$RECOVERY_WORKER view.

See Also:

"V$RECOVERY_WORKER"

9.168 V$RECOVERY_STATUS

V$RECOVERY_STATUS contains statistics of the current recovery process. This view contains useful information only for the Oracle process doing the recovery. When Recovery Manager

directs a server process to perform recovery, only Recovery Manager can view the relevant information in this view. V$RECOVERY_STATUS will be empty to all other Oracle users.

Column	Datatype	Description
RECOVERY_CHECKPOINT	DATE	Point in time to which the recovery has occurred. If no logs have been applied, this is the point in time the recovery starts.
THREAD	NUMBER	Number of the redo thread currently being processed
SEQUENCE_NEEDED	NUMBER	Log sequence number of the log needed by the recovery process. The value is 0 if no log is needed.
SCN_NEEDED	VARCHAR2(16)	Low SCN of the log needed by recovery. The value is 0 if unknown or no log is needed.
TIME_NEEDED	DATE	Time when the log was created. The value is midnight on 1/1/88 if the time is unknown or if no log is needed.
PREVIOUS_LOG_NAME	VARCHAR2(513)	Filename of the log
PREVIOUS_LOG_STATUS	VARCHAR2(13)	Status of the previous log. Contains one of the following values: RELEASE; WRONG NAME; MISSING NAME; UNNEEDED NAME; NONE
REASON	VARCHAR2(13)	Reason recovery is returning control to the user (NEED LOG|LOG REUSED|THREAD DISABLED)
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

See Also:

Oracle Database Backup and Recovery User's Guide

9.169 V$RECOVERY_WORKER

V$RECOVERY_WORKER is used to track database media recovery processes to monitor their performance statistics and analyze a media recovery session.

Column	Datatype	Description
START_TIME	DATE	Start time of the recovery process
TYPE	VARCHAR2(64)	Type of recovery process being performed: • Media Recovery Applier # (where # represents the worker ID number) • Media Recovery Log Merger • Serial Media Recovery

Column	Datatype	Description
ITEM	VARCHAR2(32)	Item being measured. When TYPE is Media Recovery Applier #, the possible values are: • Recovery ID • Process ID • Number of Redo Cache Full • Number of Redo Cache Copy • Number of CV Cached • CV Applied OK • CV Applied Stuck • CV Applied Repair • CV Applied Corrupt • CV Applied Ckpt • CV Applied Reapplied • Total CV Processed Size • Total CV Applied • Number of Buffer Cache Full • Number of Buffer Retries • Number of Max Reads Issued • Number of Unrcv Condition • Number of Influx Buffer Flushed • Number of Reap Request • Number of Reap Wait IO • Number of Reap No Buffer • Number of Wait All Read • Number of Buffer Pinged • Buffer Ping Time • Read Issue Time • Number of Read Request Issued When TYPE is Media Recovery Log Merger, the possible values are: • Recovery ID • Process ID • Number of Redo Cache Full • Number of Redo Cache Copy • Number of CV Cached • Total Redo Read Bytes • Total CV Parsed When TYPE is Serial Media Recovery, the possible values include values from the other two TYPEs.
UNITS	VARCHAR2(32)	The units of measurement for each item
SOFAR	NUMBER	Amount of work done so far
TOTAL	NUMBER	Total amount of work expected
COMMENTS	VARCHAR2(248)	Miscellaneous notes, which may display the recovery ID and process ID for current recovery session

Column	Datatype	Description
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

> **Note:**
>
> This view is available starting with Oracle Database 23ai.

> **See Also:**
>
> *Oracle Database Backup and Recovery User's Guide*

9.170 V$REDO_DEST_RESP_HISTOGRAM

V$REDO_DEST_RESP_HISTOGRAM provides statistical information for each redo transport destination.

Column	Datatype	Description
DEST_ID	NUMBER	A nonnegative integer value representing a SYNC standby destination
TIME	VARCHAR2(20)	A text string that shows the last wall-clock time that a bucket was hit
DURATION	NUMBER	A positive integer value that represents a bucket of seconds, 1, 2, 3, up to 300 seconds, followed by 5 additional buckets that represent 600, 1200, 2400, 4800, and 9600 (>= 4801) seconds
FREQUENCY	NUMBER	A nonnegative integer that shows the number of times a particular bucket was hit by the destination specified by DEST_ID
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.171 V$REQDIST

V$REQDIST lists statistics for the histogram of shared server dispatcher request times, divided into 12 buckets, or ranges of time. The time ranges grow exponentially as a function of the bucket number.

Column	Datatype	Description
BUCKET	NUMBER	Bucket number: 0 - 11; the maximum time for each bucket is (4 * 2^N)/100 seconds
COUNT	NUMBER	Count of requests whose total time to complete (excluding wait time) falls in this range
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.172 V$RESERVED_WORDS

V$RESERVED_WORDS displays a list of all SQL keywords. To determine whether a particular keyword is reserved in any way, check the RESERVED, RES_TYPE, RES_ATTR, and RES_SEMI columns.

Column	Datatype	Description
KEYWORD	VARCHAR2(128)	Name of the keyword
LENGTH	NUMBER	Length of the keyword
RESERVED	VARCHAR2(1)	Indicates whether the keyword cannot be used as an identifier (Y) or whether the keyword is not reserved (N)
RES_TYPE	VARCHAR2(1)	Indicates whether the keyword cannot be used as a type name (Y) or whether the keyword is not reserved (N)
RES_ATTR	VARCHAR2(1)	Indicates whether the keyword cannot be used as an attribute name (Y) or whether the keyword is not reserved (N)
RES_SEMI	VARCHAR2(1)	Indicates whether the keyword is not allowed as an identifier in certain situations, such as in DML (Y) or whether the keyword is not reserved (N)
DUPLICATE	VARCHAR2(1)	Indicates whether the keyword is a duplicate of another keyword (Y) or whether the keyword is not a duplicate (N)
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.173 V$RESOURCE

V$RESOURCE contains resource name and address information.

Column	Datatype	Description
ADDR	RAW(4 | 8)	Address of the resource object
TYPE	VARCHAR2(2)	Resource type; the resource types are listed in Table 9-1
ID1	NUMBER	Resource identifier #1
ID2	NUMBER	Resource identifier #2
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.174 V$RESOURCE_LIMIT

V$RESOURCE_LIMIT displays information about global resource use for some of the system resources. Use this view to monitor the consumption of resources so that you can take corrective action, if necessary. Many of the resources correspond to initialization parameters listed in Table 9-5.
Some resources, those used by DLM for example, have an initial allocation (soft limit), and the hard limit, which is theoretically infinite (although in practice it is limited by SGA size). During SGA reservation/initialization, a place is reserved in SGA for the INITIAL_ALLOCATION of resources, but if this allocation is exceeded, additional resources are allocated up to the value indicated by LIMIT_VALUE. The CURRENT_UTILIZATION column indicates whether the initial allocation has been exceeded. When the initial allocation value is exceeded, the additional required resources are allocated from the shared pool, where they must compete for space with other resources.

A good choice for the value of INITIAL_ALLOCATION will avoid the contention for space. For most resources, the value for INITIAL_ALLOCATION is the same as the LIMIT_VALUE. Exceeding LIMIT_VALUE results in an error.

Column	Datatype	Description
RESOURCE_NAME	VARCHAR2(128)	Name of the resource (see Table 9-5)
CURRENT_UTILIZATION	NUMBER	Number of (resources, locks, or processes) currently being used
MAX_UTILIZATION	NUMBER	Maximum consumption of this resource since the last instance start-up
INITIAL_ALLOCATION	VARCHAR2(10)	Initial allocation. This will be equal to the value specified for the resource in the initialization parameter file (UNLIMITED for infinite allocation).
LIMIT_VALUE	VARCHAR2(10)	Unlimited for resources and locks. This can be greater than the initial allocation value (UNLIMITED for infinite limit).
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

Table 9-5 Values for the RESOURCE_NAME Column

Resource Name	Corresponds to
DML_LOCKS	See "DML_LOCKS"
ENQUEUE_LOCKS	This value is computed by the Oracle Database. See V$ENQUEUE_LOCK to obtain more information about the enqueue locks.
GES_LOCKS	Global Enqueue Service locks
GES_PROCS	Global Enqueue Service processes
GES_RESS	Global Enqueue Service resources
MAX_SHARED_SERVERS	See "MAX_SHARED_SERVERS"
PARALLEL_MAX_SERVERS	See "PARALLEL_MAX_SERVERS"
PROCESSES	See "PROCESSES"
SESSIONS	See "SESSIONS"
SORT_SEGMENT_LOCKS	This value is computed by the Oracle Database
TEMPORARY_LOCKS	This value is computed by the Oracle Database
TRANSACTIONS	See "TRANSACTIONS"

9.175 V$RESTORE_POINT

V$RESTORE_POINT displays information about restore points.

Column	Datatype	Description
SCN	NUMBER	Database SCN when the restore point was created
DATABASE_INCARNATION#	NUMBER	Database incarnation number when the restore point was created
GUARANTEE_FLASHBACK_DATABASE	VARCHAR2(3)	Indicates whether flashback log files will be kept to ensure a flashback to this point (YES) or not (NO)
STORAGE_SIZE	NUMBER	Approximate number of bytes of disk space currently tied up supporting this restore point, and which would no longer be tied up if this restore point is the oldest restore point and it is dropped. This will always be zero for non-guaranteed restore points.
TIME	TIMESTAMP(9)	Wall-clock time when the restore point was created
RESTORE_POINT_TIME	TIMESTAMP(9)	Time that was specified when the restore point was created. If a time was not specified, this value is NULL.
PRESERVED	VARCHAR2(3)	Indicates whether the restore point must be explicitly deleted (YES) or not (NO)
NAME	VARCHAR2(128)	Name of the restore point
PDB_RESTORE_POINT	VARCHAR2(3)	Indicates whether there is a PDB restore point for this PDB (YES) or not (NO)
CLEAN_PDB_RESTORE_POINT	VARCHAR2(3)	Indicates whether there is a clean PDB restore point for this PDB (YES) or not (NO)
PDB_INCARNATION#	NUMBER	Pluggable database (PDB) incarnation number. This value is meaningful only for PDB restore points.

Column	Datatype	Description
REPLICATED	VARCHAR2(3)	This column is useful in Oracle Data Guard environments. It indicates the method by which a restore point was created. Possible values: • YES - The restore point was automatically replicated from the primary database to this database when this database was a standby database. The string _PRIMARY is appended to the name of such a restore point. • NO - The restore point was created by a user and was not replicated from the primary database.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.176 V$RESULT_CACHE_DEPENDENCY

V$RESULT_CACHE_DEPENDENCY displays the depends-on relationship between cached results and dependencies.

Column	Datatype	Description
RESULT_ID	NUMBER	Cached result
DEPEND_ID	NUMBER	Dependency object
OBJECT_NO	NUMBER	Dictionary object number of the dependency object
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.177 V$RESULT_CACHE_MEMORY

V$RESULT_CACHE_MEMORY displays all the memory blocks and their status.

Column	Datatype	Description
ID	NUMBER	Unique block identifier (that is, the block number)
CHUNK	NUMBER	Chunk to which the block belongs, relative to the subcache in which the memory object resides (the upper 27 bits of the ID)
OFFSET	NUMBER	Offset of the block within its chunk, relative to the subcache in which the memory object resides (the lower 5 bits of the ID)
SUBCACHE_ID	NUMBER	Subcache ID
FREE	VARCHAR2(3)	Indicates whether the block is free (YES) or not (NO)

Column	Datatype	Description
OBJECT_ID	NUMBER	Cache object to which the memory block belongs; NULL if the memory block is not allocated to a cache object (FREE = YES)
POSITION	NUMBER	Position of the block in the cached object; NULL if the memory block is not allocated to a cache object (FREE = YES)
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.178 V$RESULT_CACHE_OBJECTS

V$RESULT_CACHE_OBJECTS displays all the objects (both cached results and dependencies) and their attributes.

Column	Datatype	Description
ID	NUMBER	Identifier for the cache object (also the ID of the first block)
TYPE	VARCHAR2(10)	Type of the cache object: • Dependency • Result • Temp
STATUS	VARCHAR2(9)	Status of the object: • New - Result is still under construction • Published - Result is available for use • Bypass - Result will be bypassed from use • Expired - Result has exceeded expiration time • Invalid - Result is no longer available for use
BUCKET_NO	NUMBER	Internal hash bucket for the object
HASH	NUMBER	Hash value for the object
NAME	VARCHAR2(387)	Name (for example, SQL prefix or PL/SQL function name)
NAMESPACE	VARCHAR2(10)	Namespace: • SQL • PLSQL • KEY VECTOR
CREATION_TIMESTAMP	DATE	Time when the object was created
CREATOR_UID	NUMBER	UID that created the object
DEPEND_COUNT	NUMBER	Number of dependencies (TYPE = Result) or dependents (TYPE = Dependency)
BLOCK_COUNT	NUMBER	Total number of blocks in the cached object
SCN	NUMBER	Build SCN (TYPE = Result) or invalidation SCN (TYPE = Dependency)
COLUMN_COUNT[1]	NUMBER	Number of columns in the cached result
PIN_COUNT[1]	NUMBER	Number of active scans on this result

Column	Datatype	Description
SCAN_COUNT[1]	NUMBER	Total number of scans initiated on the cached result
ROW_COUNT[1]	NUMBER	Total number of rows in the cached result
ROW_SIZE_MAX[1]	NUMBER	Size of the largest row (in bytes)
ROW_SIZE_MIN[1]	NUMBER	Size of the smallest row (in bytes)
ROW_SIZE_AVG[1]	NUMBER	Average size of a row (in bytes)
BUILD_TIME[1]	NUMBER	Amount of time (in hundredths of a second) it took to build the cached result
LRU_NUMBER[1]	NUMBER	LRU list position (the larger the value, the more recent the usage)
OBJECT_NO[2]	NUMBER	Dictionary object number of the dependency object
INVALIDATIONS[2]	NUMBER	Number of times the object has invalidated its dependents
TIME_SAVED[2]	NUMBER	Amount of time (in microseconds) saved by reuse
INVALIDATION_COST[2]	NUMBER	Amount of time (in microseconds) spent on invalidation
SPACE_OVERHEAD[1]	NUMBER	Overhead (in bytes) for the result
SPACE_UNUSED[1]	NUMBER	Unused space (in bytes) for the result
CACHE_ID	VARCHAR2(387)	`CacheId` for the result (object name if it's a dependency)
CACHE_KEY	VARCHAR2(387)	`CacheKey` for the result (object name if it's a dependency)
CHECKSUM	NUMBER	Checksum for the result object. The checksum is computed over all the blocks in the result cache object minus the object header.
EDITION_ID	NUMBER	Shows the edition's object ID that was in use when the result was calculated
DB_LINK	VARCHAR2(3)	Possible values: • `YES`: The result cache object references a remote database object • `NO`: The result cache object does not reference a remote database object
GLOBAL	VARCHAR2(3)	Possible values: • `YES`: The object was fetched from the result cache of a remote database instance • `NO`: The object was recomputed on this instance, either because it was not available on a remote database instance or because the system deemed it was inefficient to fetch the object from a remote database instance
SUBCACHE_ID	NUMBER	Subcache ID
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • `0`: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • `1`: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

1 This column is valid only for `TYPE` = `Result`; otherwise, its value is 0.

2 This column is valid only for `TYPE` = `Dependency`; otherwise, its value is 0.

9.179 V$RESULT_CACHE_STATISTICS

V$RESULT_CACHE_STATISTICS displays various result cache settings and usage statistics.

Column	Datatype	Description
ID	NUMBER	Statistic number
NAME	VARCHAR2(128)	Name of the statistic (see Table 9-6)
VALUE	VARCHAR2(81)	Value of the statistic
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

Table 9-6 V$RESULT_CACHE_STATISTICS and V$RESULT_SUBCACHE_STATISTICS Statistics

Statistic Name	Description
Block Count Current	Number of memory blocks currently allocated
Block Count Maximum	Maximum number of memory blocks allowed
Block Size (Bytes)	Size of each memory block
Create Count Failure	Number of cache results that failed to create
Create Count Success	Number of cache results successfully created
Delete Count Invalid	Number of invalid cached results deleted
Delete Count Valid	Number of valid cached results deleted
Find Copy Count	Number of results copied directly out of the cache
Find Count	Number of cached results that were successfully found
Global Load Rate	Rate at which cached results were fetched from remote instances (in bytes per 10 milliseconds)
Global Prune By Self Count	Number of times this instance did not provide a cached result to a remote instance because this instance deemed it inefficient to do so
Global Prune Count	Number of times this instance did not fetch a cached result from a remote instance because it was deemed inefficient to do so
Hash Chain Length	Average length of items in the hash chain
Invalidation Count	Total number of invalidations
Result Size Maximum (Blocks)	Maximum number of blocks allowed for a single result
Subcache Count	Number of subcaches in the result cache

See Also:

"V$RESULT_SUBCACHE_STATISTICS"

9.180 V$RESULT_SUBCACHE_STATISTICS

V$RESULT_SUBCACHE_STATISTICS displays various result cache settings and usage statistics on a per subcache basis.

Column	Datatype	Description
ID	NUMBER	Statistic number
NAME	VARCHAR2(128)	Name of the statistic This view displays the same statistics as the V$RESULT_CACHE_STATISTICS view. See Table 9-6 in the documentation for V$RESULT_CACHE_STATISTICS.
VALUE	VARCHAR2(81)	Value of the statistic
SUBCACHE_ID	NUMBER	Subcache ID
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

See Also:

"V$RESULT_CACHE_STATISTICS"

9.181 V$RMAN_BACKUP_JOB_DETAILS

V$RMAN_BACKUP_JOB_DETAILS displays details about backup jobs.

Column	Datatype	Description
SESSION_KEY	NUMBER	Session identifier
SESSION_RECID	NUMBER	Together, with SESSION_KEY and SESSION_STAMP, used to uniquely identify job output from V$RMAN_OUTPUT
SESSION_STAMP	NUMBER	Together, with SESSION_KEY and SESSION_RECID, used to uniquely identify job output from V$RMAN_OUTPUT
COMMAND_ID	VARCHAR2(33)	Either a user-specified SET COMMAND ID or a unique command ID generated by RMAN
START_TIME	DATE	Start time of the first BACKUP command in the job
END_TIME	DATE	End time of the last BACKUP command in the job
INPUT_BYTES	NUMBER	Sum of all input file sizes backed up by this job
OUTPUT_BYTES	NUMBER	Output size of all pieces generated by this job
STATUS_WEIGHT	NUMBER	Used internally by Enterprise Manager
OPTIMIZED_WEIGHT	NUMBER	Used internally by Enterprise Manager

Column	Datatype	Description
OBJECT_TYPE_WEIGHT	NUMBER	Used internally by Enterprise Manager
OUTPUT_DEVICE_TYPE	VARCHAR2(17)	Can be DISK, SBT, or *. An * indicates more than one device (in most cases, it will be DISK or SBT).
AUTOBACKUP_COUNT	NUMBER	Number of autobackups performed by this job
BACKED_BY_OSB	VARCHAR2(3)	A value of YES means the backup was done to Oracle Secure Backup. Otherwise, backed up by other third party tape library.
AUTOBACKUP_DONE	VARCHAR2(3)	YES or NO, depending upon whether or not a control file autobackup was done as part of this backup job
STATUS	VARCHAR2(23)	One of the following values: • RUNNING WITH WARNINGS • RUNNING WITH ERRORS • COMPLETED • COMPLETED WITH WARNINGS • COMPLETED WITH ERRORS • FAILED
INPUT_TYPE	VARCHAR2(13)	Contains one of the following values. If the user command does not satisfy one of them, then preference is given in order, from top to bottom of the list. • DB FULL • RECVR AREA • DB INCR • DATAFILE FULL • DATAFILE INCR • ARCHIVELOG • CONTROLFILE • SPFILE
OPTIMIZED	VARCHAR2(3)	YES or NO, depending on whether optimization was applied. Applicable to backup jobs only.
ELAPSED_SECONDS	NUMBER	Number of elapsed seconds
COMPRESSION_RATIO	NUMBER	Compression ratio
INPUT_BYTES_PER_SEC	NUMBER	Input read-rate-per-second
OUTPUT_BYTES_PER_SEC	NUMBER	Output write-rate-per-second
INPUT_BYTES_DISPLAY	VARCHAR2(4000)	Values in user-displayable form. They will be converted to a format of nM, nG, nT, nP, and so on.
OUTPUT_BYTES_DISPLAY	VARCHAR2(4000)	Values in user-displayable form. They will be converted to a format of nM, nG, nT, nP, and so on
INPUT_BYTES_PER_SEC_DISPLAY	VARCHAR2(4000)	Input read-rate-per-second. These values are in user-displayable form. They will be converted to a format of nM, nG, nT, nP, and so on.
OUTPUT_BYTES_PER_SEC_DISPLAY	VARCHAR2(4000)	Output write-rate-per-second. These values are in user-displayable form. They will be converted to a format of nM, nG, nT, nP, and so on.
TIME_TAKEN_DISPLAY	VARCHAR2(4000)	Time taken, shown in user-displayable format <nn>h:<nn>m:<nn>s

Column	Datatype	Description
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.182 V$RMAN_BACKUP_SUBJOB_DETAILS

V$RMAN_BACKUP_SUBJOB_DETAILS merges similar operations within an RMAN session into a single row. For example, if there are four BACKUP DATAFILE <n> commands, three RECOVERY COPY OF DATAFILE commands, and one BACKUP RECOVERY AREA command, this view will contain three rows - one each for BACKUP, ROLLFORWARD, and COPY_DISK_TO_TAPE operation.

Column	Datatype	Description
SESSION_KEY	NUMBER	Session identifier
SESSION_RECID	NUMBER	Together with SESSION_KEY and SESSION_STAMP, used to uniquely identify job output from V$RMAN_OUTPUT
SESSION_STAMP	NUMBER	Together with SESSION_KEY and SESSION_RECID, used to uniquely identify job output from V$RMAN_OUTPUT
OPERATION	VARCHAR2(33)	Can be BACKUP, ROLLFORWARD, VALIDATE, or COPY_DISK_TO_TAPE. A row for each suboperation type for the session will be in the output view.
COMMAND_ID	VARCHAR2(33)	Either a user-specified SET COMMAND ID or a unique command ID generated by RMAN
START_TIME	DATE	Start time of the first BACKUP command in the job
END_TIME	DATE	End time of the last BACKUP command in the job
INPUT_BYTES	NUMBER	Sum of all input file sizes backed up by this job
OUTPUT_BYTES	NUMBER	Output size of all pieces generated by this job
STATUS_WEIGHT	NUMBER	Used internally by Enterprise Manager
OBJECT_TYPE_WEIGHT	NUMBER	Used internally by Enterprise Manager
OPTIMIZED_WEIGHT	NUMBER	Used internally by Enterprise Manager
OUTPUT_DEVICE_TYPE	VARCHAR2(17)	Can be DISK, SBT, or *. An * indicates more than one device (in most cases, it will be DISK or SBT).
BACKED_BY_OSB	VARCHAR2(3)	A value of YES means the backup was done to Oracle Secure Backup. Otherwise, backed up by other third party tape library.
AUTOBACKUP_DONE	VARCHAR2(3)	YES or NO, depending upon whether or not a control file autobackup was done as part of this job

Column	Datatype	Description
STATUS	VARCHAR2(23)	One of the following values: • RUNNING WITH WARNINGS • RUNNING WITH ERRORS • COMPLETED • COMPLETED WITH WARNINGS • COMPLETED WITH ERRORS • FAILED
INPUT_TYPE	VARCHAR2(13)	Contains one of the following values. If the user command does not satisfy one of them, then preference is given in order, from top to bottom of the list. • DB FULL • RECVR AREA • DB INCR • DATAFILE FULL • DATAFILE INCR • ARCHIVELOG • CONTROLFILE • SPFILE
OPTIMIZED	VARCHAR2(3)	YES or NO, depending on whether optimization was applied. Applicable to backup jobs only.
AUTOBACKUP_COUNT	NUMBER	Number of autobackups performed by this job
COMPRESSION_RATIO	NUMBER	Compression ratio
INPUT_BYTES_DISPLAY	VARCHAR2(4000)	Values in user-displayable form. They will be converted to a format of nM, nG, nT, nP, and so on.
OUTPUT_BYTES_DISPLAY	VARCHAR2(4000)	Values in user-displayable form. They will be converted to a format of nM, nG, nT, nP, and so on.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.183 V$RMAN_BACKUP_TYPE

V$RMAN_BACKUP_TYPE displays information about RMAN backup types.

Column	Datatype	Description
WEIGHT	NUMBER	Used to set precedence order of different backup types in reports.
INPUT_TYPE	VARCHAR2(13)	Used to represent possible filters used in creating various reporting screens.

Column	Datatype	Description
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.184 V$RMAN_COMPRESSION_ALGORITHM

V$RMAN_COMPRESSION_ALGORITHM provides descriptions of supported compression algorithms. It is used by the RMAN client.

Column	Datatype	Description
ALGORITHM_ID	NUMBER	Algorithm ID
ALGORITHM_NAME	VARCHAR2(64)	Name of the algorithm (for example, LOW, MEDIUM, DEFAULT, or HIGH)
INITIAL_RELEASE	VARCHAR2(18)	First Oracle Database release when this compression algorithm was available
TERMINAL_RELEASE	VARCHAR2(18)	Last Oracle Database release that supported using this compression algorithm to create new backups. Existing backups can always be restored, even if they use a deprecated compression algorithm.
ALGORITHM_DESCRIPTION	VARCHAR2(64)	Description of the algorithm
ALGORITHM_COMPATIBILITY	VARCHAR2(18)	Required database compatibility level for the algorithm (for example, 11.2.0 for DEFAULT)
IS_VALID	VARCHAR2(3)	Indicates whether the algorithm is valid with regard to the compatibility setting (YES) or not (NO). The value is YES if ALGORITHM_COMPATIBILITY <= DATABASE_COMPATIBILITY.
REQUIRES_ACO	VARCHAR2(3)	Indicates whether the algorithm requires the Advanced Compression Option (YES) or not (NO)
IS_DEFAULT	VARCHAR2(3)	Indicates whether the algorithm is the default compression algorithm that RMAN uses to create compressed backup sets (YES) or not (NO)
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.185 V$RMAN_CONFIGURATION

V$RMAN_CONFIGURATION lists information about RMAN persistent configuration settings.

Column	Datatype	Description
CONF#	NUMBER	A unique key identifying this configuration record within the target database that owns it.
NAME	VARCHAR2(65)	The type of configuration. All options of the CONFIGURE command are valid types except: • CONFIGURE EXCLUDE (described in RC_TABLESPACE) • CONFIGURE AUXNAME (described in RC_DATAFILE) • CONFIGURE SNAPSHOT CONTROLFILE (stored only in control file)
VALUE	VARCHAR2(1025)	The CONFIGURE command setting. Example: RETENTION POLICY TO RECOVERY WINDOW OF 10 DAYS
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.186 V$RMAN_ENCRYPTION_ALGORITHMS

V$RMAN_ENCRYPTION_ALGORITHMS displays supported encryption algorithms. It is used by the RMAN client to validate user-requested algorithms.

This view lists AES128, AES192, and AES256 encryption algorithms for the current release. The default algorithm is AES256.

Column	Datatype	Description
ALGORITHM_ID	NUMBER	Number to identify the algorithm
ALGORITHM_NAME	VARCHAR2(64)	Name of the algorithm (for example, AES128, AES192, or AES256)
ALGORITHM_DESCRIPTION	VARCHAR2(64)	Description of the algorithm
IS_DEFAULT	VARCHAR2(3)	Indicates whether this is the default encryption algorithm for CFB encryption mode (YES) or not (NO). CFB encryption mode is used when the COMPATIBLE initialization parameter is set to a value less than 23.0.0.0. Otherwise, refer to the IS_DEFAULTXTS column. This value is set by Oracle Database and may vary for each Release (that is, it is not dependent on user-specified RMAN configuration).
RESTORE_ONLY	VARCHAR2(3)	Indicates whether this algorithm can be used for restore only (YES) or not (NO). If the value is NO, then the algorithm is also allowed for backup. This column is useful in determining whether an encryption algorithm is deprecated for backup purpose.
IS_DEFAULTXTS	VARCHAR2(3)	Indicates whether this is the default encryption algorithm for XTS encryption mode (YES) or not (NO). XTS encryption mode is used when the COMPATIBLE initialization parameter is set to a value greater than or equal to 23.0.0.0. Otherwise, refer to the IS_DEFAULT column. This value is set by Oracle Database and may vary for each Release (that is, it is not dependent on user-specified RMAN configuration).

Column	Datatype	Description
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.187 V$RMAN_OUTPUT

V$RMAN_OUTPUT displays messages reported by RMAN.

This is an in-memory view and is not recorded in the controlfile. The view can hold 32768 rows.

Column	Datatype	Description
SID	NUMBER	Session ID of the session which is running this RMAN operation
RECID	NUMBER	Record ID of the corresponding V$RMAN_STATUS row
STAMP	NUMBER	Timestamp of the corresponding V$RMAN_STATUS row
SESSION_RECID	NUMBER	Record ID of the session (corresponding V$RMAN_STATUS row with ROW_LEVEL = 0)
SESSION_STAMP	NUMBER	Timestamp of the session (corresponding V$RMAN_STATUS row with ROW_LEVEL = 0)
OUTPUT	VARCHAR2(130)	Output text reported by RMAN
RMAN_STATUS_RECID	NUMBER	Owning V$RMAN_STATUS record ID
RMAN_STATUS_STAMP	NUMBER	Owning V$RMAN_STATUS record stamp
SESSION_KEY	NUMBER	Session identifier
GUID	NUMBER	The guid of the pluggable database (PDB) that did the RMAN operation. V$RMAN_OUTPUT captures the rman output. If a PDB sysdba did the backup, the guid of the PDB is displayed in V$RMAN_OUTPUT. If root did the backup, then guid of the root is displayed in V$RMAN_OUTPUT. Root A root user can see all of V$RMAN_OUTPUT rows (that is, the rows owned by all PDBs), but a PDB user can only see that PDB's V$RMAN_OUTPUT rows (that is, no rows from root or other PDBs).
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.188 V$RMAN_STATUS

V$RMAN_STATUS displays the finished and on-going RMAN jobs. For on-going jobs, this view displays progress and status. The jobs which are in progress are stored only in memory while the finished jobs are stored in the controlfile.

Column	Datatype	Description
SID	NUMBER	Session ID of the session which is running this RMAN operation
RECID	NUMBER	Record ID of the row in the controlfile
STAMP	NUMBER	Timestamp of the row (RECID + STAMP is unique)
PARENT_RECID	NUMBER	Record ID of the parent row of this row (corresponding V$RMAN_STATUS row with ROW_LEVEL = ROW_LEVEL - 1)
PARENT_STAMP	NUMBER	Timestamp of the parent row of this row (corresponding V$RMAN_STATUS row with ROW_LEVEL = ROW_LEVEL - 1)
SESSION_RECID	NUMBER	Record ID of the session (corresponding V$RMAN_STATUS row with ROW_LEVEL = 0)
SESSION_STAMP	NUMBER	Timestamp of the session (corresponding V$RMAN_STATUS row with ROW_LEVEL = 0)
ROW_LEVEL	NUMBER	Level of the row. The session has level 0.
ROW_TYPE	VARCHAR2(19)	Type of the row: • SESSION • COMMAND • RECURSIVE OPERATION
COMMAND_ID	VARCHAR2(33)	Command ID set by the RMAN SET COMMAND ID command. If not set, then RMAN will create a unique number.
OPERATION	VARCHAR2(33)	Name of the command in the execution explained by this row
STATUS	VARCHAR2(23)	Status of the operation: • RUNNING • RUNNING WITH WARNINGS • RUNNING WITH ERRORS • COMPLETED • COMPLETED WITH WARNINGS • COMPLETED WITH ERRORS • FAILED
MBYTES_PROCESSED	NUMBER	Percentage of the job completed; null if not applicable for the operation
START_TIME	DATE	Start time of the job
END_TIME	DATE	End time of the job
INPUT_BYTES	NUMBER	Number of input bytes read
OUTPUT_BYTES	NUMBER	Number of output bytes written
OPTIMIZED	VARCHAR2(3)	YES, if backup optimization was applied during the backup job. Otherwise, NO.
OBJECT_TYPE	VARCHAR2(13)	Identifies types of objects backed up
OUTPUT_DEVICE_TYPE	VARCHAR2(17)	DISK, SBT_TAPE, or *. An * indicates that output was written to more than one device type.
OSB_ALLOCATED	VARCHAR2(3)	A value of YES means an Oracle Secure Backup channel was allocated during the specified operation identified by the V$RMAN_STATUS view.

Column	Datatype	Description
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.189 V$RO_USER_ACCOUNT

V$RO_USER_ACCOUNT is populated only on Oracle databases that are open in read-only mode. When a database is read-only, security data cannot be stored in normal catalogue tables. Instead, the security data is stored in an in-memory table that is queried through this view. In an Oracle Data Guard environment, some of the security information for user accounts on the standby is inherited from the primary server. For example, if the account is locked out unlimited on the primary, then it will be locked on the standby database(s). The information stored on the standby is volatile information that user actions on the standby database(s) can affect, such as the number of failed logins, and the time the account was locked on the standby due to failed access attempts. Note that failed login attempts on standbys do not affect the account status on primaries.

If this view is queried from the root in a multitenent container database (CDB), then only common users and the SYS user are returned.

If this view is queried from a pluggable database (PDB), only rows that pertain to the current PDB are returned.

Column	Datatype	Description
USERID	NUMBER	User ID number
PASSW_EXPIRED	NUMBER	Indicates whether the password has expired (1) or not (0)
PASSW_IN_GRACE	NUMBER	Indicates whether the account is in grace (1) or not (0)
PASSW_LOCKED	NUMBER	Indicates whether the account is locked (1) or not (0)
PASSW_LOCK_UNLIM	NUMBER	Indicates whether the account is locked for an unlimited time (1) or not (0)
FAILED_LOGINS	NUMBER	The number of failed login attempts. The count is not cumulative; it is reset upon successful logon to the account
EXPIRATION_AFTER_GRACE	TIMESTAMP(3)	The expiration time after grace
PASSW_LOCK_TIME	TIMESTAMP(3)	The time the account was locked out

Column	Datatype	Description
CON_ID	NUMBER	The ID of the container where the failed login occurred. For users that are not common users, the CON_IDis the PDB ID where the failed login attempt occurred. For common users, the CON_ID is 0. The login attempts that occurred on a PDB are not displayed when you query V$RO_USER_ACCOUNT from another PDB. You only see the failed login attempts of any users (that are not common users) if those failed login attempts occurred on the same PDB from which you are querying V$RO_USER_ACCOUNT. The failed login attempts of common users (and of the SYS user) are only displayed when V$RO_USER_ACCOUNT is queried from the root of a CDB, not when it is queried from a PDB. In a non-CDB, the value is always 0.
USERNAME	VARCHAR2(128)	User name

9.190 V$ROLLNAME

V$ROLLNAME lists the names of all online rollback segments. It can only be accessed when the database is open.

Column	Datatype	NULL	Description
USN	NUMBER		Rollback (undo) segment number
NAME	VARCHAR2(30)	NOT NULL	Rollback segment name
CON_ID	NUMBER		The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.191 V$ROLLSTAT

V$ROLLSTAT contains rollback segment statistics.

Column	Datatype	Description
USN	NUMBER	Rollback segment number
LATCH	NUMBER	Latch for the rollback segment
EXTENTS	NUMBER	Number of extents in the rollback segment
RSSIZE	NUMBER	Size (in bytes) of the rollback segment. This value differs by the number of bytes in one database block from the value of the BYTES column of the *_SEGMENTS view. **See Also:** *Oracle Database Administrator's Guide* for more information about space management for rollback segments
WRITES	NUMBER	Number of bytes written to the rollback segment

Column	Datatype	Description
XACTS	NUMBER	Number of active transactions
GETS	NUMBER	Number of header gets
WAITS	NUMBER	Number of header waits
OPTSIZE	NUMBER	Optimal size of the rollback segment
HWMSIZE	NUMBER	High watermark of the rollback segment size
SHRINKS	NUMBER	Number of times the size of a rollback segment decreases
WRAPS	NUMBER	Number of times rollback segment is wrapped
EXTENDS	NUMBER	Number of times rollback segment size is extended
AVESHRINK	NUMBER	Average shrink size
AVEACTIVE	NUMBER	Current size of active extents, averaged over time.
STATUS	VARCHAR2(15)	Rollback segment status: • ONLINE • PENDING OFFLINE • OFFLINE • FULL
CUREXT	NUMBER	Current extent
CURBLK	NUMBER	Current block
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.192 V$ROWCACHE

V$ROWCACHE displays statistics for data dictionary activity. Each row contains statistics for one data dictionary cache.

Column	Datatype	Description
CACHE#	NUMBER	Row cache ID number
TYPE	VARCHAR2(11)	Parent or subordinate row cache type
SUBORDINATE#	NUMBER	Subordinate set number
PARAMETER	VARCHAR2(32)	Name of the initialization parameter that determines the number of entries in the data dictionary cache
COUNT	NUMBER	Total number of entries in the cache
USAGE	NUMBER	Number of cache entries that contain valid data
FIXED	NUMBER	Number of fixed entries in the cache
GETS	NUMBER	Total number of requests for information on the data object
FASTGETS	NUMBER	Reserved for internal use
GETMISSES	NUMBER	Number of data requests resulting in cache misses
SCANS	NUMBER	Number of scan requests

Column	Datatype	Description
SCANMISSES	NUMBER	Number of times a scan failed to find the data in the cache
SCANCOMPLETES	NUMBER	For a list of subordinate entries, the number of times the list was scanned completely
MODIFICATIONS	NUMBER	Number of inserts, updates, and deletions
FLUSHES	NUMBER	Number of times flushed to disk
DLM_REQUESTS	NUMBER	Number of DLM requests
DLM_CONFLICTS	NUMBER	Number of DLM conflicts This column is obsolete and is maintained for backward compatibility.
DLM_RELEASES	NUMBER	Number of DLM releases This column is obsolete and is maintained for backward compatibility.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.193 V$ROWCACHE_PARENT

V$ROWCACHE_PARENT displays information for parent objects in the data dictionary. There is one row per lock owner, and one waiter for each object. This row shows the mode held or requested. For objects with no owners or waiters, a single row is displayed.

Column	Datatype	Description
INDX	NUMBER	Index of the row
HASH	NUMBER	Hash value
ADDRESS	RAW(4 | 8)	Address of the parent object
CACHE#	NUMBER	Parent cache ID
CACHE_NAME	VARCHAR2(64)	Parent cache name
EXISTENT	VARCHAR2(1)	Indicates whether the object is an existing object
LOCK_MODE	NUMBER	Mode the lock is held in
LOCK_REQUEST	NUMBER	Mode the lock is requested in
TXN	RAW(4 | 8)	Transaction currently locking the object
SADDR	RAW(4 | 8)	Address of the session
INST_LOCK_REQUEST	NUMBER	Mode in which instance lock is being requested. This column is only relevant for Real Application Clusters.
INST_LOCK_RELEASE	NUMBER	Whether the instance lock needs to be released. This column is only relevant for Real Application Clusters.
INST_LOCK_TYPE	VARCHAR2(2)	Type of instance lock. This column is only relevant for Real Application Clusters.
INST_LOCK_ID1	RAW(4)	ID associated with the instance lock. This column is only relevant for Real Application Clusters.

Column	Datatype	Description
INST_LOCK_ID2	RAW(4)	ID associated with the instance lock. This column is only relevant for Real Application Clusters.
KEY	RAW(100)	Contents of the key. This column is only relevant for Real Application Clusters.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.194 V$ROWCACHE_SUBORDINATE

V$ROWCACHE_SUBORDINATE displays information for subordinate objects in the data dictionary.

Column	Datatype	Description
INDX	NUMBER	The index
HASH	NUMBER	The hash value
ADDRESS	RAW(4 | 8)	Address of the subordinate object
CACHE#	NUMBER	The parent cache ID
SUBCACHE#	NUMBER	The subcache ID
SUBCACHE_NAME	VARCHAR2(64)	The subcache name
EXISTENT	VARCHAR2(1)	Whether the object is an existing object
PARENT	RAW(4 | 8)	Address of the parent object
KEY	RAW(100)	The contents of the key
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.195 V$RSRC_CONS_GROUP_HISTORY

V$RSRC_CONS_GROUP_HISTORY displays a history of consumer group statistics for each entry in V$RSRC_PLAN_HISTORY that has a non-NULL plan.

When the STATISTICS_LEVEL is set to TYPICAL or ALL, this view contains information about CPU utilization and wait times even when no Resource Manager plan is set or when the Resource Manager plan does not monitor CPU or session resources.

A new window is created in V$RSRC_CON_GROUP_HISTORY when a pluggable database (PDB) changes its resource plan. The plan windows inside a PDB are not impacted by a multitenant container database (CDB) resource plan change.

Since PDB plans can be set independently across different PDBs, V$RSRC_CON_GROUP_HISTORY will not cover the same time period across different PDBs. Therefore, this view is not useful for comparing statistics across different PDBs.

Column	Datatype	Description
SEQUENCE#	NUMBER	A sequential counter that uniquely describes the V$RSRC_PLAN_HISTORY entry to which these consumer group statistics apply. When the instance is restarted, this value is reset to zero.
ID	NUMBER	Consumer group object ID (a unique number, consistent across database shutdowns and startups)
NAME	VARCHAR2(30)	Name of the consumer group
REQUESTS	NUMBER	Cumulative number of requests that were executed in the consumer group
CPU_WAIT_TIME	NUMBER	Cumulative amount of time that sessions waited for CPU on the resmgr: cpu quantum wait event because of resource management (in milliseconds). This does not include waits due to latch or enqueue contention, I/O waits, and so on. When CPU resources are not being actively managed, this value is set to zero.
CPU_WAITS	NUMBER	Cumulative number of times all sessions in the consumer group had to wait for CPU on the resmgr: cpu quantum wait event because of resource management. This does not include waits due to latch or enqueue contention, I/O waits, and so on. When CPU resources are not being actively managed, this value is set to zero.
CONSUMED_CPU_TIME	NUMBER	Cumulative amount of CPU time consumed by all sessions in the consumer group (in milliseconds)
YIELDS	NUMBER	Cumulative number of times that sessions in the consumer group had to yield CPU to other sessions because of quantum expiration. When CPU resources are not being actively managed, this value is set to zero.
CPU_DECISIONS	NUMBER	Percentage of CPU decisions for which the consumer group was present. When CPU resources are not being actively managed, this value is set to zero.
CPU_DECISIONS_EXCLUSIVE	NUMBER	Percentage of the CPU decisions for which the consumer group was present and was the only consumer group present. When CPU resources are not being actively managed, this value is set to zero.
CPU_DECISIONS_WON	NUMBER	Percentage of the CPU decisions that the consumer group won. When CPU resources are not being actively managed, this value is set to zero.
ACTIVE_SESS_LIMIT_HIT	NUMBER	Number of times that sessions in the consumer group were queued because the consumer group reached its active session limit
UNDO_LIMIT_HIT	NUMBER	Number of times that queries in the consumer group were canceled because the consumer group reached its UNDO_POOL limit
SWITCHES_IN_CPU_TIME	NUMBER	Number of switches into the consumer group because of the Resource Manager plan's SWITCH_TIME limit
SWITCHES_OUT_CPU_TIME	NUMBER	Number of switches out of the consumer group because of the Resource Manager plan's SWITCH_TIME limit
SWITCHES_IN_IO_MEGABYTES	NUMBER	Number of switches into the consumer group because of the Resource Manager plan's SWITCH_IO_MEGABYTES limit
SWITCHES_OUT_IO_MEGABYTES	NUMBER	Number of switches out of the consumer group because of the Resource Manager plan's SWITCH_IO_MEGABYTES limit
SWITCHES_IN_IO_REQUESTS	NUMBER	Number of switches into the consumer group because of the Resource Manager plan's SWITCH_IO_REQS limit

Column	Datatype	Description
SWITCHES_OUT_IO_REQUESTS	NUMBER	Number of switches out of the consumer group because of the Resource Manager plan's SWITCH_IO_REQS limit
SWITCHES_IN_IO_LOGICAL	NUMBER	Number of switches into the consumer group because of the Resource Manager plan's SWITCH_IO_LOGICAL limit
SWITCHES_OUT_IO_LOGICAL	NUMBER	Number of switches out of the consumer group because of the Resource Manager plan's SWITCH_IO_LOGICAL limit
SWITCHES_IN_ELAPSED_TIME	NUMBER	Number of switches into the consumer group because of the Resource Manager plan's SWITCH_ELAPSED_TIME limit
SWITCHES_OUT_ELAPSED_TIME	NUMBER	Number of switches out of the consumer group because of the Resource Manager plan's SWITCH_ELAPSED_TIME limit
SQL_CANCELED	NUMBER	Number of times that SQL queries running in the consumer group were terminated because they exceeded one of the Resource Manager plan's SWITCH limits and CANCEL_SQL was specified as the Resource Manager plan's SWITCH_GROUP
ACTIVE_SESS_KILLED	NUMBER	Number of times that sessions running in the consumer group were terminated because they exceeded one of the Resource Manager plan's SWITCH limits and KILL_SESSION was specified as the Resource Manager plan's SWITCH_GROUP
IDLE_SESS_KILLED	NUMBER	Number of times that sessions in the consumer group were terminated because they were idle for too long (reached MAX_IDLE_TIME)
IDLE_BLKR_SESS_KILLED	NUMBER	Number of times that sessions in the consumer group were terminated because they were idle too long (reached MAX_IDLE_BLOCKER_TIME) and were blocking other sessions
QUEUED_TIME	NUMBER	Total amount of time that sessions in the consumer group have spent in the QUEUED state because of the active session limit (in milliseconds)
QUEUE_TIME_OUTS	NUMBER	Number of times that requests from sessions in the consumer group timed out because they were queued for too long (reached QUEUEING_P1)
IO_SERVICE_TIME	NUMBER	Cumulative I/O wait time (in milliseconds)
IO_SERVICE_WAITS	NUMBER	Total number of wait requests
SMALL_READ_MEGABYTES	NUMBER	Number of single block megabytes read
SMALL_WRITE_MEGABYTES	NUMBER	Number of single block megabytes written
LARGE_READ_MEGABYTES	NUMBER	Number of multiblock megabytes read
LARGE_WRITE_MEGABYTES	NUMBER	Number of multiblock megabytes written
SMALL_READ_REQUESTS	NUMBER	Number of single block read requests
SMALL_WRITE_REQUESTS	NUMBER	Number of single block write requests
LARGE_READ_REQUESTS	NUMBER	Number of multiblock read requests
LARGE_WRITE_REQUESTS	NUMBER	Number of multiblock write requests
PQS_COMPLETED	NUMBER	Total number of completed parallel statements in the consumer group
PQ_SERVERS_USED	NUMBER	Total number of parallel servers used by completed parallel statements in the consumer group
PQS_QUEUED	NUMBER	Number of times that sessions in the consumer group were queued when trying to run parallel statements
PQ_ACTIVE_TIME	NUMBER	Cumulative sum of the parallel active times for all completed parallel statements in the consumer group (in milliseconds)

Column	Datatype	Description
PQ_QUEUED_TIME	NUMBER	Total amount of time that sessions in the consumer group were queued when trying to run parallel statements (in milliseconds)
PQ_QUEUE_TIME_OUTS	NUMBER	Number of times that parallel statements from sessions in the consumer group timed out because their queue time exceeded the Resource Manager plan's PARALLEL_QUEUE_TIMEOUT limit
PGA_LIMIT_SESSIONS_KILLED	NUMBER	Number of times that sessions in the consumer group were terminated because their untunable PGA usage exceeded the SESSION_PGA_LIMIT limit
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

See Also:

- "V$RSRC_PDB_HISTORY"
- "V$RSRC_PLAN_HISTORY"
- "STATISTICS_LEVEL"

9.196 V$RSRC_CONSUMER_GROUP

V$RSRC_CONSUMER_GROUP displays data related to currently active resource consumer groups.

When the STATISTICS_LEVEL is set to TYPICAL or ALL, this view contains information about CPU utilization and wait times even when no Resource Manager plan is set or when the Resource Manager plan does not monitor CPU or session resources.

Statistics in V$RSRC_CONSUMER_GROUP are reset when a pluggable database (PDB) changes its resource plan. They are not impacted by multitenant container database (CDB) resource plan changes.

Since PDB plans can be set independently across different PDBs, V$RSRC_CONSUMER_GROUP will not cover the same time period across different PDBs. Therefore, this view is not useful for comparing statistics across different PDBs.

Column	Datatype	Description
ID	NUMBER	Consumer group object ID (a unique number, consistent across database shutdowns and startups)
NAME	VARCHAR2(32)	Name of the consumer group
ACTIVE_SESSIONS	NUMBER	Number of currently active sessions in the consumer group
EXECUTION_WAITERS	NUMBER	Number of currently active sessions waiting for an execution time slice in which they will be able to use CPU

Column	Datatype	Description
REQUESTS	NUMBER	Cumulative number of requests that were executed in the consumer group
CPU_WAIT_TIME	NUMBER	Cumulative amount of time that sessions waited for CPU on the resmgr: cpu quantum wait event because of resource management (in milliseconds). This does not include waits due to latch or enqueue contention, I/O waits, and so on. When CPU resources are not being actively managed, this value is set to zero.
CPU_WAITS	NUMBER	Cumulative number of times all sessions in the consumer group had to wait for CPU on the resmgr: cpu quantum wait event because of resource management. This does not include waits due to latch or enqueue contention, I/O waits, and so on. When CPU resources are not being actively managed, this value is set to zero.
CONSUMED_CPU_TIME	NUMBER	Cumulative amount of CPU time consumed by all sessions in the consumer group (in milliseconds)
YIELDS	NUMBER	Cumulative number of times that sessions in the consumer group had to yield CPU to other sessions because of quantum expiration. When CPU resources are not being actively managed, this value is set to zero.
CPU_DECISIONS	NUMBER	Percentage of CPU decisions for which the consumer group was present. When CPU resources are not being actively managed, this value is set to zero.
CPU_DECISIONS_EXCLUSIVE	NUMBER	Percentage of the CPU decisions for which the consumer group was present and was the only consumer group present. When CPU resources are not being actively managed, this value is set to zero.
CPU_DECISIONS_WON	NUMBER	Percentage of the CPU decisions that the consumer group won. When CPU resources are not being actively managed, this value is set to zero.
QUEUE_LENGTH	NUMBER	Number of sessions waiting in the queue
CURRENT_UNDO_CONSUMPTION	NUMBER	Current amount (in KB) of undo consumed by the consumer group
ACTIVE_SESSION_LIMIT_HIT	NUMBER	Number of times that sessions in the consumer group were queued because the consumer group reached its active session limit
UNDO_LIMIT_HIT	NUMBER	Number of times that queries in the consumer group were canceled because the consumer group reached its UNDO_POOL limit
SWITCHES_IN_CPU_TIME	NUMBER	Number of switches into the consumer group because of the Resource Manager plan's SWITCH_TIME limit
SWITCHES_OUT_CPU_TIME	NUMBER	Number of switches out of the consumer group because of the Resource Manager plan's SWITCH_TIME limit
SWITCHES_IN_IO_MEGABYTES	NUMBER	Number of switches into the consumer group because of the Resource Manager plan's SWITCH_IO_MEGABYTES limit
SWITCHES_OUT_IO_MEGABYTES	NUMBER	Number of switches out of the consumer group because of the Resource Manager plan's SWITCH_IO_MEGABYTES limit
SWITCHES_IN_IO_REQUESTS	NUMBER	Number of switches into the consumer group because of the Resource Manager plan's SWITCH_IO_REQS limit
SWITCHES_OUT_IO_REQUESTS	NUMBER	Number of switches out of the consumer group because of the Resource Manager plan's SWITCH_IO_REQS limit
SWITCHES_IN_IO_LOGICAL	NUMBER	Number of switches into the consumer group because of the Resource Manager plan's SWITCH_IO_LOGICAL limit
SWITCHES_OUT_IO_LOGICAL	NUMBER	Number of switches out of the consumer group because of the Resource Manager plan's SWITCH_IO_LOGICAL limit

Column	Datatype	Description
SWITCHES_IN_ELAPSED_TIME	NUMBER	Number of switches into the consumer group because of the Resource Manager plan's SWITCH_ELAPSED_TIME limit
SWITCHES_OUT_ELAPSED_TIME	NUMBER	Number of switches out of the consumer group because of the Resource Manager plan's SWITCH_ELAPSED_TIME limit
SQL_CANCELED	NUMBER	Number of times that SQL queries running in the consumer group were terminated because they exceeded one of the Resource Manager plan's SWITCH limits and CANCEL_SQL was specified as the Resource Manager plan's SWITCH_GROUP
ACTIVE_SESSIONS_KILLED	NUMBER	Number of times that sessions running in the consumer group were terminated because they exceeded one of the Resource Manager plan's SWITCH limits and KILL_SESSION was specified as the Resource Manager plan's SWITCH_GROUP
IDLE_SESSIONS_KILLED	NUMBER	Number of times that sessions in the consumer group were terminated because they were idle for too long (reached MAX_IDLE_TIME)
IDLE_BLKR_SESSIONS_KILLED	NUMBER	Number of times that sessions in the consumer group were terminated because they were idle too long (reached MAX_IDLE_BLOCKER_TIME) and were blocking other sessions
QUEUED_TIME	NUMBER	Total amount of time that sessions in the consumer group have spent in the QUEUED state because of the active session limit (in milliseconds)
QUEUE_TIME_OUTS	NUMBER	Number of times that requests from sessions in the consumer group timed out because they were queued for too long (reached QUEUEING_P1)
IO_SERVICE_TIME	NUMBER	Cumulative I/O wait time (in milliseconds)
IO_SERVICE_WAITS	NUMBER	Total number of wait requests
SMALL_READ_MEGABYTES	NUMBER	Number of single block megabytes read
SMALL_WRITE_MEGABYTES	NUMBER	Number of single block megabytes written
LARGE_READ_MEGABYTES	NUMBER	Number of multiblock megabytes read
LARGE_WRITE_MEGABYTES	NUMBER	Number of multiblock megabytes written
SMALL_READ_REQUESTS	NUMBER	Number of single block read requests
SMALL_WRITE_REQUESTS	NUMBER	Number of single block write requests
LARGE_READ_REQUESTS	NUMBER	Number of multiblock read requests
LARGE_WRITE_REQUESTS	NUMBER	Number of multiblock write requests
CURRENT_PQS_ACTIVE	NUMBER	Number of active parallel statements in the consumer group. This value does not include parallel statements that are never queued, such as GV$ queries.
CURRENT_PQ_SERVERS_ACTIVE	NUMBER	Number of active parallel servers in the consumer group. This value does not include servers running parallel statements that are never queued, such as GV$ queries.
PQS_QUEUED	NUMBER	Number of times that sessions in the consumer group were queued when trying to run parallel statements
PQS_COMPLETED	NUMBER	Total number of completed parallel statements in the consumer group
PQ_SERVERS_USED	NUMBER	Total number of parallel servers used by completed parallel statements in the consumer group
PQ_ACTIVE_TIME	NUMBER	Cumulative sum of the parallel active times for all completed parallel statements in the consumer group (in milliseconds)
CURRENT_PQS_QUEUED	NUMBER	Number of sessions in the consumer group that are waiting in the parallel statement queue trying to run parallel statements

Column	Datatype	Description
PQ_QUEUED_TIME	NUMBER	Total amount of time that sessions in the consumer group were queued when trying to run parallel statements (in milliseconds)
PQ_QUEUE_TIME_OUTS	NUMBER	Number of times that parallel statements from sessions in the consumer group timed out because their queue time exceeded the Resource Manager plan's PARALLEL_QUEUE_TIMEOUT limit
PGA_LIMIT_SESSIONS_KILLED	NUMBER	Number of times that sessions in the consumer group were terminated because their untunable PGA usage exceeded the SESSION_PGA_LIMIT limit
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

See Also:

- "STATISTICS_LEVEL"
- "V$RSRC_PDB"
- *Oracle Database Administrator's Guide* for information on resource groups
- *Oracle Database PL/SQL Packages and Types Reference* for information on creating resource groups with the DBMS_RESOURCE_MANAGER package

9.197 V$RSRC_CONSUMER_GROUP_CPU_MTH

V$RSRC_CONSUMER_GROUP_CPU_MTH displays all resource allocation methods defined for resource consumer groups.

Column	Datatype	Description
NAME	VARCHAR2(40)	Name of the CPU resource allocation method
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

See Also:

- *Oracle Database Administrator's Guide* for information on resource allocation methods
- *Oracle Database PL/SQL Packages and Types Reference* on defining resource allocation methods for consumer groups with the `DBMS_RESOURCE_MANAGER` package
- "V$RSRC_PLAN_CPU_MTH" for a listing of all resource allocation methods defined for resource plans

9.198 V$RSRC_PDB

`V$RSRC_PDB` displays data related to currently active resource consumer groups by pluggable database (PDB).

When the `STATISTICS_LEVEL` is set to `TYPICAL` or `ALL`, this view contains information about CPU utilization and wait times even when no Resource Manager plan is set or when the Resource Manager plan does not monitor CPU or session resources.

Statistics in `V$RSRC_PDB` are reset when a multitenant container database (CDB) changes its resource plan. They are not impacted by PDB resource plan changes.

`V$RSRC_PDB` covers the same time period for all PDBs. This view is specifically designed for comparing statistics across different PDBs.

Since `V$RSRC_PDB` does not contain information at the consumer group level, it is not useful for comparing consumer groups inside a PDB.

Column	Datatype	Description
PDB_NAME	VARCHAR2(32)	PDB name
CPU_WAIT_TIME	NUMBER	Cumulative amount of time that sessions waited for CPU because of resource management. This does not include waits due to latch or enqueue contention, I/O waits, and so on. When CPU resources are not being actively managed, this value is set to zero.
CPU_WAITS	NUMBER	Cumulative number of times all sessions in the consumer group had to wait for CPU because of resource management. This does not include waits due to latch or enqueue contention, I/O waits, and so on. When CPU resources are not being actively managed, this value is set to zero.
CONSUMED_CPU_TIME	NUMBER	Cumulative amount of CPU time consumed by all sessions in the consumer group (in milliseconds)
YIELDS	NUMBER	Cumulative number of times that sessions in the consumer group had to yield CPU to other sessions because of quantum expiration. When CPU resources are not being actively managed, this value is set to zero.
IO_SERVICE_TIME	NUMBER	Cumulative I/O wait time (in milliseconds)
IO_SERVICE_WAITS	NUMBER	Total number of wait requests
SMALL_READ_MEGABYTES	NUMBER	Number of single block megabytes read
SMALL_WRITE_MEGABYTES	NUMBER	Number of single block megabytes written
LARGE_READ_MEGABYTES	NUMBER	Number of multiblock megabytes read

Column	Datatype	Description
LARGE_WRITE_MEGABYTES	NUMBER	Number of multiblock megabytes written
SMALL_READ_REQUESTS	NUMBER	Number of single block read requests
SMALL_WRITE_REQUESTS	NUMBER	Number of single block write requests
LARGE_READ_REQUESTS	NUMBER	Number of multiblock read requests
LARGE_WRITE_REQUESTS	NUMBER	Number of multiblock write requests
PQS_COMPLETED	NUMBER	Total number of completed parallel statements in the consumer group
PQ_SERVERS_USED	NUMBER	Total number of parallel servers used by completed parallel statements in the consumer group
PQS_QUEUED	NUMBER	Number of times that sessions in the consumer group were queued when trying to run parallel statements
PQ_ACTIVE_TIME	NUMBER	Cumulative sum of the parallel active times for all completed parallel statements in the consumer group (in milliseconds)
PQ_QUEUED_TIME	NUMBER	Total amount of time that sessions in the consumer group were queued when trying to run parallel statements (in milliseconds)
PQ_QUEUE_TIME_OUTS	NUMBER	Number of times that parallel statements from sessions in the consumer group timed out because their queue time exceeded the Resource Manager plan's PARALLEL_QUEUE_TIMEOUT limit
CURRENT_PQS_ACTIVE	NUMBER	Number of active parallel statements in the consumer group. This value does not include parallel statements that are never queued, such as GV$ queries.
CURRENT_PQ_SERVERS_ACTIVE	NUMBER	Number of active parallel servers in the consumer group. This value does not include servers running parallel statements that are never queued, such as GV$ queries.
CURRENT_PQS_QUEUED	NUMBER	Number of sessions in the consumer group that are waiting in the parallel statement queue trying to run parallel statements
SGA_BYTES	NUMBER	The current SGA usage by this PDB in bytes
BUFFER_CACHE_BYTES	NUMBER	The current usage of buffer cache by this PDB in bytes
SHARED_POOL_BYTES	NUMBER	The current usage of shared pool by this PDB in bytes
PGA_BYTES	NUMBER	The current usage of PGA by this PDB in bytes
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

See Also:

"V$RSRC_CONSUMER_GROUP"

9.199 V$RSRC_PDB_HISTORY

V$RSRC_PDB_HISTORY displays a history of consumer group statistics for each entry in V$RSRC_PDB that has a non-NULL plan by pluggable database (PDB).

When the STATISTICS_LEVEL is set to TYPICAL or ALL, this view contains information about CPU utilization and wait times even when no Resource Manager plan is set or when the Resource Manager plan does not monitor CPU or session resources.

A new window is created in V$RSRC_PDB_HISTORY when a multitenant container database (CDB) changes its resource plan. The plan windows for the CDB are not impacted by a PDB resource plan change.

V$RSRC_PDB_HISTORY covers the same time period for all PDBs. This view is specifically designed for comparing statistics across different PDBs.

Since V$RSRC_PDB_HISTORY does not contain information at the consumer group level, it is not useful for comparing consumer groups inside a PDB.

Column	Datatype	Description
SEQUENCE#	NUMBER	A sequential counter that uniquely describes the V$RSRC_PDB entry to which these consumer group statistics apply. When the instance is restarted, this value is reset to zero.
PDB_NAME	VARCHAR2(32)	PBD name
CPU_WAIT_TIME	NUMBER	Cumulative amount of time that sessions waited for CPU because of resource management. This does not include waits due to latch or enqueue contention, I/O waits, and so on. When CPU resources are not being actively managed, this value is set to zero.
CPU_WAITS	NUMBER	Cumulative number of times all sessions in the consumer group had to wait for CPU because of resource management. This does not include waits due to latch or enqueue contention, I/O waits, and so on. When CPU resources are not being actively managed, this value is set to zero.
CONSUMED_CPU_TIME	NUMBER	Cumulative amount of CPU time consumed by all sessions in the consumer group (in milliseconds)
YIELDS	NUMBER	Cumulative number of times that sessions in the consumer group had to yield CPU to other sessions because of quantum expiration. When CPU resources are not being actively managed, this value is set to zero.
IO_SERVICE_TIME	NUMBER	Cumulative I/O wait time (in milliseconds)
IO_SERVICE_WAITS	NUMBER	Total number of wait requests
SMALL_READ_MEGABYTES	NUMBER	Number of single block megabytes read
SMALL_WRITE_MEGABYTES	NUMBER	Number of single block megabytes written
LARGE_READ_MEGABYTES	NUMBER	Number of multiblock megabytes read
LARGE_WRITE_MEGABYTES	NUMBER	Number of multiblock megabytes written
SMALL_READ_REQUESTS	NUMBER	Number of single block read requests
SMALL_WRITE_REQUESTS	NUMBER	Number of single block write requests
LARGE_READ_REQUESTS	NUMBER	Number of multiblock read requests
LARGE_WRITE_REQUESTS	NUMBER	Number of multiblock write requests
PQS_COMPLETED	NUMBER	Total number of completed parallel statements in the consumer group

Column	Datatype	Description
PQ_SERVERS_USED	NUMBER	Total number of parallel servers used by completed parallel statements in the consumer group
PQS_QUEUED	NUMBER	Number of times that sessions in the consumer group were queued when trying to run parallel statements
PQ_ACTIVE_TIME	NUMBER	Cumulative sum of the parallel active times for all completed parallel statements in the consumer group (in milliseconds)
PQ_QUEUED_TIME	NUMBER	Total amount of time that sessions in the consumer group were queued when trying to run parallel statements (in milliseconds)
PQ_QUEUE_TIME_OUTS	NUMBER	Number of times that parallel statements from sessions in the consumer group timed out because their queue time exceeded the Resource Manager plan's PARALLEL_QUEUE_TIMEOUT limit
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

See Also:

"V$RSRC_CONS_GROUP_HISTORY"

9.200 V$RSRC_PLAN

V$RSRC_PLAN displays the names of all currently active resource plans.

Column	Datatype	Description
ID	NUMBER	Resource plan ID (a unique number, consistent across database shutdowns and startups). This is also the data dictionary object ID.
NAME	VARCHAR2(32)	Name of the resource plan
IS_TOP_PLAN	VARCHAR2(5)	Indicates whether the resource plan is the current top plan (TRUE) or whether the resource plan is a subplan of the current top plan (FALSE)
CPU_MANAGED	VARCHAR2(3)	Indicates whether the resource plan has parameters that specify a policy for how the Resource Manager should schedule sessions to manage CPU usage (ON) or whether Resource Manager is not managing CPU usage (OFF)
CPU_SCOPE	VARCHAR2(13)	Scope of the policy for how the Resource Manager should schedule sessions to manage CPU usage. Possible values: • OFF - Resource Manager is not managing CPU usage (CPU_MANAGED = OFF) • INSTANCE_ONLY - Resource Manage is managing CPU usage only within the database instance • SERVER_WIDE - Resource Manager is using server-level inter-instance CPU resource management

Column	Datatype	Description
INSTANCE_CAGING	VARCHAR2(3)	Indicates whether instance caging is enabled (ON) or disabled (OFF). Instance caging is enabled if the CPU_COUNT initialization parameter is explicitly modified to a value other than 0 and Resource Manager is enabled.
PARALLEL_SERVERS_ACTIVE	NUMBER	Total number of active parallel servers on the instance
PARALLEL_SERVERS_TOTAL	NUMBER	The value of PARALLEL_SERVERS_TARGET for the instance. Parallel statements are queued if the total number of active parallel servers exceeds this value.
PARALLEL_EXECUTION_MANAGED	VARCHAR2(32)	State of parallel statement queuing: • OFF - Parallel statement queuing is disabled • STARTUP - Parallel statement queuing is enabled. This is a temporary state that can occur when an Oracle RAC database is undergoing configuration changes • FIFO - Parallel statement queuing is enabled. All parallel statements are managed in a single Oracle RAC FIFO queue • FULL - Parallel statement queuing is enabled. All parallel statements are managed in per-consumer group queues according to the current resource plan. This state is used when a resource plan that contains resource allocation directives (MGMT_P*) is enabled. • DISABLED - Parallel statement queuing is disabled. This state can occur when memory is unavailable for use by parallel statement queuing in the System Global Area (SGA). Restart the Oracle instance to re-enable parallel statement queuing. For an Oracle RAC database, only the instance running as primary Database Resource Manager (DBRM) shows the correct state of parallel statement queuing. All other instances default to the value FULL. A single instance database always shows the correct value for this field.
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data
DIRECTIVE_TYPE	VARCHAR2(32)	The type of directive used by this PDB: • DEFAULT_DIRECTIVE: The default plan directive • PDB: A PDB directive • PROFILE: A profile directive
SHARES	NUMBER	Resource allocation for this PDB, expressed in shares
UTILIZATION_LIMIT	NUMBER	Maximum percentage of the container's resources allowed for this PDB
PARALLEL_SERVER_LIMIT	NUMBER	Maximum percentage of the parallel target the PDB can use before queuing subsequent parallel queries
MEMORY_MIN	NUMBER	The percentage of Exadata Smart Flash Cache and Exadata PMEM cache that is guaranteed to the PDB This percentage is based on the total amount of space allocated to the CDB for Exadata Smart Flash Cache and Exadata PMEM cache. See *Oracle Exadata System Software User's Guide* for more information.

Column	Datatype	Description
MEMORY_LIMIT	NUMBER	The maximum percentage of Exadata Smart Flash Cache and Exadata PMEM cache that the PDB can use This percentage is based on the total amount of space allocated to the CDB for Exadata Smart Flash Cache and Exadata PMEM cache. See *Oracle Exadata System Software User's Guide* for more information.
PROFILE	VARCHAR2(32)	The value of the DB_PERFORMANCE_PROFILE initialization parameter for this PDB
CPU_COUNT	NUMBER	Value of the CPU_COUNT initialization parameter in the container to which the data pertains
CPU_MIN_COUNT	NUMBER	Value of the CPU_MIN_COUNT initialization parameter in the container to which the data pertains

The resource plan with CON_ID=ROOT is the CDB resource plan.

See Also:

- "DBA_RSRC_PLANS" for a listing of all plans in the database
- "DBA_CDB_RSRC_PLANS" for information about CDB resource plans
- "DBA_CDB_RSRC_PLAN_DIRECTIVES" for information about CDB resource plan directives
- "DB_PERFORMANCE_PROFILE"
- *Oracle Database Administrator's Guide* for information on resource plans
- *Oracle Database PL/SQL Packages and Types Reference* for information on defining resource allocation methods for consumer groups with the DBMS_RESOURCE_MANAGER package

9.201 V$RSRC_PLAN_CPU_MTH

V$RSRC_PLAN_CPU_MTH displays all available CPU resource allocation methods defined for resource plans.

Column	Datatype	Description
NAME	VARCHAR2(40)	Name of the resource allocation method
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

See Also:

- "V$RSRC_CONSUMER_GROUP_CPU_MTH" for a listing of resource allocation methods defined for consumer groups
- *Oracle Database Administrator's Guide* for information on resource plans
- *Oracle Database PL/SQL Packages and Types Reference* for information on defining resource allocation methods for consumer plans with the DBMS_RESOURCE_MANAGER package

9.202 V$RSRC_PLAN_HISTORY

V$RSRC_PLAN_HISTORY displays a history of when a resource plan was enabled, disabled, or modified on the instance. Up to 15 of the most recent entries are shown. Once the database is opened, this view shows at least one row. The row with the most recent START_TIME and with END_TIME equal to NULL gives information about the current resource plan.

Column	Datatype	Description
SEQUENCE#	NUMBER	A sequential counter that uniquely describes a row. When the instance is restarted, this value is reset to zero.
ID	NUMBER	Resource plan ID; NULL if the Resource Manager was disabled
NAME	VARCHAR2(30)	Resource plan name; NULL if the Resource Manager was disabled
START_TIME	DATE	Time that the resource plan was enabled
END_TIME	DATE	Time that the resource plan was disabled; NULL if the row contains the current resource plan information
ENABLED_BY_SCHEDULER	VARCHAR2(5)	Indicates whether the plan was enabled by a Job Scheduler window (TRUE) or not (FALSE)
WINDOW_NAME	VARCHAR2(128)	Job Scheduler window that triggered the resource plan event; NULL if a user triggered the resource plan event
ALLOWED_AUTOMATED_SWITCHES	VARCHAR2(5)	Indicates whether automated plan switches were allowed after this resource plan event (TRUE) or whether automated plan switches were disabled after this resource plan event (FALSE) **See Also:** the SWITCH_PLAN procedure
CPU_MANAGED	VARCHAR2(3)	Indicates whether the resource plan has parameters that specify a policy for how the Resource Manager should schedule sessions to manage CPU usage (ON) or whether Resource Manager is not managing CPU usage (OFF)
CPU_SCOPE	VARCHAR2(13)	Scope of the policy for how the Resource Manager should schedule sessions to manage CPU usage. Possible values: • OFF - Resource Manager is not managing CPU usage (CPU_MANAGED = OFF) • INSTANCE_ONLY - Resource Manage is managing CPU usage only within the database instance • SERVER_WIDE - Resource Manager is using server-level inter-instance CPU resource management
INSTANCE_CAGING	VARCHAR2(3)	Indicates whether instance caging is enabled (ON) or disabled (OFF). Instance caging is enabled if the CPU_COUNT initialization parameter is explicitly modified to a value other than 0 and Resource Manager is enabled.

Column	Datatype	Description
PARALLEL_EXECUTION_MANAGED	VARCHAR2(32)	State of parallel statement queuing: • OFF - Parallel statement queuing is disabled • STARTUP - Parallel statement queuing is enabled. This is a temporary state that can occur when an Oracle RAC database is undergoing configuration changes • FIFO - Parallel statement queuing is enabled. All parallel statements are managed in a single Oracle RAC FIFO queue • FULL - Parallel statement queuing is enabled. All parallel statements are managed in per-consumer group queues according to the current resource plan. This state is used when a resource plan that contains resource allocation directives (MGMT_P*) is enabled. • DISABLED - Parallel statement queuing is disabled. This state can occur when memory is unavailable for use by parallel statement queuing in the System Global Area (SGA). Restart the Oracle instance to re-enable parallel statement queuing. For an Oracle RAC database, only the instance running as primary Database Resource Manager (DBRM) shows the correct state of parallel statement queuing. All other instances default to the value FULL. A single instance database always shows the correct value for this field.
CPU_COUNT	NUMBER	Value of the CPU_COUNT initialization parameter in the container to which the data pertains
CPU_MIN_COUNT	NUMBER	Value of the CPU_MIN_COUNT initialization parameter in the container to which the data pertains
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.203 V$RSRC_SESSION_INFO

V$RSRC_SESSION_INFO displays Resource Manager statistics per session.

Column	Datatype	Description
SID	NUMBER	Session identifier
CURRENT_CONSUMER_GROUP_ID	NUMBER	Object ID of the consumer group in which the session currently belongs; NULL if the session has not yet logged in
CURRENT_CONSUMER_GROUP	VARCHAR2(32)	The name of the consumer group in which the session currently belongs
ORIG_CONSUMER_GROUP_ID	NUMBER	Object ID of the consumer group in which the session was placed by the consumer group mappings; NULL if the session has not yet logged in This group may not be the current group because the SWITCH_GROUP directive in the current plan may have changed the session's current group. This group may not equal the MAPPED_CONSUMER_GROUP because the MAPPED_CONSUMER_GROUP may not be part of the current plan.

Column	Datatype	Description
MAPPING_ATTRIBUTE	VARCHAR2(32)	Session attribute that was used to map the session into its original consumer group; NULL if no mapping was used **See Also:** "DBA_RSRC_GROUP_MAPPINGS" for more details
MAPPED_CONSUMER_GROUP	VARCHAR2(32)	Consumer group to which the session was originally mapped; NULL if no mapping was used This may not correspond to the original consumer group, because the mapped group may not be part of the current plan. **See Also:** "DBA_RSRC_GROUP_MAPPINGS" for more details
STATE	VARCHAR2(32)	Current state of the session: • NOT MANAGED - Session has not logged in or the current Resource Manager plan does not require the session to be managed at this point in time • RUNNING - Session is currently running on the CPU • WAITING FOR CPU - Session is ready to run. It is waiting for a CPU quantum to run. • QUEUED - Session is queued because the active session limit was reached • IDLE - Session is idle • IDLE BLKR - Session is idle and blocking another session • WAITING - Session is currently in a wait **See Also:** "V$SESSION_WAIT" for the wait type • WAITING_FOR_IO - Session is waiting to submit an I/O request • UNBOUND - Session is not bound to any process • EXITING - Session is about to terminate
ACTIVE	VARCHAR2(5)	Indicates whether the session is currently active (TRUE) or not (FALSE). This includes when one of the following conditions is true: • Session is in the top call • Session has a transaction in progress • Session is using temporary space objects • Session holds user enqueues
CURRENT_IDLE_TIME	NUMBER	Number of milliseconds the session has been idle (in states IDLE or IDLE BLKR) while in this consumer group; NULL if the current Resource Manager plan does not require updating this statistic. This value is reset to zero when the session becomes active.
CURRENT_CPU_WAIT_TIME	NUMBER	Amount of time (in milliseconds) the session has waited for CPU because of resource management (in state WAIT FOR CPU) while in the current consumer group; NULL if the current Resource Manager plan does not require updating this statistic. This does not include waits due to latch or enqueue contention, I/O waits, and so on. If SWITCH_TIME_IN_CALL is used, then this will be reset at the end of every call.
CPU_WAIT_TIME	NUMBER	Cumulative amount of time (in milliseconds) the session has waited for CPU (in its lifetime) because of resource management. This does not include waits due to latch or enqueue contention, I/O waits, and so on.
CURRENT_CPU_WAITS	NUMBER	Number of times the session had to wait for CPU because of resource management while in this consumer group; NULL if the current Resource Manager plan does not require updating this statistic. This does not include waits due to latch or enqueue contention, I/O waits, and so on. If SWITCH_TIME_IN_CALL is used, then this will be reset at the end of every call.

Column	Datatype	Description
CPU_WAITS	NUMBER	Cumulative number of times the session had to wait for CPU (in its lifetime) because of resource management. This does not include waits due to latch or enqueue contention, I/O waits, and so on.
CURRENT_CONSUMED_CPU_TIME	NUMBER	Amount of CPU time (in milliseconds) consumed by the session while in the current consumer group; NULL if the current Resource Manager plan does not require updating this statistic. If SWITCH_TIME_IN_CALL is used, then this will be reset at the end of every call.
CONSUMED_CPU_TIME	NUMBER	Cumulative amount of CPU time consumed by the session (in its lifetime) (in milliseconds)
CURRENT_ACTIVE_TIME	NUMBER	Amount of time (in milliseconds) the session has been active while in the current consumer group; NULL if the current Resource Manager plan does not require updating this statistic. Active time is time spent running and waiting while executing a call. It does not include the time a session waited for CPU resources. If SWITCH_TIME_IN_CALL is used, then this is reset at the end of every call.
ACTIVE_TIME	NUMBER	Cumulative amount of active time (in milliseconds) consumed by the session (in its lifetime)
CURRENT_QUEUED_TIME	NUMBER	Amount of time (in milliseconds) the current request from the session has been queued (in state QUEUED). If the session does not have a request currently queued up, then this number will be zero.
QUEUED_TIME	NUMBER	Total amount of time (in milliseconds) the session has spent in the QUEUED state (in its lifetime)
CURRENT_YIELDS	NUMBER	Number of times the session had to yield the CPU to other sessions (due to quantum expiration) while in the current consumer group; NULL if the current Resource Manager plan does not require updating this statistic. If SWITCH_TIME_IN_CALL is used, then this is reset at the end of every call.
YIELDS	NUMBER	Cumulative number of times the session had to yield CPU to other sessions due to quantum expiration (in its lifetime)
CURRENT_UNDO_CONSUMPTION	NUMBER	Current amount (in KB) of undo consumed by the session; NULL if the current Resource Manager plan does not have an UNDO_POOL directive
MAX_UNDO_CONSUMPTION	NUMBER	Maximum amount of undo consumed (in KB) during the session's lifetime. This value may not be updated because the current Resource Manager plan may not have an UNDO_POOL directive.
SQL_CANCELED	NUMBER	Number of times SQL queries running in the session were canceled due to exceeding the Resource Manager plan's SWITCH_TIME limit.
QUEUE_TIME_OUTS	NUMBER	Number of times requests from the session timed out because they queued longer than the Resource Manager plan's limit
ESTIMATED_EXECUTION_LIMIT_HIT	NUMBER	Number of times requests from the session were not run because the optimizer's estimated time to execute the query exceeded the MAX_EST_EXEC_TIME limit
CURRENT_IO_SERVICE_TIME	NUMBER	Current I/O wait time of the session (in milliseconds) for the current SQL operation
IO_SERVICE_TIME	NUMBER	Cumulative amount of I/O wait time by the session (in its lifetime) (in milliseconds)
CURRENT_IO_SERVICE_WAITS	NUMBER	Current I/O waits by session for the current SQL operation
IO_SERVICE_WAITS	NUMBER	Cumulative I/O waits by session (in its lifetime)
CURRENT_SMALL_READ_MEGABYTES	NUMBER	Number of single block megabytes read by the session for the current SQL operation

Column	Datatype	Description
SMALL_READ_MEGABYTES	NUMBER	Total number of single block megabytes read by the session (in its lifetime)
CURRENT_LARGE_READ_MEGABYTES	NUMBER	Number of multiblock megabytes read by the session for the current SQL operation
LARGE_READ_MEGABYTES	NUMBER	Total number of multiblock megabytes read by the session (in its lifetime)
CURRENT_SMALL_WRITE_MEGABYTES	NUMBER	Number of single block megabytes written by the session for the current SQL operation
SMALL_WRITE_MEGABYTES	NUMBER	Total number of single block megabytes written by the session (in its lifetime)
CURRENT_LARGE_WRITE_MEGABYTES	NUMBER	Number of multiblock megabytes written by the session for the current SQL operation
LARGE_WRITE_MEGABYTES	NUMBER	Total number of multiblock megabytes written by the session (in its lifetime)
CURRENT_SMALL_READ_REQUESTS	NUMBER	Number of single block read requests by the session for the current SQL operation
SMALL_READ_REQUESTS	NUMBER	Total number of single block read requests by the session (in its lifetime)
CURRENT_SMALL_WRITE_REQUESTS	NUMBER	Number of single block write requests by the session for the current SQL operation
SMALL_WRITE_REQUESTS	NUMBER	Total number of single block write requests by the session (in its lifetime)
CURRENT_LARGE_READ_REQUESTS	NUMBER	Number of multiblock read requests by the session for the current SQL operation
LARGE_READ_REQUESTS	NUMBER	Total number of multiblock read requests by the session (in its lifetime)
CURRENT_LARGE_WRITE_REQUESTS	NUMBER	Number of multiblock write requests by the session for the current SQL operation
LARGE_WRITE_REQUESTS	NUMBER	Total number of multiblock write requests by the session (in its lifetime)
CURRENT_PQ_ACTIVE_TIME	NUMBER	Amount of time that the current active parallel statement has been executing for the current SQL operation, not including the amount of time that the statement has been queued (in milliseconds). If the parallel statement is queued, then the value is 0.
PQ_ACTIVE_TIME	NUMBER	Cumulative amount of time that parallel statements have been executed over the lifetime of the session (in milliseconds)
DOP	NUMBER	Degree of parallelism for the active or queued parallel statement, if there are any in the session
PQ_SERVERS	NUMBER	The number of active parallel servers if the session is active and running the parallel query. If the query is queued, the number of parallel servers that this query is trying to run with is shown.
ESTIMATED_EXECUTION_TIME	NUMBER	Estimated execution time for the parallel statement, as estimated by the optimizer (in milliseconds). You can compare this value to CURRENT_PQ_ACTIVE_TIME to estimate how much longer the parallel statement will run.
CURRENT_PQ_QUEUED_TIME	NUMBER	Amount of time that the current parallel statement in the session has been queued (in milliseconds) for the current SQL operation. If the session does not have a queued parallel statement, then the value is 0.
PQ_QUEUED_TIME	NUMBER	Total amount of time that the session has spent in the PQ_QUEUED state in its lifetime (in milliseconds)

Column	Datatype	Description
PQ_QUEUED	NUMBER	Number of times that parallel statements in the session have been queued
PQ_QUEUE_TIME_OUTS	NUMBER	Number of times that parallel statements in the session timed out because their queue time exceeded the Resource Manager plan's PARALLEL_QUEUE_TIMEOUT limit
PQ_ACTIVE	VARCHAR2(5)	Indicates whether the session is actively running a parallel statement (TRUE) or not (FALSE).
PQ_STATUS	VARCHAR2(44)	The status of the parallel statement running in this session: • Active: The parallel statement is currently active and running • Queued: The parallel statement is queued and is not a possible candidate to be run next • Queue head: The parallel statement is queued due to the global systemwide limit and is the next parallel statement to be dequeued and run • Queue head - waiting on CG limit: The parallel statement is queued due to the consumer group limit and is the one to be run next from the session's consumer group • Queue head - waiting on service availability: The parallel statement is queued since there are not enough parallel servers available for the service this session is running on. This session is the one to be run next from the session's service • Queue head - waiting on PDB limit: The parallel statement is queued due to the PDB limit and is the one to be run next from the session's PDB • NULL: The session currently does not involve a parallel execution
CURRENT_LOGICAL_IOS	NUMBER	Number of logical I/O requests by the session for the current SQL operation
LOGICAL_IOS	NUMBER	Total number of logical I/O requests in this session's lifetime
CURRENT_ELAPSED_TIME	NUMBER	Elapsed time of the session's current SQL operation
ELAPSED_TIME	NUMBER	Total elapsed time for all of this session's SQL operations in its lifetime
LAST_ACTION	VARCHAR2(48)	The most recent action that was taken on this SQL operation by Resource Manager. Its value is one of the following: • CANCEL_SQL • KILL_SESSION • LOG_ONLY • SWITCH TO <CG NAME> For the last value, <CG NAME> is the name of the consumer group that the SQL operation was switched to. If the Resource Plan has since been changed then <CG NAME> is the ID of the consumer group.
LAST_ACTION_REASON	VARCHAR2(30)	The reason for the most recent action that was taken on this SQL operation by Resource Manager. Its value is one of the following: • SWITCH_CPU_TIME • SWITCH_IO_REQS • SWITCH_IO_MBS • SWITCH_ELAPSED_TIME • SWITCH_IO_LOGICAL
LAST_ACTION_TIME	DATE	The time of the most recent action that was taken on this SQL operation by Resource Manager
CURRENT_PGA	NUMBER	Amount of PGA memory (in bytes) used by the session for the current SQL operation

Column	Datatype	Description
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.204 V$RSRCMGRMETRIC

V$RSRCMGRMETRIC displays information about resources consumed and wait times per consumer group.

When the STATISTICS_LEVEL is set to TYPICAL or ALL, this view contains information about CPU utilization and wait times even when no Resource Manager plan is set or when the Resource Manager plan does not monitor CPU or session resources. Metrics are collected and stored every minute when CPU utilization is not being monitored.

Column	Datatype	Description
BEGIN_TIME	DATE	Begin time of the interval
END_TIME	DATE	End time of the interval
INTSIZE_CSEC	NUMBER	Interval size (in hundredths of a second)
SEQUENCE#	NUMBER	A sequential counter that uniquely describes the V$RSRC_PLAN_HISTORY entry to which these consumer group statistics apply. When the instance is restarted, this value is reset to zero.
CONSUMER_GROUP_ID	NUMBER	Consumer group object ID (a unique number, consistent across database shutdowns and startups)
CONSUMER_GROUP_NAME	VARCHAR2(30)	Name of the consumer group
CPU_CONSUMED_TIME	NUMBER	Cumulative amount of CPU time consumed by all sessions in the consumer group, in milliseconds
CPU_WAIT_TIME	NUMBER	Cumulative amount of time that sessions waited for CPU because of resource management, in milliseconds. This does not include waits due to latch or enqueue contention, I/O waits, and so on. When CPU resources are not being actively managed, this value is set to zero.
CPU_EXHAUSTED	NUMBER	Percentage of time that CPUs are exhausted, out of the total number of checks in the consumer group
NUM_CPUS	NUMBER	Number of CPUs that the Resource Manager is utilizing. If instance caging is enabled, then this column is equal to the value of the CPU_COUNT initialization parameter. If instance caging is not enabled, then this column is equal to the total number of CPUs in the system.
RUNNING_SESSIONS_LIMIT	NUMBER	Maximum number of sessions in the consumer group that can run simultaneously. The value of this column is NUM_CPUS multiplied by the consumer group's MAX_UTILIZATION_LIMIT directive in the current Resource Manager plan.
AVG_RUNNING_SESSIONS	NUMBER	Average number of sessions in the consumer group that are currently running

Column	Datatype	Description
AVG_WAITING_SESSIONS	NUMBER	Average number of sessions in the consumer group that are waiting for CPU due to resource management. When CPU resources are not being actively managed, this value is set to zero.
CPU_UTILIZATION_LIMIT	NUMBER	Maximum percentage of CPU that the consumer group can use at any time, with respect to the total number of CPUs in the system. The value of this column is RUNNING_SESSIONS_LIMIT divided by the number of CPUs in the system. If instance caging is enabled, then this value is derived using the CPU_COUNT initialization parameter.
AVG_CPU_UTILIZATION	NUMBER	Average percentage of CPU consumed by the consumer group, with respect to the total number of CPUs in the system
CPU_DECISIONS	NUMBER	Percentage of CPU decisions for which the consumer group was present. When CPU resources are not being actively managed, this value is set to zero. This column is deprecated.
CPU_DECISIONS_EXCLUSIVE	NUMBER	Percentage of the CPU decisions for which the consumer group was present and was the only consumer group present. When CPU resources are not being actively managed, this value is set to zero. This column is deprecated.
CPU_DECISIONS_WON	NUMBER	Percentage of the CPU decisions that the consumer group won. When CPU resources are not being actively managed, this value is set to zero. This column is deprecated.
IO_REQUESTS	NUMBER	I/O requests
IO_MEGABYTES	NUMBER	I/O megabytes
AVG_ACTIVE_PARALLEL_STMTS	NUMBER	The average number of parallel statements that were running during the 1-minute metric window
AVG_QUEUED_PARALLEL_STMTS	NUMBER	The average number of parallel statements that were queued during the 1-minute metric window
AVG_ACTIVE_PARALLEL_SERVERS	NUMBER	The average number of parallel servers that were actively running as part of a parallel statement during the 1-minute metric window
AVG_QUEUED_PARALLEL_SERVERS	NUMBER	The average number of parallel servers that were requested by queued parallel statements during the 1-minute metric window
PARALLEL_SERVERS_LIMIT	NUMBER	The number of parallel servers allowed to be used by this consumer group during the 1-minute metric window
PLAN_NAME	VARCHAR2(30)	Resource Manager plan name
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

See Also:

"STATISTICS_LEVEL"

9.205 V$RSRCMGRMETRIC_HISTORY

V$RSRCMGRMETRIC_HISTORY displays a history (the last one hour) of resource manager metrics, taken from V$RSRCMGRMETRIC. When a resource plan is set, this history is cleared and restarted. This view provides information about resources consumed and wait times per consumer group. The columns for V$RSRCMGRMETRIC_HISTORY are the same as those for V$RSRCMGRMETRIC.

> **See Also:**
>
> "V$RSRCMGRMETRIC"

9.206 V$RSRCPDBMETRIC

V$RSRCPDBMETRIC displays information about resources consumed and wait times per PDB.

When the STATISTICS_LEVEL is set to TYPICAL or ALL, this view contains information about CPU utilization and wait times even when no Resource Manager plan is set or when the Resource Manager plan does not monitor CPU or session resources. Metrics are collected and stored every minute when CPU utilization is not being monitored.

Column	Datatype	Description
BEGIN_TIME	DATE	Begin time of the interval
END_TIME	DATE	End time of the interval
INTSIZE_CSEC	NUMBER	Interval size (in hundredths of a second)
SEQUENCE#	NUMBER	A sequential counter that uniquely describes the V$RSRC_PLAN_HISTORY entry to which these PDB statistics apply. When the instance is restarted, this value is reset to zero.
CPU_CONSUMED_TIME	NUMBER	Cumulative amount of CPU time consumed by all sessions in the PDB, in milliseconds
CPU_WAIT_TIME	NUMBER	Cumulative amount of time that sessions waited for CPU because of resource management, in milliseconds. This does not include waits due to latch or enqueue contention, I/O waits, and so on. When CPU resources are not being actively managed, this value is set to zero.
CPU_EXHAUSTED	NUMBER	Percentage of time that CPUs are exhausted, out of the total number of checks in the PDB
NUM_CPUS	NUMBER	Number of CPUs that the Resource Manager is utilizing. If instance caging is enabled, then this column is equal to the value of the CPU_COUNT initialization parameter. If instance caging is not enabled, then this column is equal to the total number of CPUs in the system.
RUNNING_SESSIONS_LIMIT	NUMBER	Maximum number of sessions in the PDB that can run simultaneously. The value of this column is NUM_CPUS multiplied by the PDB's MAX_UTILIZATION_LIMIT directive in the current Resource Manager plan.
AVG_RUNNING_SESSIONS	NUMBER	Average number of sessions in the PDB that are currently running
AVG_WAITING_SESSIONS	NUMBER	Average number of sessions in the PDB that are waiting for CPU due to resource management. When CPU resources are not being actively managed, this value is set to zero.

Column	Datatype	Description
CPU_UTILIZATION_LIMIT	NUMBER	Maximum percentage of CPU that the PDB can use at any time, with respect to the total number of CPUs in the system. The value of this column is RUNNING_SESSIONS_LIMIT divided by the number of CPUs in the system. If instance caging is enabled, then this value is derived using the CPU_COUNT initialization parameter.
AVG_CPU_UTILIZATION	NUMBER	Average percentage of CPU consumed by the PDB, with respect to the total number of CPUs in the system
IOPS	NUMBER	I/O operations per second during the previous minute for this PDB
IOMBPS	NUMBER	I/O megabytes per second during the previous minute for this PDB
IOPS_THROTTLE_EXEMPT	NUMBER	Indicates how much of the I/O per second in the current PDB was exempted from throttling. For example, if the value in the IOPS column is 20 I/Os and the value in the IOPS_THROTTLE_EXEMPT column is 5 I/Os, then 5 I/Os of the 20 I/Os in that second were exempted from throttling. I/O throttling is defined by the MAX_IOPS database initialization parameter.
IOMBPS_THROTTLE_EXEMPT	NUMBER	Indicates how many megabytes of I/O executed per second in the current PDB were exempted from throttling. For example, if the value in the IOMBPS column is 200 megabytes and the value in the IOMBPS_THROTTLE_EXEMPT column is 50 megabytes, then 50 megabytes of the 200 megabytes were exempt from throttling. I/O megabytes per second throttling is defined by the MAX_MBPS database initialization parameter.
AVG_IO_THROTTLE	NUMBER	Average throttle time per I/O operation in milliseconds during the previous minute for this PDB
AVG_ACTIVE_PARALLEL_STMTS	NUMBER	The average number of parallel statements that were running during the 1-minute metric window
AVG_QUEUED_PARALLEL_STMTS	NUMBER	The average number of parallel statements that were queued during the 1-minute metric window
AVG_ACTIVE_PARALLEL_SERVERS	NUMBER	The average number of parallel servers that were actively running as part of a parallel statement during the 1-minute metric window
AVG_QUEUED_PARALLEL_SERVERS	NUMBER	The average number of parallel servers that were requested by queued parallel statements during the 1-minute metric window
PARALLEL_SERVERS_LIMIT	NUMBER	The number of parallel servers allowed to be used by this PDB during the 1-minute metric window
SGA_BYTES	NUMBER	The current SGA usage by this PDB in bytes
BUFFER_CACHE_BYTES	NUMBER	The current usage of the buffer cache by this PDB in bytes
SHARED_POOL_BYTES	NUMBER	The current usage of the shared pool by this PDB in bytes
PGA_BYTES	NUMBER	The current PGA usage by this PDB in bytes
PLAN_NAME	VARCHAR2(30)	Resource Manager plan name
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

> **See Also:**
>
> "STATISTICS_LEVEL"

9.207 V$RSRCPDBMETRIC_HISTORY

V$RSRCPDBMETRIC_HISTORY displays a history (the last one hour) of resource manager metrics for a PDB, taken from V$RSRCPDBMETRIC. When a resource plan is set, this history is cleared and restarted. This view provides information about resources consumed and wait times per consumer group.
The columns for V$RSRCPDBMETRIC_HISTORY are the same as those for V$RSRCPDBMETRIC.

> **See Also:**
>
> "V$RSRCPDBMETRIC"

9.208 V$RULE

V$RULE displays rule statistics. This view has a row for every rule loaded into shared memory.

Column	Datatype	Description
RULE_SET_OBJECT_ID	NUMBER	Rule set object ID
EVALUATION_CONTEXT_OBJECT_ID	NUMBER	Evaluation context object ID
RULE_OWNER	VARCHAR2(128)	Owner of the rule
RULE_NAME	VARCHAR2(128)	Name of the rule
RULE_CONDITION	VARCHAR2(200)	Rule condition
TRUE_HITS	NUMBER	Number of times the rule evaluated to TRUE
MAYBE_HITS	NUMBER	Number of times the rule evaluated to MAYBE
SQL_EVALUATIONS	NUMBER	Number of evaluations of the rule that were performed by issuing SQL
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

9.209 V$RULE_SET

V$RULE_SET displays rule set statistics. This view has a row for every rule set loaded into shared memory.

> **Note:**
>
> Querying the V$RULE_SET view may have a negative impact on performance if a database has a large library cache.

Column	Datatype	Description
OWNER	VARCHAR2(128)	Owner of the rule set
NAME	VARCHAR2(128)	Name of the rule set
CPU_TIME	NUMBER	Total CPU time (in hundredths of a second) spent in evaluation of the rule set
ELAPSED_TIME	NUMBER	Total elapsed time (in hundredths of a second) spent in evaluation of the rule set
FIRST_LOAD_TIME	DATE	First time the current cached copy of the rule set was loaded
LAST_LOAD_TIME	DATE	Last time the current cached copy of the rule set was loaded
LAST_LOADING_TIME	NUMBER	Total elapsed time (in hundredths of a second) spent to load the rule set the last time it was loaded
SHARABLE_MEM	NUMBER	Shared memory (in bytes) used by the rule set
RELOADS	NUMBER	Number of times the rule set object was reloaded in shared memory
INVALIDATIONS	NUMBER	Number of times the rule set object was invalidated
EVALUATIONS	NUMBER	Number of evaluations on the rule set
FIRST_HIT_EVALUATIONS	NUMBER	Number of evaluations on the rule set, with stop_on_first_hit set to TRUE
SIMPLE_RULES_ONLY_EVALUATIONS	NUMBER	Number of evaluations on the rule set, with simple_rules_only set to TRUE
SQL_FREE_EVALUATIONS	NUMBER	Number of evaluations on the rule set which did not internally issue SQL to evaluate rules
SQL_EXECUTIONS	NUMBER	Total number of SQL statements executed during evaluation of the rule set
CONDITIONS_PROCESSED	NUMBER	Total number of fast (indexed) conditions processed during evaluation of the rule set
TRUE_RULES	NUMBER	Total number of TRUE rules returned during evaluation of the rule set
MAYBE_RULES	NUMBER	Total number of MAYBE rules returned during evaluation of the rule set
VARIABLE_VALUE_FUNCTION_CALLS	NUMBER	Total number of calls made to user-defined functions to retrieve variable values (specified by the variable_value_function field in RE$VARIABLE_TYPE) made during evaluation of the rule set
VARIABLE_METHOD_FUNCTION_CALLS	NUMBER	Total number of calls made to user-defined functions to retrieve variable method values (specified by the variable_method_function field in RE$VARIABLE_TYPE) made during evaluation of the rule set

Column	Datatype	Description
EVALUATION_FUNCTION_CALLS	NUMBER	Total number of calls made to user-defined evaluation functions (specified as the evaluation_function argument to the DBMS_RULE_ADM.CREATE_EVALUATION_CONTEXT procedure) made during evaluation of the rule set
RESULT_CACHE_HITS	NUMBER	Number of result cache hits across all the sessions evaluating this rule set
IS_RESULT_CACHE	VARCHAR2(3)	Indicates whether this rule set result can be cached (YES) or not (NO)
RESULT_CACHE_ELEMENTS	NUMBER	Number of elements cached within the result cache
CON_ID	NUMBER	The ID of the container to which the data pertains. Possible values include: • 0: This value is used for rows containing data that pertain to the entire CDB. This value is also used for rows in non-CDBs. • 1: This value is used for rows containing data that pertain to only the root • *n*: Where *n* is the applicable container ID for the rows containing data

See Also:

Oracle Database PL/SQL Packages and Types Reference for more information about the DBMS_RULE_ADM.CREATE_EVALUATION_CONTEXT procedure

9.210 V$RULE_SET_AGGREGATE_STATS

V$RULE_SET_AGGREGATE_STATS displays statistics aggregated over all evaluations on all rule sets. This view has a row for each type of statistic.

Column	Datatype	Description
NAME	VARCHAR2(80)	Name of the statistic: • rule set evaluations (all) - Total number of evaluations on all rule sets • rule set evaluations (first_hit) - Total number of evaluations on rule sets with stop_on_first_hit set to TRUE • rule set evaluations (simple_rules_only) - Total number of evaluations on rule sets with simple_rules_only set to TRUE • rule set evaluations (SQL free) - Total number of evaluations on rule sets which did not internally issue SQL to evaluate rules • rule set evaluation time (CPU) - Total CPU time (in hundredths of a second) spent in evaluations on rule sets • rule set evaluation time (elapsed) - Total elapsed time (in hundredths of a second) spent in evaluations on rule sets • rule set SQL executions - Total number of SQL statements executed during evaluations on rule sets • rule set conditions processed - Total number of fast (indexed) conditions processed during evaluations on rule sets • rule set true rules - Total number of TRUE rules returned during evaluations on rule sets • rule set maybe rules - Total number of MAYBE rules returned during evaluations on rule sets • rule set user function calls (variable value function) - Total number of calls made to user-defined functions to retrieve variable values (specified by the variable_value_function field in RE$VARIABLE_TYPE) made during evaluations on rule sets • rule set user function calls (variable method function) - Total number of calls made to user-defined functions to retrieve variable method values (specified by the variable_method_function field in RE$VARIABLE_TYPE) made during evaluations on rule sets • rule set user function calls (evaluation function) - Total number of calls made to user-defined evaluation functions (specified as the evaluation_function argument to the DBMS_RULE_ADM.CREATE_EVALUATION_CONTEXT procedure) made during evaluations on rule sets
VALUE	NUMBER	Statistic value
CON_ID	NUMBER	The ID of the container to which the data pertains. The CON_ID value in this view is always 0. The rows pertain to the entire CDB or to the non-CDB.

See Also:

Oracle Database PL/SQL Packages and Types Reference for more information about the DBMS_RULE_ADM.CREATE_EVALUATION_CONTEXT procedure