6

Static Data Dictionary Views: DBA_HIST_LATCH to DBA_STORED_SETTINGS

This chapter contains the static data dictionary views DBA_HIST_LATCH through DBA_STORED_SETTINGS.

6.1 DBA_HIST_LATCH

DBA_HIST_LATCH displays historical aggregate latch statistics for both parent and child latches, grouped by latch name.

This view contains snapshots of V$LATCH.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
LATCH_HASH	NUMBER	NOT NULL	Latch hash
LATCH_NAME	VARCHAR2(64)	NOT NULL	Latch name
LEVEL#	NUMBER		Latch level
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)
SPIN_GETS	NUMBER		Number of willing-to-wait latch requests which missed the first try but succeeded while spinning
SLEEP[1 | 2 | 3 | 4]	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.
WAIT_TIME	NUMBER		Elapsed time spent waiting for the latch (in microseconds)
CON_DBID	NUMBER		The database ID of the PDB for the sampled session

Column	Datatype	NULL	Description
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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:

"V$LATCH"

6.2 DBA_HIST_LATCH_CHILDREN

DBA_HIST_LATCH_CHILDREN displays historical statistics about child latches.

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

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
LATCH_HASH	NUMBER	NOT NULL	Latch hash
LATCH_NAME	VARCHAR2(64)	NOT NULL	Latch name
CHILD#	NUMBER	NOT NULL	Child latch number (unique only to each parent latch)
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)
SPIN_GETS	NUMBER		Number of willing-to-wait latch requests which missed the first try but succeeded while spinning

Column	Datatype	NULL	Description
SLEEP[1 | 2 | 3 | 4]	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.
WAIT_TIME	NUMBER		Elapsed time spent waiting for the latch (in microseconds)
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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:

- "DBA_HIST_LATCH"
- "V$LATCH"

6.3 DBA_HIST_LATCH_MISSES_SUMMARY

DBA_HIST_LATCH_MISSES_SUMMARY displays historical summary statistics about missed attempts to acquire a latch.

This view contains snapshots of V$LATCH_MISSES.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
PARENT_NAME	VARCHAR2(64)	NOT NULL	Latch name of a parent latch
WHERE_IN_CODE	VARCHAR2(80)	NOT NULL	Location that attempted to acquire the latch
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
CON_DBID	NUMBER		The database ID of the PDB for the sampled session

Column	Datatype	NULL	Description
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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_MISSES"

6.4 DBA_HIST_LATCH_NAME

DBA_HIST_LATCH_NAME displays information about decoded latch names for the latches shown in DBA_HIST_LATCH.

This view contains a snapshot of V$LATCHNAME.

Column	Datatype	NULL	Description
DBID	NUMBER	NOT NULL	Database ID
LATCH_HASH	NUMBER	NOT NULL	Latch hash
LATCH_NAME	VARCHAR2(64)	NOT NULL	Latch name
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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:**
>
> - "DBA_HIST_LATCH"
> - "V$LATCHNAME"

6.5 DBA_HIST_LATCH_PARENT

DBA_HIST_LATCH_PARENT displays historical statistics about parent latches.

This view contains snapshots of V$LATCH_PARENT.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
LATCH_HASH	NUMBER	NOT NULL	Latch hash
LATCH_NAME	VARCHAR2(64)	NOT NULL	Latch name
LEVEL#	NUMBER	NOT NULL	Latch level
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)
SPIN_GETS	NUMBER		Number of willing-to-wait latch requests which missed the first try but succeeded while spinning
SLEEP[1 | 2 | 3 | 4]	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.
WAIT_TIME	NUMBER		Elapsed time spent waiting for the latch (in microseconds)
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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_PARENT"

6.6 DBA_HIST_LIBRARYCACHE

DBA_HIST_LIBRARYCACHE displays historical statistics about library cache performance and activity.

This view contains snapshots of V$LIBRARYCACHE.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
NAMESPACE	VARCHAR2(15)	NOT NULL	Library cache namespace
GETS	NUMBER		Number of times a lock was requested for objects of the namespace
GETHITS	NUMBER		Number of times an object's handle was found in memory
PINS	NUMBER		Number of times a PIN was requested for objects of the namespace
PINHITS	NUMBER		Number of times all of the metadata pieces of the library object were found in memory
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 the 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_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$LIBRARYCACHE"

6.7 DBA_HIST_LOG

DBA_HIST_LOG displays historical log file information from the control file. This view contains snapshots of V$LOG.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
GROUP#	NUMBER	NOT NULL	Log group number
THREAD#	NUMBER	NOT NULL	Log thread number
SEQUENCE#	NUMBER	NOT NULL	Log sequence number
BYTES	NUMBER		Size of the log (in bytes)
MEMBERS	NUMBER		Number of members in the log group
ARCHIVED	VARCHAR2(3)		Archive status (YES) or NO)
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. • INVALIDATED - Archived the current redo log without a log switch.
FIRST_CHANGE#	NUMBER		Lowest system change number (SCN) in the log
FIRST_TIME	DATE		Time of the first SCN in the log
CON_DBID	NUMBER		The database ID of the PDB for the sampled session

Column	Datatype	NULL	Description
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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"

6.8 DBA_HIST_MEM_DYNAMIC_COMP

DBA_HIST_MEM_DYNAMIC_COMP displays historical memory component sizes.

This view contains snapshots of V$MEMORY_DYNAMIC_COMPONENTS.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
COMPONENT	VARCHAR2(64)	NOT NULL	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

Column	Datatype	NULL	Description
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$MEMORY_DYNAMIC_COMPONENTS"

6.9 DBA_HIST_MEMORY_RESIZE_OPS

DBA_HIST_MEMORY_RESIZE_OPS displays memory resize operations history.

This view contains snapshots of V$MEMORY_RESIZE_OPS.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
COMPONENT	VARCHAR2(64)	NOT NULL	Component name
OPER_TYPE	VARCHAR2(13)	NOT NULL	Operation type: • STATIC • INITIALIZING • DISABLED • GROW • SHRINK • SHRINK_CANCEL
START_TIME	DATE	NOT NULL	Start time of the operation
END_TIME	DATE	NOT NULL	End time of the operation
TARGET_SIZE	NUMBER	NOT NULL	Requested value of the parameter after the resize
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
FINAL_SIZE	NUMBER		Real value of the parameter after the resize

Column	Datatype	NULL	Description
STATUS	VARCHAR2(9)		Completion status of the operation: • INACTIVE • PENDING • COMPLETE • CANCELLED • ERROR
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$MEMORY_RESIZE_OPS"

6.10 DBA_HIST_MEMORY_TARGET_ADVICE

DBA_HIST_MEMORY_TARGET_ADVICE displays memory target advice history.

This view contains snapshots of V$MEMORY_TARGET_ADVICE.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
MEMORY_SIZE	NUMBER	NOT NULL	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.
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.

Column	Datatype	NULL	Description
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
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$MEMORY_TARGET_ADVICE"

6.11 DBA_HIST_METRIC_NAME

DBA_HIST_METRIC_NAME describes attributes of the set of RDBMS metrics.

This view contains a snapshot of V$METRICNAME.

Column	Datatype	NULL	Description
DBID	NUMBER	NOT NULL	Database ID
GROUP_ID	NUMBER	NOT NULL	Metric Group ID
GROUP_NAME	VARCHAR2(64)		Metric group name
METRIC_ID	NUMBER	NOT NULL	Metric ID
METRIC_NAME	VARCHAR2(64)	NOT NULL	Metric name
METRIC_UNIT	VARCHAR2(64)	NOT NULL	Unit of measurement
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$METRICNAME"

6.12 DBA_HIST_MTTR_TARGET_ADVICE

DBA_HIST_MTTR_TARGET_ADVICE displays historical predictions of the number of physical I/O requests for the MTTR corresponding to each row.

The data also includes a physical I/O factor, which is the ratio of the number of estimated I/O requests to the number of I/O requests actually performed by the current MTTR setting during the measurement interval. This view contains snapshots of V$MTTR_TARGET_ADVICE.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
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 SET • READY SET • OFF
DIRTY_LIMIT	NUMBER		Dirty buffer limit derived from the MTTR being simulated
ESTD_CACHE_WRITES	NUMBER		Estimated number of cache physical writes under the MTTR
ESTD_CACHE_WRITE_FACTOR	NUMBER		Estimated cache physical write ratio under the 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 the MTTR
ESTD_TOTAL_WRITE_FACTOR	NUMBER		Estimated total physical write ratio under the 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/O requests under the MTTR
ESTD_TOTAL_IO_FACTOR	NUMBER		Estimated total I/O ratio under the MTTR. It is the ratio of the estimated total number of I/O requests to the total number of I/O requests under the current MTTR setting.
CON_DBID	NUMBER		The database ID of the PDB for the sampled session

Column	Datatype	NULL	Description
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$MTTR_TARGET_ADVICE"

6.13 DBA_HIST_MUTEX_SLEEP

DBA_HIST_MUTEX_SLEEP displays mutex sleep summary historical statistics information.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database identifier for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
MUTEX_TYPE	VARCHAR2(32)	NOT NULL	Mutex type
LOCATION	VARCHAR2(40)	NOT NULL	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_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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

6.14 DBA_HIST_OPTIMIZER_ENV

DBA_HIST_OPTIMIZER_ENV displays the optimizer environments that have been captured in the Workload Repository.

This view is used with the DBA_HIST_SQLSTAT view.

Column	Datatype	NULL	Description
DBID	NUMBER	NOT NULL	Database ID
OPTIMIZER_ENV_HASH_VALUE	NUMBER	NOT NULL	Hash value for the optimizer environment
OPTIMIZER_ENV	RAW(2000)		Optimizer environment
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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:

"DBA_HIST_SQLSTAT"

6.15 DBA_HIST_OPTIMIZER_ENV_DETAILS

DBA_HIST_OPTIMIZER_ENV_DETAILS displays parameter values for the optimizer environments that have been captured in the Workload Repository.

This view is used with the DBA_HIST_SQLSTAT view.

Column	Datatype	NULL	Description
DBID	NUMBER	NOT NULL	Database ID
OPTIMIZER_ENV_HASH_VALUE	NUMBER	NOT NULL	Hash value for the optimizer environment
NAME	VARCHAR2(50)		Parameter name
VALUE	VARCHAR2(25)		Parameter value
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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.

6.16 DBA_HIST_OSSTAT

DBA_HIST_OSSTAT displays historical operating system statistics.

This view contains snapshots of V$OSSTAT.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
STAT_ID	NUMBER	NOT NULL	Statistic ID
STAT_NAME	VARCHAR2(64)	NOT NULL	Statistic name
VALUE	NUMBER		Statistic value
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$OSSTAT"

6.17 DBA_HIST_OSSTAT_NAME

DBA_HIST_OSSTAT_NAME displays the names of the operating system statistics.

This view is used with DBA_HIST_OSSTAT.

Column	Datatype	NULL	Description
DBID	NUMBER	NOT NULL	Database ID
STAT_ID	NUMBER	NOT NULL	Statistic ID
STAT_NAME	VARCHAR2(64)	NOT NULL	Statistic name
CON_DBID	NUMBER		The database ID of the PDB for the sampled session

Column	Datatype	NULL	Description
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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:**
>
> "DBA_HIST_OSSTAT"

6.18 DBA_HIST_PARAMETER

DBA_HIST_PARAMETER displays historical information about the initialization parameters that were in effect for the instance.

This view contains snapshots of V$SYSTEM_PARAMETER.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
PARAMETER_HASH	NUMBER	NOT NULL	Parameter hash
PARAMETER_NAME	VARCHAR2(64)	NOT NULL	Name of the parameter
VALUE	VARCHAR2(512)		Parameter value for the session (if modified within the session); otherwise, the instance-wide parameter value
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)
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
CON_DBID	NUMBER		The database ID of the PDB for the sampled session

Column	Datatype	NULL	Description
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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"

6.19 DBA_HIST_PARAMETER_NAME

DBA_HIST_PARAMETER_NAME displays information about the parameter names captured in the workload repository.

This view is used with the DBA_HIST_PARAMETER view.

Column	Datatype	NULL	Description
DBID	NUMBER	NOT NULL	Database ID
PARAMETER_HASH	NUMBER	NOT NULL	Parameter hash
PARAMETER_NAME	VARCHAR2(64)	NOT NULL	Name of the parameter
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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:**
>
> "DBA_HIST_PARAMETER"

6.20 DBA_HIST_PDB_IN_SNAP

DBA_HIST_PDB_IN_SNAP captures a list of open pluggable databases (PDBs) at the time of the Automatic Workload Repository (AWR) snapshot. This view can be used with other DBA_HIST_ views to construct the number of opened PDBs at the time of the snapshot.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	AWR snapshot ID
DBID	NUMBER	NOT NULL	Database ID of the database that took this snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number of the instance that took this snapshot
CON_DBID	NUMBER		DBID of an open PDB at the time of the snapshot
FLAG	NUMBER		Flag field in capture properties of the PDB. Not used at this 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
OPEN_TIME_TZ	TIMESTAMP(3) WITH TIME ZONE		Time the PDB was last opened

6.21 DBA_HIST_PDB_INSTANCE

DBA_HIST_PDB_INSTANCE displays the pluggable databases (PDBs) and instances in the Workload Repository.

Column	Datatype	NULL	Description
DBID	NUMBER	NOT NULL	Database ID
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number
STARTUP_TIME	TIMESTAMP(3)	NOT NULL	Startup time of the instance
CON_DBID	NUMBER	NOT NULL	The database ID of the PDB for the sampled session
OPEN_TIME	TIMESTAMP(3)	NOT NULL	Time the PDB was last opened
OPEN_MODE	VARCHAR2(16)		Open mode of the database
PDB_NAME	VARCHAR2(128)		PDB name

Column	Datatype	NULL	Description
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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
SNAP_ID	NUMBER		The unique snapshot identifier of the snapshot that flushed the corresponding row
STARTUP_TIME_TZ	TIMESTAMP(3) WITH TIME ZONE		Startup time of the instance
OPEN_TIME_TZ	TIMESTAMP(3) WITH TIME ZONE		Time the PDB was last opened

6.22 DBA_HIST_PERSISTENT_QMN_CACHE

DBA_HIST_PERSISTENT_QMN_CACHE displays the historical summary background queue table activity.

This view contains snapshots from V$PERSISTENT_QMN_CACHE.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
QUEUE_TABLE_ID	NUMBER	NOT NULL	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)		Time when the queue table should be serviced by QMON servers
WINDOW_END_TIME	TIMESTAMP(3)		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)

Column	Datatype	NULL	Description
LAST_TMGR_PROCESSING_TIME	TIMESTAMP(3)		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)		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
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)		Last index block cleanup time
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$PERSISTENT_QMN_CACHE"

6.23 DBA_HIST_PERSISTENT_QUEUES

DBA_HIST_PERSISTENT_QUEUES displays Oracle Database AQ persistent queues historical statistics information.

This view contains snapshots of V$PERSISTENT_QUEUES.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot

Column	Datatype	NULL	Description
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
QUEUE_SCHEMA	VARCHAR2(128)	NOT NULL	Owner of the queue
QUEUE_NAME	VARCHAR2(128)	NOT NULL	Name of the queue
QUEUE_ID	NUMBER	NOT NULL	Identifier for 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
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
CON_DBID	NUMBER		The database ID of the PDB for the sampled session

Column	Datatype	NULL	Description
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$PERSISTENT_QUEUES"

6.24 DBA_HIST_PERSISTENT_SUBS

DBA_HIST_PERSISTENT_SUBS displays Oracle Database AQ persistent queue subscribers historical statistics information.

This view contains snapshots of V$PERSISTENT_SUBSCRIBERS.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
QUEUE_SCHEMA	VARCHAR2(128)	NOT NULL	Owner of the queue
QUEUE_NAME	VARCHAR2(128)	NOT NULL	Name of the queue
SUBSCRIBER_ID	NUMBER	NOT NULL	Internal subscriber number
SUBSCRIBER_NAME	VARCHAR2(128)		Name of the subscriber
SUBSCRIBER_ADDRESS	VARCHAR2(1024)		Address 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
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)

Column	Datatype	NULL	Description
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
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$PERSISTENT_SUBSCRIBERS"

6.25 DBA_HIST_PGA_TARGET_ADVICE

DBA_HIST_PGA_TARGET_ADVICE displays historical predictions of 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.

This view contains snapshots of V$PGA_TARGET_ADVICE.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
PGA_TARGET_FOR_ESTIMATE	NUMBER	NOT NULL	Value of PGA_AGGREGATE_TARGET for the 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
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

Column	Datatype	NULL	Description
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, PGA_AGGREGATE_TARGET should not be set to PGA_TARGET_FOR_ESTIMATE since Oracle will not be able to honor that target.
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$PGASTAT"
- "V$PGA_TARGET_ADVICE"
- "PGA_AGGREGATE_TARGET"

6.26 DBA_HIST_PGASTAT

DBA_HIST_PGASTAT displays historical PGA memory usage statistics as well as statistics about the automatic PGA memory manager when it is enabled.

This view contains snapshots of V$PGASTAT.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot

Column	Datatype	NULL	Description
INSTANCE_NUMBER	NUMBER	NOT NULL	Database instance number
NAME	VARCHAR2(64)	NOT NULL	Name of the statistic: • aggregate PGA auto target • aggregate PGA target parameter • bytes processed • cache hit percentage • extra bytes read/written • global memory bound • max processes count • maximum PGA allocated • maximum PGA used for auto workareas • maximum PGA used for manual workareas • over allocation count • PGA memory freed back to OS • process count • recompute count (total) • total freeable PGA memory • total PGA allocated • total PGA inuse • total PGA used for auto workareas • total PGA used for manual workareas **See Also:** V$PGASTAT for descriptions of the statistics
VALUE	NUMBER		Statistic value
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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"

6.27 DBA_HIST_PLAN_OPERATION_NAME

DBA_HIST_PLAN_OPERATION_NAME displays historical information about SQL plan operation names.

Column	Datatype	NULL	Description
DBID	NUMBER	NOT NULL	Database identifier
OPERATION_ID	NUMBER	NOT NULL	Plan operation identifier

Column	Datatype	NULL	Description
OPERATION_NAME	VARCHAR2(64)		Plan operation name. This value also appears in the SQL_PLAN_OPERATION column of the DBA_HIST_ACTIVE_SESS_HISTORY view.
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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

6.28 DBA_HIST_PLAN_OPTION_NAME

DBA_HIST_PLAN_OPTION_NAME displays historical information about SQL plan option names.

Column	Datatype	NULL	Description
DBID	NUMBER	NOT NULL	Database identifier
OPTION_ID	NUMBER	NOT NULL	Plan option identifier
OPTION_NAME	VARCHAR2(64)		Plan option name. This value also appears in the SQL_PLAN_OPTIONS column of the DBA_HIST_ACTIVE_SESS_HISTORY view.
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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

6.29 DBA_HIST_PROCESS_MEM_SUMMARY

DBA_HIST_PROCESS_MEM_SUMMARY displays historical information about dynamic PGA memory usage by named component categories for each process.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot

Column	Datatype	NULL	Description
CATEGORY	VARCHAR2(15)	NOT NULL	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
IS_INSTANCE_WIDE	NUMBER		This column shows whether the process memory detail is for only this container or for the whole instance. If the value is 1, the detail is for the whole instance. Any other value is the container ID for the container to which the detail pertains, as seen in the CON_ID column.
NUM_PROCESSES	NUMBER		Number of processes
NON_ZERO_ALLOCS	NUMBER		Number of processes with nonzero allocations
USED_TOTAL	NUMBER		Bytes of PGA memory used by the process for the category
ALLOCATED_TOTAL	NUMBER		Total number of bytes of PGA memory allocated by the process for the category.
ALLOCATED_AVG	NUMBER		Average number of bytes of PGA memory allocated by the process for the category
ALLOCATED_STDDEV	NUMBER		Standard deviation of the number of bytes of PGA memory allocated by the process for the category
ALLOCATED_MAX	NUMBER		Maximum bytes of PGA memory ever allocated by the process for the category
MAX_ALLOCATED_MAX	NUMBER		Maximum bytes of PGA memory that can be allocated by the process for the category
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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

6.30 DBA_HIST_PROCESS_WAITTIME

DBA_HIST_PROCESS_WAITTIME displays CPU and wait time by process types.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
PROCESS_TYPE	VARCHAR2(5)	NOT NULL	Process type

Column	Datatype	NULL	Description
DESCRIPTION	VARCHAR2(64)	NOT NULL	Process description
WAIT_CLASS_TYPE	VARCHAR2(64)	NOT NULL	Type of wait class
VALUE	NUMBER		Wait time or CPU used time in milliseconds
CON_DBID	NUMBER		The database ID of the PDB for the process
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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

6.31 DBA_HIST_RECOVERY_PROGRESS

DBA_HIST_RECOVERY_PROGRESS displays database recovery progress information for an instance.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
START_TIME	DATE	NOT NULL	Start time of the recovery operation
TYPE	VARCHAR2(64)	NOT NULL	Type of recovery operation being performed: • CRASH RECOVERY • INSTANCE RECOVERY • MEDIA RECOVERY
ITEM	VARCHAR2(32)	NOT NULL	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	NULL	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
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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

6.32 DBA_HIST_REPLICATION_TBL_STATS

DBA_HIST_REPLICATION_TBL_STATS displays replication table statistics for Oracle GoldenGate and XStream sessions. This view is intended for use with Automatic Workload Repository (AWR).

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
APPLY_NAME	VARCHAR2(128)	NOT NULL	Name of the apply process
TABLE_NAME	VARCHAR2(128)	NOT NULL	Name of the table
TABLE_OWNER	VARCHAR2(128)	NOT NULL	Owner of the table
SESSION_MODULE	VARCHAR2(64)	NOT NULL	Session module. Valid values: • XStream • GoldenGate
TOTAL_INSERTS	NUMBER		Number of insert operations on this table processed by this apply server
TOTAL_UPDATES	NUMBER		Number of update operations on this table processed by this apply server
TOTAL_DELETES	NUMBER		Number of delete operations on this table processed by this apply server
CDR_SUCCESSFUL	NUMBER		Number of successfully resolved conflicts
CDR_FAILED	NUMBER		Number of conflicts that could not be resolved due to an error during resolution
REPERR_CNT	NUMBER		The total number of errors for the replication operation
HANDLE_COLLISIONS	NUMBER		Number of collisions on this table handled by this apply server
CON_DBID	NUMBER		The database ID of the PDB for the sampled session

Column	Datatype	NULL	Description
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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

6.33 DBA_HIST_REPLICATION_TXN_STATS

DBA_HIST_REPLICATION_TXN_STATS displays replication transaction statistics for Oracle GoldenGate and XStream sessions.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
OBJECT_NAME	VARCHAR2(128)	NOT NULL	Object name
SESSION_TYPE	VARCHAR2(64)	NOT NULL	Type of session
SESSION_MODULE	VARCHAR2(64)	NOT NULL	Session module. Valid values: • XStream • GoldenGate
SOURCE_DATABASE	VARCHAR2(128)		Database where the transaction originated
SOURCE_TXN_ID	VARCHAR2(128)	NOT NULL	Original transaction ID at the source database
FIRST_LCR_TIME	DATE		Time of the first LCR (message in an error transaction)
TOTAL_LCRS_COUNT	NUMBER		Total number of LCRs for this replication transaction
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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

6.34 DBA_HIST_REPORTS

DBA_HIST_REPORTS displays information about XML reports captured into Automatic Workload Repository (AWR).

The reports themselves belong to components such as SQL Monitor, DBOP, and Real-Time ADDM.

Each XML report contains details about some activity of a component. For example, a SQL Monitor report contains a detailed report about a particular execution of a SQL statement, or a Real-Time ADDM report contains system performance data analyzed by Real-Time ADDM.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER		ID of the first Automatic Workload Repository (AWR) snapshot that will be taken after this report is generated
DBID	NUMBER		Database ID of the current database for the report
INSTANCE_NUMBER	NUMBER		Instance number (for an Oracle RAC system)
REPORT_ID	NUMBER		ID of the captured report
COMPONENT_ID	NUMBER		ID of the component (for example, SQL Monitor) whose report is captured
SESSION_ID	NUMBER		ID of the session corresponding to the captured report (currently used only for SQL Monitor reports)
SESSION_SERIAL#	NUMBER		Session serial number corresponding to the captured report (currently used only for SQL Monitor reports)
PERIOD_START_TIME	DATE		Time when the activity period started
PERIOD_END_TIME	DATE		Time when the activity period ended
GENERATION_TIME	DATE		Time when this report was generated
COMPONENT_NAME	VARCHAR2(128)		Name of the component whose report this is
REPORT_NAME	VARCHAR2(128)		Name of this report
REPORT_PARAMETERS	VARCHAR2(1024)		Parameters associated with this report
KEY1	VARCHAR2(128)		Key1 associated with the captured report
KEY2	VARCHAR2(128)		Key2 associated with the captured report
KEY3	VARCHAR2(128)		Key3 associated with the captured report
KEY4	VARCHAR2(256)		Key4 associated with the captured report
GENERATION_COST_SECONDS	NUMBER		Time taken to generate this report (in seconds)
REPORT_SUMMARY	VARCHAR2(4000)		Summary of this report
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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:

"DBA_HIST_REPORTS_DETAILS"

6.35 DBA_HIST_REPORTS_CONTROL

DBA_HIST_REPORTS_CONTROL contains control information about the report capture mechanism that automatically captures XML reports to Automatic Workload Repository (AWR).

Reports are captured automatically for components like SQL Monitor and Real-Time Automatic Database Diagnostic Monitor (Real-Time ADDM).

Column	Datatype	NULL	Description
DBID	NUMBER		Database ID of the current database for the report
EXECUTION_MODE	VARCHAR2(12)		Mode of execution of automatic report capture. Possible values: • REGULAR: Regular per-minute report capture subject to DBTIME budget • FULL_CAPTURE: Report capture will be run per minute without the DBTIME budget constraints and is provided to capture a more comprehensive set of reports **NOTE**: The FULL_CAPTURE mode can be started and ended respectively by executing the START_REPORT_CAPTURE and FINISH_REPORT_CAPTURE APIs in the DBMS_AUTO_REPORT package. At all other times, the execution mode should be REGULAR.

> **See Also:**
>
> *Oracle Database PL/SQL Packages and Types Reference* for more information about the DBMS_AUTO_REPORT package

6.36 DBA_HIST_REPORTS_DETAILS

DBA_HIST_REPORTS_DETAILS displays details about each report captured in Automatic Workload Repository (AWR).

Metadata for each report appears in the DBA_HIST_REPORTS view while the actual report is available in the DBA_HIST_REPORTS_DETAILS view.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER		ID of the first AWR snapshot that will be taken after this report is generated
DBID	NUMBER		Database ID of the current database for the report
INSTANCE_NUMBER	NUMBER		Instance number (for an Oracle RAC system)
REPORT_ID	NUMBER		ID of the captured report
SESSION_ID	NUMBER		ID of the session corresponding to the captured report (currently used only for SQL Monitor reports)
SESSION_SERIAL#	NUMBER		Session serial number relevant to this report (currently used only for SQL Monitor reports)

Column	Datatype	NULL	Description
GENERATION_TIME	DATE		Time when this report was generated
REPORT_COMPRESSED	BLOB		Actual XML report in compressed form
REPORT	CLOB		Full uncompressed report
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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:**
>
> "DBA_HIST_REPORTS"

6.37 DBA_HIST_REPORTS_TIMEBANDS

DBA_HIST_REPORTS_TIMEBANDS contains bands of time with a new row created every day corresponding to a band of time.

Each band of time represents a period of time, and has a row for every report captured automatically into Automatic Workload Repository (AWR) during that time. If the activity period of a report spans across two bands of time (for example. the activity started before midnight and ended after midnight), then the view contains two rows for that report, with one row for each band of time. The view is partitioned to provide fast access to all reports captured in a given time frame.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER		The AWR snapshot id corresponding to the report
DBID	NUMBER		Database ID of the current database for the report
INSTANCE_NUMBER	NUMBER		Instance number (for an Oracle RAC system)
CON_DBID	NUMBER		CDB ID of the captured report
COMPONENT_ID	NUMBER		ID of the component (for example, SQL Monitor) whose report is captured
COMPONENT_NAME	VARCHAR2(128)		Name of the component whose report is captured
BAND_START_TIME	DATE		Starting time of the time band
BAND_LENGTH	NUMBER		Length of time band in days (currently unused)
REPORT_ID	NUMBER		ID of the captured report
REPORT_GENERATION_TIME	DATE		Time when the report was generated
PERIOD_START_TIME	DATE		Time when the activity period started
PERIOD_END_TIME	DATE		Time when the activity period ended

Column	Datatype	NULL	Description
KEY1	VARCHAR2(128)		Key1 associated with the captured report
KEY2	VARCHAR2(128)		Key2 associated with the captured report
KEY3	VARCHAR2(128)		Key3 associated with the captured report
KEY4	VARCHAR2(256)		Key4 associated with the captured report
SESSION_ID	NUMBER		ID of the session corresponding to the captured report (currently used only for SQL Monitor reports)
SESSION_SERIAL#	NUMBER		Session serial number corresponding to the captured report (currently used only for SQL Monitor reports)
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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

6.38 DBA_HIST_RESOURCE_LIMIT

DBA_HIST_RESOURCE_LIMIT displays historical information about global resource use for some of the system resource.

This view contains snapshots of V$RESOURCE_LIMIT.

If time is of interest, join this view with DBA_HIST_SNAPSHOT.END_INTERVAL_TIME.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
RESOURCE_NAME	VARCHAR2(30)	NOT NULL	Name of the resource
CURRENT_UTILIZATION	NUMBER		Number of (resources, locks, or processes) currently being used
MAX_UTILIZATION	NUMBER		Maximum consumption of the 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_DBID	NUMBER		The database ID of the PDB for the sampled session

Column	Datatype	NULL	Description
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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

6.39 DBA_HIST_ROWCACHE_SUMMARY

DBA_HIST_ROWCACHE_SUMMARY displays historical summary statistics for data dictionary activity.

This view contains snapshots of V$ROWCACHE.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
PARAMETER	VARCHAR2(32)	NOT NULL	Name of the initialization parameter that determines the number of entries in the data dictionary cache
TOTAL_USAGE	NUMBER		Sum of the total number of entries in the cache
USAGE	NUMBER		Number of cache entries that contain valid data
GETS	NUMBER		Total number of requests for information on the data object
GETMISSES	NUMBER		Number of data requests resulting in cache misses
SCANS	NUMBER		Number of scan requests
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
DLM_RELEASES	NUMBER		Number of DLM releases
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$ROWCACHE"

6.40 DBA_HIST_RSRC_CONSUMER_GROUP

DBA_HIST_RSRC_CONSUMER_GROUP displays historical information about Resource Manager consumer groups.

This view contains snapshots of V$RSRC_CONS_GROUP_HISTORY.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
SEQUENCE#	NUMBER	NOT NULL	A sequential counter that uniquely describes the DBA_HIST_RSRC_PLAN entry to which these consumer group statistics apply. When the instance is restarted, this value is reset to zero.
CONSUMER_GROUP_ID	NUMBER	NOT NULL	Consumer group object ID (a unique number, consistent across database shutdowns and startups)
CONSUMER_GROUP_NAME	VARCHAR2(128)	NOT NULL	Name of the consumer group
REQUESTS	NUMBER	NOT NULL	Cumulative number of requests that were executed in the consumer group
CPU_WAIT_TIME	NUMBER	NOT NULL	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.
CPU_WAITS	NUMBER	NOT NULL	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.
CONSUMED_CPU_TIME	NUMBER	NOT NULL	Cumulative amount of CPU time consumed by all sessions in the consumer group (in milliseconds)
YIELDS	NUMBER	NOT NULL	Cumulative number of times that sessions in the consumer group had to yield CPU to other sessions because of quantum expiration
ACTIVE_SESS_LIMIT_HIT	NUMBER	NOT NULL	Number of times that sessions in the consumer group were queued because the consumer group reached its active session limit
UNDO_LIMIT_HIT	NUMBER	NOT NULL	Number of times that queries in the consumer group were cancelled because the consumer group reached its UNDO_POOL limit
SWITCHES_IN_CPU_TIME	NUMBER	NOT NULL	Number of switches into the consumer group because of the Resource Manager plan's SWITCH_TIME limit
SWITCHES_OUT_CPU_TIME	NUMBER	NOT NULL	Number of switches out of the consumer group because of the Resource Manager plan's SWITCH_TIME limit

Column	Datatype	NULL	Description
SWITCHES_IN_IO_MEGABYTES	NUMBER	NOT NULL	Number of switches into the consumer group because of the Resource Manager plan's SWITCH_IO_MEGABYTES limit
SWITCHES_OUT_IO_MEGABYTES	NUMBER	NOT NULL	Number of switches out of the consumer group because of the Resource Manager plan's SWITCH_IO_MEGABYTES limit
SWITCHES_IN_IO_REQUESTS	NUMBER	NOT NULL	Number of switches into the consumer group because of the Resource Manager plan's SWITCH_IO_REQS limit
SWITCHES_OUT_IO_REQUESTS	NUMBER	NOT NULL	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
PGA_LIMIT_SESSIONS_KILLED	NUMBER		The number of sessions that were terminated because their PGA allocation exceeded the PGA limit specified in the Resource Plan's SESSION_PGA_LIMIT directive
SQL_CANCELED	NUMBER	NOT NULL	Number of times that SQL queries running in the consumer group were terminated because they exceeded the Resource Manager plan's SWITCH_TIME limit and CANCEL_SQL was specified as the Resource Manager plan's SWITCH_GROUP
ACTIVE_SESS_KILLED	NUMBER	NOT NULL	Number of times that sessions running in the consumer group were terminated because they exceeded the Resource Manager plan's SWITCH_TIME limit and KILL_SESSION was specified as the Resource Manager plan's SWITCH_GROUP
IDLE_SESS_KILLED	NUMBER	NOT NULL	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	NOT NULL	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	NOT NULL	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	NOT NULL	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	NOT NULL	Cumulative I/O wait time (in milliseconds)

Column	Datatype	NULL	Description
IO_SERVICE_WAITS	NUMBER	NOT NULL	Total number of wait requests
SMALL_READ_MEGABYTES	NUMBER	NOT NULL	Number of single block megabytes read
SMALL_WRITE_MEGABYTES	NUMBER	NOT NULL	Number of single block megabytes written
LARGE_READ_MEGABYTES	NUMBER	NOT NULL	Number of multiblock megabytes read
LARGE_WRITE_MEGABYTES	NUMBER	NOT NULL	Number of multiblock megabytes written
SMALL_READ_REQUESTS	NUMBER	NOT NULL	Number of single block read requests
SMALL_WRITE_REQUESTS	NUMBER	NOT NULL	Number of single block write requests
LARGE_READ_REQUESTS	NUMBER	NOT NULL	Number of multiblock read requests
LARGE_WRITE_REQUESTS	NUMBER	NOT NULL	Number of multiblock write requests
PQS_QUEUED	NUMBER		Number of times that sessions in the consumer group were queued when trying to run parallel statements
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
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)
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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"

6.41 DBA_HIST_RSRC_METRIC

DBA_HIST_RSRC_METRIC displays information about historical Resource Manager metrics for the past hour.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
BEGIN_TIME	TIMESTAMP(3)	NOT NULL	Begin time of the interval
END_TIME	TIMESTAMP(3)	NOT NULL	End time of the interval
INTSIZE_CSEC	NUMBER	NOT NULL	Interval size (in hundredths of a second)
SEQUENCE#	NUMBER	NOT NULL	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	NOT NULL	Consumer group object ID (a unique number, consistent across database shutdowns and startups)
CPU_CONSUMED_TIME	NUMBER	NOT NULL	Cumulative amount of CPU time consumed by all sessions in the consumer group, in milliseconds
CPU_WAIT_TIME	NUMBER	NOT NULL	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.
AVG_RUNNING_SESSIONS	NUMBER		Average number of sessions in the consumer group that are currently running
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.
RUNNING_SESSION_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_CPU_UTILIZATION	NUMBER		Average percentage of CPU consumed by the consumer group, with respect to the total number of CPUs in the system
IO_REQUESTS	NUMBER	NOT NULL	I/O requests
IO_MEGABYTES	NUMBER	NOT NULL	I/O megabytes
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
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

Column	Datatype	NULL	Description
AVG_QUEUED_PARALLEL_SERVERS	NUMBER		The average number of parallel servers that were requested by queued parallel statements during the 1-minute metric window
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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

6.42 DBA_HIST_RSRC_PDB_METRIC

DBA_HIST_RSRC_PDB_METRIC displays information about historical Resource Manager metrics for the past hour by PDB.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
BEGIN_TIME	TIMESTAMP(3)	NOT NULL	Begin time of the interval
END_TIME	TIMESTAMP(3)	NOT NULL	End time of the interval
INTSIZE_CSEC	NUMBER	NOT NULL	Interval size (in hundredths of a second)
SEQUENCE#	NUMBER	NOT NULL	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	NOT NULL	Cumulative amount of CPU time consumed by all sessions in the PDB, in milliseconds
CPU_WAIT_TIME	NUMBER	NOT NULL	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.
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.
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

Column	Datatype	NULL	Description
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
SGA_BYTES	NUMBER	NOT NULL	The current SGA usage for this PDB in bytes
BUFFER_CACHE_BYTES	NUMBER	NOT NULL	The current usage of the buffer cache by this PDB in bytes
SHARED_POOL_BYTES	NUMBER	NOT NULL	The current usage of the shared pool by this PDB in bytes
PGA_BYTES	NUMBER	NOT NULL	The current PGA usage for this PDB in bytes
PLAN_ID	NUMBER	NOT NULL	Resource Manager plan identifier
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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

Column	Datatype	NULL	Description
MAX_AVG_RUNNING_SESSIONS	NUMBER	NOT NULL	Maximum value of AVG_RUNNING_SESSIONS during the last hour

Note:

- "MAX_IOPS"
- "MAX_MBPS"

6.43 DBA_HIST_RSRC_PLAN

DBA_HIST_RSRC_PLAN displays historical information about resource plans.

This view contains snapshots of V$RSRC_PLAN_HISTORY.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
SEQUENCE#	NUMBER	NOT NULL	A sequential counter that uniquely describes a row. When the instance is restarted, this value is reset to zero.
START_TIME	DATE	NOT NULL	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
PLAN_ID	NUMBER	NOT NULL	Resource plan ID; NULL if the Resource Manager was disabled
PLAN_NAME	VARCHAR2(128)	NOT NULL	Resource plan name; NULL if the Resource Manager was disabled
CPU_MANAGED	VARCHAR2(4)	NOT NULL	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)

Column	Datatype	NULL	Description
PARALLEL_EXECUTION_MANAGED	VARCHAR2(4)		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.
INSTANCE_CAGING	VARCHAR2(4)		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.
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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_PLAN_HISTORY"

6.44 DBA_HIST_RULE_SET

DBA_HIST_RULE_SET displays historical information about rule set statistics.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
OWNER	VARCHAR2(128)	NOT NULL	Owner of the rule set
NAME	VARCHAR2(128)	NOT NULL	Name of the rule set
STARTUP_TIME	DATE	NOT NULL	Startup time of the instance

Column	Datatype	NULL	Description
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
EVALUATIONS	NUMBER		Number of evaluations on the rule set
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
RELOADS	NUMBER		Number of times the rule set object was reloaded in shared memory
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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

6.45 DBA_HIST_SAGAS

DBA_HIST_SAGAS provides a history of all completed sagas in the database. Its columns are the same as those in ALL_HIST_SAGAS.

This view displays sagas that were either initiated in the current PDB or joined by participants in the current PDB.

Completed sagas are retained in this view for the length of time specified by the SAGA_HIST_RETENTION initialization parameter. The default is 30 days.

Note:

This view is available starting with Oracle Database 23ai.

See Also:

- "ALL_HIST_SAGAS"
- "SAGA_HIST_RETENTION"

6.46 DBA_HIST_SEG_STAT

DBA_HIST_SEG_STAT displays historical information about segment-level statistics.

This view captures the top segments based on a set of criteria and captures information from V$SEGSTAT. The total value is the value of the statistics since instance startup. The delta value is the value of the statistics from the BEGIN_INTERVAL_TIME to the END_INTERVAL_TIME in the DBA_HIST_SNAPSHOT view.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
TS#	NUMBER	NOT NULL	Tablespace number
OBJ#	NUMBER	NOT NULL	Dictionary object number
DATAOBJ#	NUMBER	NOT NULL	Data object number
LOGICAL_READS_TOTAL	NUMBER		Cumulative value for logical reads
LOGICAL_READS_DELTA	NUMBER		Delta value for logical reads
BUFFER_BUSY_WAITS_TOTAL	NUMBER		Cumulative value for buffer busy waits
BUFFER_BUSY_WAITS_DELTA	NUMBER		Delta value for buffer busy waits
DB_BLOCK_CHANGES_TOTAL	NUMBER		Cumulative value for db block changes (in blocks)
DB_BLOCK_CHANGES_DELTA	NUMBER		Delta value for db block changes (in blocks)
PHYSICAL_READS_TOTAL	NUMBER		Cumulative value for physical reads (in blocks)
PHYSICAL_READS_DELTA	NUMBER		Delta value for physical reads (in blocks)
PHYSICAL_WRITES_TOTAL	NUMBER		Cumulative value for physical writes (in blocks)
PHYSICAL_WRITES_DELTA	NUMBER		Delta value for physical writes (in blocks)
PHYSICAL_READS_DIRECT_TOTAL	NUMBER		Cumulative value for physical reads direct (in blocks)
PHYSICAL_READS_DIRECT_DELTA	NUMBER		Delta value for physical reads direct (in blocks)
PHYSICAL_WRITES_DIRECT_TOTAL	NUMBER		Cumulative value for physical writes direct (in blocks)
PHYSICAL_WRITES_DIRECT_DELTA	NUMBER		Delta value for physical writes direct (in blocks)
ITL_WAITS_TOTAL	NUMBER		Cumulative value for ITL waits
ITL_WAITS_DELTA	NUMBER		Delta value for ITL waits
ROW_LOCK_WAITS_TOTAL	NUMBER		Cumulative value for row lock waits
ROW_LOCK_WAITS_DELTA	NUMBER		Delta value for row lock waits
GC_CR_BLOCKS_SERVED_TOTAL	NUMBER		Cumulative value for global cache CR blocks served
GC_CR_BLOCKS_SERVED_DELTA	NUMBER		Delta value for global cache CR blocks served
GC_CU_BLOCKS_SERVED_TOTAL	NUMBER		Cumulative value for global cache current blocks served
GC_CU_BLOCKS_SERVED_DELTA	NUMBER		Delta value for global cache current blocks served

Column	Datatype	NULL	Description
GC_BUFFER_BUSY_TOTAL	NUMBER		Cumulative value for global cache buffer busy
GC_BUFFER_BUSY_DELTA	NUMBER		Delta value for global cache buffer busy
GC_CR_BLOCKS_RECEIVED_TOTAL	NUMBER		Cumulative value for global cache CR blocks received
GC_CR_BLOCKS_RECEIVED_DELTA	NUMBER		Delta value for global cache CR blocks received
GC_CU_BLOCKS_RECEIVED_TOTAL	NUMBER		Cumulative value for global cache current blocks received
GC_CU_BLOCKS_RECEIVED_DELTA	NUMBER		Delta value for global cache current blocks received
SPACE_USED_TOTAL	NUMBER		Number of bytes used by user data
SPACE_USED_DELTA	NUMBER		Delta value for space used by user data (in bytes). A negative value indicates the number of bytes deleted in the segment.
SPACE_ALLOCATED_TOTAL	NUMBER		The number of bytes that are allocated
SPACE_ALLOCATED_DELTA	NUMBER		Delta value for the space allocated (in bytes). A negative value indicates the number of bytes deallocated to the tablespace.
TABLE_SCANS_TOTAL	NUMBER		Cumulative value for table scans
TABLE_SCANS_DELTA	NUMBER		Delta value for table scans
CHAIN_ROW_EXCESS_TOTAL	NUMBER		Cumulative value of number of chained row pieces that can be eliminated by table reorganization
CHAIN_ROW_EXCESS_DELTA	NUMBER		Delta value of number of chained row pieces that can be eliminated by table reorganization
PHYSICAL_READ_REQUESTS_TOTAL	NUMBER		Cumulative value of number of physical read I/O requests issued for the monitored segment
PHYSICAL_READ_REQUESTS_DELTA	NUMBER		Delta value of number of physical read I/O requests issued for the monitored segment
PHYSICAL_WRITE_REQUESTS_TOTAL	NUMBER		Cumulative value of number of physical write I/O requests issued for the monitored segment
PHYSICAL_WRITE_REQUESTS_DELTA	NUMBER		Delta value of number of physical write I/O requests issued for the monitored segment
OPTIMIZED_PHYSICAL_READS_TOTAL	NUMBER		Cumulative value of number of physical reads from Database Smart Flash Cache for the monitored segment
OPTIMIZED_PHYSICAL_READS_DELTA	NUMBER		Delta value of number of physical reads from Database Smart Flash Cache for the monitored segment
IM_SCANS_TOTAL	NUMBER		Count of segment statistics
IM_SCANS_DELTA	NUMBER		Delta values for in-memory scans
POPULATE_CUS_TOTAL	NUMBER		Count of compression units (CUs) populated per segment
POPULATE_CUS_DELTA	NUMBER		Delta value for compression unit (CU) populate operations
REPOPULATE_CUS_TOTAL	NUMBER		Count of CUs repopulated per segment
REPOPULATE_CUS_DELTA	NUMBER		Delta value for compression unit (CU) repopulate operations

Column	Datatype	NULL	Description
IM_DB_BLOCK_CHANGES_TOTAL	NUMBER		The total number of changes that were part of an update or delete operation that were made to segment blocks
IM_DB_BLOCK_CHANGES_DELTA	NUMBER		Delta value for database block changes
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$SEGSTAT"
- "DBA_HIST_SNAPSHOT"

6.47 DBA_HIST_SEG_STAT_OBJ

DBA_HIST_SEG_STAT_OBJ displays all the names of the segments captured in the workload repository.

This view is used with the DBA_HIST_SEG_STAT view.

Column	Datatype	NULL	Description
DBID	NUMBER	NOT NULL	Database ID
TS#	NUMBER	NOT NULL	Tablespace number
OBJ#	NUMBER	NOT NULL	Dictionary object number
DATAOBJ#	NUMBER	NOT NULL	Data object number
OWNER	VARCHAR2(128)	NOT NULL	Owner of the object
OBJECT_NAME	VARCHAR2(128)	NOT NULL	Name of the object
SUBOBJECT_NAME	VARCHAR2(128)		Name of the subobject (for example: partition)
OBJECT_TYPE	VARCHAR2(18)		Type of the object for example: table, tablespace)
TABLESPACE_NAME	VARCHAR2(30)		Tablespace Name for the object
PARTITION_TYPE	VARCHAR2(8)		Partition Type, if relevant
CON_DBID	NUMBER		The database ID of the PDB for the sampled session

Column	Datatype	NULL	Description
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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:**
>
> "DBA_HIST_SEG_STAT"

6.48 DBA_HIST_SERVICE_NAME

DBA_HIST_SERVICE_NAME displays the names of the Services tracked by the Workload Repository.

This view contains information for V$SERVICES.

Column	Datatype	NULL	Description
DBID	NUMBER	NOT NULL	Database ID for the snapshot
SERVICE_NAME_HASH	NUMBER	NOT NULL	Hash of the service name
SERVICE_NAME	VARCHAR2(64)	NOT NULL	Name of the service
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$SERVICES"

6.49 DBA_HIST_SERVICE_STAT

DBA_HIST_SERVICE_STAT displays the history of important service statistics tracked by the Workload Repository.

The call rate statistics in this view can be used for making run-time routing decisions, for tracking service levels, and for per-instance diagnostics per call rate.

The elapsed timing for each call provides a relative value across instances for how well a node is processing SQL calls issued under a service name. When aggregation is enabled for the service name, this view provides historical data on the timing and work done for calls issued for the whole service. This view contains information from V$SERVICE_STATS.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
SERVICE_NAME_HASH	NUMBER	NOT NULL	Hash of the service name
SERVICE_NAME	VARCHAR2(64)	NOT NULL	Name of the service
STAT_ID	NUMBER	NOT NULL	Statistic identifier
STAT_NAME	VARCHAR2(64)	NOT NULL	Statistic name
VALUE	NUMBER		Value of the statistic
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$SERVICE_STATS"

6.50 DBA_HIST_SERVICE_WAIT_CLASS

DBA_HIST_SERVICE_WAIT_CLASS displays the history of wait class information for services as tracked by the Workload Repository.

This view contains information from V$SERVICE_WAIT_CLASS.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID

Column	Datatype	NULL	Description
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
SERVICE_NAME_HASH	NUMBER	NOT NULL	Hash of the service name
SERVICE_NAME	VARCHAR2(64)	NOT NULL	Name of the service
WAIT_CLASS_ID	NUMBER	NOT NULL	Identifier for the class of the wait event
WAIT_CLASS	VARCHAR2(64)		Name for the class of the wait event
TOTAL_WAITS	NUMBER		Total number of waits for this event
TIME_WAITED	NUMBER		Total amount of time waited for this event (in hundredths of a second)
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$SERVICE_WAIT_CLASS"

6.51 DBA_HIST_SESS_SGA_STATS

DBA_HIST_SESS_SGA_STATS shows usage statistics for high utilization Oracle GoldenGate and XStream sessions.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
OBJECT_NAME	VARCHAR2(128)	NOT NULL	Object name
SESSION_TYPE	VARCHAR2(64)	NOT NULL	Type of session
SESSION_MODULE	VARCHAR2(64)	NOT NULL	Session module. Valid values: • XStream • GoldenGate
SGA_USED	NUMBER		The total amount of shared memory (in bytes) currently used by the session out of the amount allocated (SGA_ALLOCATED)
SGA_ALLOCATED	NUMBER		The total amount of shared memory (in bytes) allocated from the pool for the session

Column	Datatype	NULL	Description
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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

6.52 DBA_HIST_SESS_TIME_STATS

DBA_HIST_SESS_TIME_STATS displays information about CPU and I/O time for interesting Oracle GoldenGate and XStream sessions.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
SESSION_TYPE	VARCHAR2(64)	NOT NULL	Type of session
MIN_LOGON_TIME	DATE		Minimum logon time
SUM_CPU_TIME	NUMBER		Total CPU time
SUM_SYS_IO_WAIT	NUMBER		Total system I/O wait time
SUM_USER_IO_WAIT	NUMBER		Total user I/O wait time
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
SESSION_MODULE	VARCHAR2(64)	NOT NULL	Session module. Valid values: • XStream • GoldenGate
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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

6.53 DBA_HIST_SESSMETRIC_HISTORY

DBA_HIST_SESSMETRIC_HISTORY displays the history of several important session metrics.

> **Note:**
>
> This view is populated only if a session metric exceeds a server metric threshold that was configured using the DBMS_SERVER_ALERT package.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
BEGIN_TIME	DATE	NOT NULL	Begin time of the interval
END_TIME	DATE	NOT NULL	End time of the interval
SESSID	NUMBER	NOT NULL	Session ID
SERIAL#	NUMBER	NOT NULL	Session serial number
INTSIZE	NUMBER	NOT NULL	Interval size (in hundredths of a second)
GROUP_ID	NUMBER	NOT NULL	Group ID
METRIC_ID	NUMBER	NOT NULL	Metric ID
METRIC_NAME	VARCHAR2(64)	NOT NULL	Metric name
VALUE	NUMBER	NOT NULL	Metric Value
METRIC_UNIT	VARCHAR2(64)	NOT NULL	Unit of measurement
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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:**
>
> The DBMS_SERVER_ALERT package in *Oracle Database PL/SQL Packages and Types Reference*

6.54 DBA_HIST_SGA

DBA_HIST_SGA displays historical summary information about the system global area (SGA).

This view contains snapshots of V$SGA.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
NAME	VARCHAR2(64)	NOT NULL	SGA component group
VALUE	NUMBER	NOT NULL	Memory size (in bytes)
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$SGA"

6.55 DBA_HIST_SGA_TARGET_ADVICE

DBA_HIST_SGA_TARGET_ADVICE provides historical information about the SGA_TARGET initialization parameter.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
SGA_SIZE	NUMBER	NOT NULL	Size of the SGA
SGA_SIZE_FACTOR	NUMBER	NOT NULL	Ratio between the SGA_SIZE and the current size of the SGA
ESTD_DB_TIME	NUMBER	NOT NULL	Estimated DB_TIME for this SGA_SIZE
ESTD_PHYSICAL_READS	NUMBER		Estimated number of physical reads
CON_DBID	NUMBER		The database ID of the PDB for the sampled session

Column	Datatype	NULL	Description
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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:**
>
> "SGA_TARGET"

6.56 DBA_HIST_SGASTAT

DBA_HIST_SGASTAT displays detailed historical information on the system global area (SGA).

This view contains snapshots of V$SGASTAT.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
NAME	VARCHAR2(64)		SGA component group
POOL	VARCHAR2(30)		Designates the pool in which the memory in NAME resides: • in-memory pool - Memory is allocated from the In-Memory pool • java pool - Memory is allocated from the Java pool • large pool - Memory is allocated from the large pool • numa pool - Memory is allocated from the NUMA pool • shared pool - Memory is allocated from the shared pool • streams pool - Memory is allocated from the Streams pool
BYTES	NUMBER		Memory size (in bytes)
CON_DBID	NUMBER		The database ID of the PDB for the sampled session

Column	Datatype	NULL	Description
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$SGASTAT"

6.57 DBA_HIST_SHARED_POOL_ADVICE

DBA_HIST_SHARED_POOL_ADVICE displays historical information about estimated parse time in the shared pool for different pool sizes.

This view contains snapshots of V$SHARED_POOL_ADVICE.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
SHARED_POOL_SIZE_FOR_ESTIMATE	NUMBER	NOT NULL	Shared pool size for the estimate (in megabytes)
SHARED_POOL_SIZE_FACTOR	NUMBER		Size factor with respect to the current shared 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 shared 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 shared pool of the specified size. This is the time that would have been spent in reloading the required objects in the shared 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 shared pool size
ESTD_LC_LOAD_TIME	NUMBER		Estimated elapsed time (in seconds) for parsing in a shared pool of the specified size.
ESTD_LC_LOAD_TIME_FACTOR	NUMBER		Estimated load time factor with respect to the current shared pool size
ESTD_LC_MEMORY_OBJECT_HITS	NUMBER		Estimated number of times a library cache memory object was found in a shared pool of the specified size

Column	Datatype	NULL	Description
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$SHARED_POOL_ADVICE"

6.58 DBA_HIST_SHARED_SERVER_SUMMARY

DBA_HIST_SHARED_SERVER_SUMMARY displays historical information for shared servers.

This includes information about shared server activity, the servers, common queues, and dispatcher queues. This view obtains information from V$SHARED_SERVER, V$DISPATCHER, V$CIRCUIT, and V$QUEUE, and is aggregated over all servers, dispatchers, queues, and circuits.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
NUM_SAMPLES	NUMBER		Total number of samples
SAMPLE_TIME	NUMBER		Last sample timestamp
SAMPLED_TOTAL_CONN	NUMBER		Cumulative sum of total number of connections over all samples. To determine the average number of connections between two snapshots, divide the difference in SAMPLED_TOTAL_CONN by the difference in NUM_SAMPLES.
SAMPLED_ACTIVE_CONN	NUMBER		Cumulative sum of active number of connections over all samples. To determine the average number of active connections between two snapshots, divide the difference in SAMPLED_ACTIVE_CONN by the difference in NUM_SAMPLES.
SAMPLED_TOTAL_SRV	NUMBER		Cumulative sum of total number of servers over all samples. To determine the average number of servers between two snapshots, divide the difference in SAMPLED_TOTAL_SRV by the difference in NUM_SAMPLES.

Column	Datatype	NULL	Description
SAMPLED_ACTIVE_SRV	NUMBER		Cumulative sum of active number of servers over all samples. To determine the average number of active servers between two snapshots, divide the difference in SAMPLED_ACTIVE_SRV by the difference in NUM_SAMPLES.
SAMPLED_TOTAL_DISP	NUMBER		Cumulative sum of total number of dispatchers over all samples. To determine the average number of dispatchers between two snapshots, divide the difference in SAMPLED_TOTAL_DISP by the difference in NUM_SAMPLES.
SAMPLED_ACTIVE_DISP	NUMBER		Cumulative sum of active number of dispatchers over all samples. To determine the average number of active dispatchers between two snapshots, divide the difference in SAMPLED_ACTIVE_DISP by the difference in NUM_SAMPLES.
SRV_BUSY	NUMBER		Total shared server busy time (in hundredths of a second)
SRV_IDLE	NUMBER		Total shared server idle time (in hundredths of a second)
SRV_IN_NET	NUMBER		Total shared server incoming network wait time (in hundredths of a second). This includes waits for receives and resets. This time is also included in SRV_BUSY.
SRV_OUT_NET	NUMBER		Total shared server outgoing network wait time (in hundredths of a second). This includes waits for sends and outbound connection requests. This time is also included in SRV_BUSY.
SRV_MESSAGES	NUMBER		Number of messages processed
SRV_BYTES	NUMBER		Total number of bytes in all messages
CQ_WAIT	NUMBER		Total time that all items in the common queue have waited (in hundredths of a second)
CQ_TOTALQ	NUMBER		Total number of items that have ever been in the common queue
DQ_TOTALQ	NUMBER		Total number of items that have ever been in a dispatcher queue
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$SHARED_SERVER"
> - "V$DISPATCHER"
> - "V$CIRCUIT"
> - "V$QUEUE"

6.59 DBA_HIST_SNAP_ERROR

DBA_HIST_SNAP_ERROR displays information about the snapshot error information in the Workload Repository.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
TABLE_NAME	VARCHAR2(128)	NOT NULL	Name of the table in which the error occurred
ERROR_NUMBER	NUMBER	NOT NULL	Error number for the error encountered
STEP_ID	NUMBER		For internal use only
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

6.60 DBA_HIST_SNAPSHOT

DBA_HIST_SNAPSHOT displays information about the snapshots in the Workload Repository.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
STARTUP_TIME	TIMESTAMP(3)	NOT NULL	Startup time of the instance
BEGIN_INTERVAL_TIME	TIMESTAMP(3)	NOT NULL	Time at the beginning of the snapshot interval
END_INTERVAL_TIME	TIMESTAMP(3)	NOT NULL	Time at the end of the snapshot interval; the actual time the snapshot was taken
FLUSH_ELAPSED	INTERVAL DAY(5) TO SECOND(1)		Amount of time to perform the snapshot

Column	Datatype	NULL	Description
SNAP_LEVEL	NUMBER		Snapshot level
ERROR_COUNT	NUMBER		Number of errors occurring in the tables for the particular snapshot
SNAP_FLAG	NUMBER		Condition under which the snapshot was inserted. Possible values are: 0 - Snapshot was taken automatically by the Manageability Monitor Process (MMON process) 1 - Manual snapshot created using a PL/SQL package 2 - Imported snapshot 4 - Snapshot taken while Diagnostic Pack or Tuning Pack was not enabled
SNAP_TIMEZONE	INTERVAL DAY(0) TO SECOND(0)		Snapshot time zone expressed as offset from UTC (Coordinated Universal Time) time zone
BEGIN_INTERVAL_TIME_TZ	TIMESTAMP(3) WITH TIME ZONE		Time at the beginning of the snapshot interval, with timezone
END_INTERVAL_TIME_TZ	TIMESTAMP(3) WITH TIME ZONE		Time at the end of the snapshot interval; the actual time the snapshot was taken, with timezone
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:**
>
> Table F-1 for more information about the MMON process

6.61 DBA_HIST_SQL_BIND_METADATA

DBA_HIST_SQL_BIND_METADATA displays historical information on metadata for bind variables used by SQL cursors.

Column	Datatype	NULL	Description
DBID	NUMBER	NOT NULL	Database ID for the snapshot
SQL_ID	VARCHAR2(13)	NOT NULL	SQL identifier of the parent cursor in the library cache
NAME	VARCHAR2(128)		Name of the bind variable
POSITION	NUMBER	NOT NULL	Position of the bind variable in the SQL statement
DUP_POSITION	NUMBER		If the binding is performed by name and the bind variable is duplicated, then this column gives the position of the primary bind variable

Column	Datatype	NULL	Description
DATATYPE	NUMBER		Internal identifier for the bind data type. Beginning in Oracle Database 12*c*, a number representing a PL/SQL data type can appear in this column.
DATATYPE_STRING	VARCHAR2(15)		Textual representation of the bind data type. Beginning in Oracle Database 12*c*, a text representation of a PL/SQL-only data type can appear in this column. If the actual data type is a PL/SQL sub type, the name of the data type, not the sub type will be displayed.
CHARACTER_SID	NUMBER		National character set identifier
PRECISION	NUMBER		Precision (for numeric binds)
SCALE	NUMBER		Scale (for numeric binds)
MAX_LENGTH	NUMBER		Maximum bind length
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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

6.62 DBA_HIST_SQL_PLAN

DBA_HIST_SQL_PLAN displays the execution plan information for each child cursor in the workload repository.

This view captures information from V$SQL_PLAN and is used with the DBA_HIST_SQLSTAT view.

Column	Datatype	NULL	Description
DBID	NUMBER	NOT NULL	Database ID
SQL_ID	VARCHAR2(13)	NOT NULL	SQL identifier of the parent cursor in the library cache
PLAN_HASH_VALUE	NUMBER	NOT NULL	Numerical representation of the SQL plan for the cursor. Comparing one PLAN_HASH_VALUE to another easily identifies whether or not two plans are the same (rather than comparing the two plans line by line).
ID	NUMBER	NOT NULL	A number assigned to each step in the execution plan
OPERATION	VARCHAR2(30)		Name of the internal operation performed in this step (for example, TABLE ACCESS)
OPTIONS	VARCHAR2(30)		A variation on the operation described in the OPERATION column (for example, FULL)
OBJECT_NODE	VARCHAR2(128)		Name of the database link used to reference the object (a table name or view name). For local queries that use parallel execution, this column describes the order in which output from operations is consumed.
OBJECT#	NUMBER		Object number of the table or the index

Column	Datatype	NULL	Description
OBJECT_OWNER	VARCHAR2(128)		Name of the user who owns the schema containing the table or index
OBJECT_NAME	VARCHAR2(128)		Name of the table or index
OBJECT_ALIAS	VARCHAR2(261)		Alias for the object
OBJECT_TYPE	VARCHAR2(20)		Type of the object
OPTIMIZER	VARCHAR2(20)		Current mode of the optimizer for the first row in the plan (statement line), for example, ALL_ROWS. When the operation is a database access (for example, TABLE ACCESS), this column indicates whether or not the object is analyzed.
PARENT_ID	NUMBER		ID of the next execution step that operates on the output of the current step
DEPTH	NUMBER		Depth (or level) of the operation in the tree. It is not necessary to issue a CONNECT BY statement to get the level information, which is generally used to indent the rows from the PLAN_TABLE table. The root operation (statement) is level 0.
POSITION	NUMBER		Order of processing for all operations that have the same PARENT_ID
SEARCH_COLUMNS	NUMBER		Number of index columns with start and stop keys (that is, the number of columns with matching predicates)
COST	NUMBER		Cost of the operation as estimated by the optimizer's cost-based approach. For statements that use the rule-based approach, this column is null.
CARDINALITY	NUMBER		Estimate, by the cost-based optimizer, of the number of rows produced by the operation
BYTES	NUMBER		Estimate, by the cost-based optimizer, of the number of bytes produced by the operation
OTHER_TAG	VARCHAR2(35)		Describes the contents of the OTHER column. See EXPLAIN PLAN for values.
PARTITION_START	VARCHAR2(64)		Start partition of a range of accessed partitions
PARTITION_STOP	VARCHAR2(64)		Stop partition of a range of accessed partitions
PARTITION_ID	NUMBER		Step that computes the pair of values of the PARTITION_START and PARTITION_STOP columns
OTHER	VARCHAR2(4000)		Other information specific to the execution step that users may find useful. See EXPLAIN PLAN for values.
DISTRIBUTION	VARCHAR2(20)		Stores the method used to distribute rows from producer query servers to consumer query servers
CPU_COST	NUMBER		CPU cost of the operation as estimated by the optimizer's cost-based approach. For statements that use the rule-based approach, this column is null.
IO_COST	NUMBER		I/O cost of the operation as estimated by the optimizer's cost-based approach. For statements that use the rule-based approach, this column is null.
TEMP_SPACE	NUMBER		Temporary space usage of the operation (sort or hash-join) as estimated by the optimizer's cost-based approach. For statements that use the rule-based approach, this column is null.

Column	Datatype	NULL	Description
ACCESS_PREDICATES	VARCHAR2(4000)		Predicates used to locate rows in an access structure. For example, start or stop predicates for an index range scan.
FILTER_PREDICATES	VARCHAR2(4000)		Predicates used to filter rows before producing them
PROJECTION	VARCHAR2(4000)		Expressions produced by the operation
TIME	NUMBER		Elapsed time (in seconds) of the operation as estimated by the optimizer's cost-based approach. For statements that use the rule-based approach, this column is null.
QBLOCK_NAME	VARCHAR2(128)		Name of the query block
REMARKS	VARCHAR2(4000)		Remarks
TIMESTAMP	DATE		Timestamp for when the plan was produced
OTHER_XML	CLOB		Provides extra information specific to an execution step of the execution plan. The content of this column is structured using XML, which allows multiple pieces of information to be stored, including the following: • Name of the schema against which the query was parsed • Release number of the Oracle Database that produced the explain plan • Hash value associated with the execution plan • Name (if any) of the outline or the SQL profile used to build the execution plan • Indication of whether or not dynamic statistics were used to produce the plan • The outline data, a set of optimizer hints that can be used to regenerate the same plan • Additional data that describes the relationship between rows in the plan table and subplans of adaptive plans
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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_PLAN"
- "DBA_HIST_SQLSTAT"

6.63 DBA_HIST_SQL_SUMMARY

DBA_HIST_SQL_SUMMARY displays historical SQL summary information.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
TOTAL_SQL	NUMBER	NOT NULL	Total number of SQLs
TOTAL_SQL_MEM	NUMBER	NOT NULL	Total sharable memory in bytes for SQLs
SINGLE_USE_SQL	NUMBER	NOT NULL	Total number of single execution SQLs
SINGLE_USE_SQL_MEM	NUMBER	NOT NULL	Total sharable memory in bytes for single execution SQLs
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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

6.64 DBA_HIST_SQL_WORKAREA_HSTGRM

DBA_HIST_SQL_WORKAREA_HSTGRM displays the historical cumulative work area execution statistics (cumulated since instance startup) for different work area groups.

This view contains snapshots of V$SQL_WORKAREA_HISTOGRAM.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
LOW_OPTIMAL_SIZE	NUMBER	NOT NULL	Lower bound for the optimal memory requirement of work areas included in the row (in bytes)
HIGH_OPTIMAL_SIZE	NUMBER	NOT NULL	Upper bound for the optimal memory requirement of work areas included in the row (in bytes)
OPTIMAL_EXECUTIONS	NUMBER		Number of work areas with an optimal memory requirement comprised between LOW_OPTIMAL_SIZE and HIGH_OPTIMAL_SIZE which have been executed in optimal mode since instance startup
ONEPASS_EXECUTIONS	NUMBER		Number of work areas with an optimal memory requirement comprised between LOW_OPTIMAL_SIZE and HIGH_OPTIMAL_SIZE which have been executed in one-pass mode since instance startup

Column	Datatype	NULL	Description
MULTIPASSES_EXECUTIONS	NUMBER		Number of work areas with an optimal memory requirement comprised between LOW_OPTIMAL_SIZE and HIGH_OPTIMAL_SIZE which have been executed in multi-pass mode since instance startup
TOTAL_EXECUTIONS	NUMBER		Sum of OPTIMAL_EXECUTIONS, ONEPASS_EXECUTIONS, and MULTIPASSES_EXECUTIONS
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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"

6.65 DBA_HIST_SQLBIND

DBA_HIST_SQLBIND displays historical information on bind variables used by SQL cursors.

This view contains snapshots of V$SQL_BIND_CAPTURE.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
SQL_ID	VARCHAR2(13)	NOT NULL	SQL identifier of the parent cursor in the library cache
NAME	VARCHAR2(128)		Name of the bind variable
POSITION	NUMBER	NOT NULL	Position of the bind variable in the SQL statement
DUP_POSITION	NUMBER		If the binding is performed by name and the bind variable is duplicated, then this column gives the position of the primary bind variable.
DATATYPE	NUMBER		Internal identifier for the bind data type. Beginning in Oracle Database 12*c*, a number representing a PL/SQL data type can appear in this column.
DATATYPE_STRING	VARCHAR2(15)		Textual representation of the bind data type. Beginning in Oracle Database 12*c*, a text representation of a PL/SQL-only data type can appear in this column. If the actual data type is a PL/SQL sub type, the name of the data type, not the sub type will be displayed.
CHARACTER_SID	NUMBER		National character set identifier

Column	Datatype	NULL	Description
PRECISION	NUMBER		Precision (for numeric binds)
SCALE	NUMBER		Scale (for numeric binds)
MAX_LENGTH	NUMBER		Maximum bind length
WAS_CAPTURED	VARCHAR2(3)		Indicates whether the bind value was captured (YES) or not (NO)
LAST_CAPTURED	DATE		Date when the bind value was captured. Bind values are captured when SQL statements are executed. To limit the overhead, binds are captured at most every 15 minutes for a given cursor.
VALUE_STRING	VARCHAR2(4000)		Value of the bind represented as a string
VALUE_ANYDATA	ANYDATA		Value of the bind represented using the self-descriptive Sys.AnyData data type. This representation is useful to programmatically decode the value of the bind variable. This column is NULL if a PL/SQL-only data type appears in the DATATYPE column.
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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_BIND_CAPTURE"

6.66 DBA_HIST_SQLCOMMAND_NAME

DBA_HIST_SQLCOMMAND_NAME displays the mapping between SQL opcodes and names.

Column	Datatype	NULL	Description
DBID	NUMBER	NOT NULL	Database ID
COMMAND_TYPE	NUMBER	NOT NULL	SQL command number
COMMAND_NAME	VARCHAR2(64)		SQL command name
CON_DBID	NUMBER		The database ID of the PDB for the sampled session

Column	Datatype	NULL	Description
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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

6.67 DBA_HIST_SQLSTAT

DBA_HIST_SQLSTAT displays historical information about SQL statistics.

This view captures the top SQL statements based on a set of criteria and captures the statistics information from V$SQL. The total value is the value of the statistics since instance startup. The delta value is the value of the statistics from the BEGIN_INTERVAL_TIME to the END_INTERVAL_TIME in the DBA_HIST_SNAPSHOT view.

This view is used with the DBA_HIST_OPTIMIZER_ENV, DBA_HIST_SQLTEXT, and DBA_HIST_SQL_PLAN views to provide a complete picture of historical SQL statistics.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
SQL_ID	VARCHAR2(13)	NOT NULL	SQL identifier of the parent cursor in the library cache
PLAN_HASH_VALUE	NUMBER	NOT NULL	Numerical representation of the SQL plan for the cursor. Comparing one PLAN_HASH_VALUE to another easily identifies whether or not two plans are the same (rather than comparing the two plans line by line).
OPTIMIZER_COST	NUMBER		Cost of the query given by the optimizer
OPTIMIZER_MODE	VARCHAR2(10)		Mode under which the SQL statement is executed
OPTIMIZER_ENV_HASH_VALUE	NUMBER		Hash Value for the optimizer environment
SHARABLE_MEM	NUMBER		Amount of shared memory used by the child cursor (in bytes)
LOADED_VERSIONS	NUMBER		Indicates whether the context heap is loaded (1) or not (0)
VERSION_COUNT	NUMBER		Number of children associated with the cursor
MODULE	VARCHAR2(64)		Contains the name of the module that was executing at the time that the SQL statement was first parsed, which is set by calling DBMS_APPLICATION_INFO.SET_MODULE
ACTION	VARCHAR2(64)		Contains the name of the action that was executing at the time that the SQL statement was first parsed, which is set by calling DBMS_APPLICATION_INFO.SET_ACTION
SQL_PROFILE	VARCHAR2(64)		Name of the applied SQL Profile

Column	Datatype	NULL	Description
FORCE_MATCHING_SIGNATURE	NUMBER		The signature used when the CURSOR_SHARING parameter is set to FORCE
PARSING_SCHEMA_ID	NUMBER		Schema ID that was used to originally build the child cursor
PARSING_SCHEMA_NAME	VARCHAR2(128)		Schema name that was used to originally build the child cursor
PARSING_USER_ID	NUMBER		User ID that was used to originally build the child cursor
FETCHES_TOTAL	NUMBER		Cumulative number of fetches associated with the SQL statement
FETCHES_DELTA	NUMBER		Delta number of fetches associated with the SQL statement
END_OF_FETCH_COUNT_TOTAL	NUMBER		Cumulative number of times this cursor was fully executed since the cursor was brought into the library cache. The value of this statistic is not incremented when the cursor is partially executed, either because it failed during the execution or because only the first few rows produced by this cursor are fetched before the cursor is closed or re-executed. By definition, the value of the END_OF_FETCH_COUNT column should be less or equal to the value of the EXECUTIONS column.
END_OF_FETCH_COUNT_DELTA	NUMBER		Delta number of times this cursor was fully executed since the cursor was brought into the library cache. The value of this statistic is not incremented when the cursor is partially executed, either because it failed during the execution or because only the first few rows produced by this cursor are fetched before the cursor is closed or re-executed.
SORTS_TOTAL	NUMBER		Cumulative number of sorts that were done for this child cursor
SORTS_DELTA	NUMBER		Delta number of sorts that were done for this child cursor
EXECUTIONS_TOTAL	NUMBER		Cumulative number of executions that took place on this object since it was brought into the library cache
EXECUTIONS_DELTA	NUMBER		Delta number of executions that took place on this object since it was brought into the library cache
PX_SERVERS_EXECS_TOTAL	NUMBER		Cumulative number of PX server executions
PX_SERVERS_EXECS_DELTA	NUMBER		Delta number of PX server executions
LOADS_TOTAL	NUMBER		Cumulative number of times the object was either loaded or reloaded
LOADS_DELTA	NUMBER		Delta number of times the object was either loaded or reloaded
INVALIDATIONS_TOTAL	NUMBER		Cumulative number of times this child cursor has been invalidated
INVALIDATIONS_DELTA	NUMBER		Delta number of times this child cursor has been invalidated
PARSE_CALLS_TOTAL	NUMBER		Cumulative number of parse calls for this child cursor
PARSE_CALLS_DELTA	NUMBER		Delta number of parse calls for this child cursor
DISK_READS_TOTAL	NUMBER		Cumulative number of disk reads for this child cursor

Column	Datatype	NULL	Description
DISK_READS_DELTA	NUMBER		Delta number of disk reads for this child cursor
BUFFER_GETS_TOTAL	NUMBER		Cumulative number of buffer gets for this child cursor
BUFFER_GETS_DELTA	NUMBER		Delta number of buffer gets for this child cursor
ROWS_PROCESSED_TOTAL	NUMBER		Cumulative number of rows the parsed SQL statement returns
ROWS_PROCESSED_DELTA	NUMBER		Delta number of rows the parsed SQL statement returns
CPU_TIME_TOTAL	NUMBER		Cumulative value of CPU time (in microseconds) used by this cursor for parsing/executing/fetching
CPU_TIME_DELTA	NUMBER		Delta value of CPU time (in microseconds) used by this cursor for parsing/executing/fetching
ELAPSED_TIME_TOTAL	NUMBER		Cumulative value of elapsed time (in microseconds) used by this cursor for parsing/executing/fetching. If the cursor uses parallel execution, then ELAPSED_TIME_TOTAL is the cumulative time for the query coordinator, plus all parallel query worker processes.
ELAPSED_TIME_DELTA	NUMBER		Delta value of elapsed time (in microseconds) used by this cursor for parsing/executing/fetching
IOWAIT_TOTAL	NUMBER		Cumulative value of user I/O wait time (in microseconds)
IOWAIT_DELTA	NUMBER		Delta value of user I/O wait time (in microseconds)
CLWAIT_TOTAL	NUMBER		Cumulative value of cluster wait time (in microseconds)
CLWAIT_DELTA	NUMBER		Delta value of cluster wait time (in microseconds)
APWAIT_TOTAL	NUMBER		Cumulative value of application wait time (in microseconds)
APWAIT_DELTA	NUMBER		Delta value of application wait time (in microseconds)
CCWAIT_TOTAL	NUMBER		Cumulative value of concurrency wait time (in microseconds)
CCWAIT_DELTA	NUMBER		Delta value of concurrency wait time (in microseconds)
DIRECT_WRITES_TOTAL	NUMBER		Cumulative value of direct writes
DIRECT_WRITES_DELTA	NUMBER		Delta value of direct writes
PLSEXEC_TIME_TOTAL	NUMBER		Cumulative value of PL/SQL Execution Time (in microseconds)
PLSEXEC_TIME_DELTA	NUMBER		Delta value of PL/SQL Execution Time (in microseconds)
JAVEXEC_TIME_TOTAL	NUMBER		Cumulative value of Java Execution Time (in microseconds)
JAVEXEC_TIME_DELTA	NUMBER		Delta value of Java Execution Time (in microseconds)
IO_OFFLOAD_ELIG_BYTES_TOTAL	NUMBER		Cumulative value of number of I/O bytes which can be filtered by the Exadata storage system **See Also:** Oracle Exadata Storage Server Software documentation for more information

Column	Datatype	NULL	Description
IO_OFFLOAD_ELIG_BYTES_DELTA	NUMBER		Delta value of number of I/O bytes which can be filtered by the Exadata storage system **See Also:** Oracle Exadata Storage Server Software documentation for more information
IO_INTERCONNECT_BYTES_TOTAL	NUMBER		Cumulative value of number of I/O bytes exchanged between Oracle Database and the storage system
IO_INTERCONNECT_BYTES_DELTA	NUMBER		Delta value of number of I/O bytes exchanged between Oracle Database and the storage system
PHYSICAL_READ_REQUESTS_TOTAL	NUMBER		Cumulative value of number of physical read I/O requests issued by the monitored SQL
PHYSICAL_READ_REQUESTS_DELTA	NUMBER		Delta value of number of physical read I/O requests issued by the monitored SQL
PHYSICAL_READ_BYTES_TOTAL	NUMBER		Cumulative value of number of bytes read from disks by the monitored SQL
PHYSICAL_READ_BYTES_DELTA	NUMBER		Delta value of number of bytes read from disks by the monitored SQL
PHYSICAL_WRITE_REQUESTS_TOTAL	NUMBER		Cumulative value of number of physical write I/O requests issued by the monitored SQL
PHYSICAL_WRITE_REQUESTS_DELTA	NUMBER		Delta value of number of physical write I/O requests issued by the monitored SQL
PHYSICAL_WRITE_BYTES_TOTAL	NUMBER		Cumulative value of number of bytes written to disks by the monitored SQL
PHYSICAL_WRITE_BYTES_DELTA	NUMBER		Delta value of number of bytes written to disks by the monitored SQL
OPTIMIZED_PHYSICAL_READS_TOTAL	NUMBER		Cumulative value of number of physical reads from the Database Smart Flash Cache or the Exadata Smart Flash Cache by the monitored SQL
OPTIMIZED_PHYSICAL_READS_DELTA	NUMBER		Delta value of number of physical reads from the Database Smart Flash Cache or the Exadata Smart Flash Cache by the monitored SQL
CELL_UNCOMPRESSED_BYTES_TOTAL	NUMBER		Cumulative value of number of uncompressed bytes (that is, size after decompression) that are offloaded to the Exadata cells **See Also:** Oracle Exadata Storage Server Software documentation for more information
CELL_UNCOMPRESSED_BYTES_DELTA	NUMBER		Delta value of number of uncompressed bytes (that is, size after decompression) that are offloaded to the Exadata cells **See Also:** Oracle Exadata Storage Server Software documentation for more information
IO_OFFLOAD_RETURN_BYTES_TOTAL	NUMBER		Cumulative value of number of bytes that are returned by the Exadata cell for smart scan only (that is, not including bytes for other database I/O) **See Also:** Oracle Exadata Storage Server Software documentation for more information
IO_OFFLOAD_RETURN_BYTES_DELTA	NUMBER		Delta value of number of bytes that are returned by the Exadata cell for smart scan only (that is, not including bytes for other database I/O) **See Also:** Oracle Exadata Storage Server Software documentation for more information

Column	Datatype	NULL	Description
BIND_DATA	RAW(2000)		Bind data
FLAG	NUMBER		Reserved for internal use
OBSOLETE_COUNT	NUMBER		Number of times that a parent cursor became obsolete
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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:

- "DBA_HIST_SNAPSHOT"
- "DBA_HIST_OPTIMIZER_ENV"
- "DBA_HIST_SQLTEXT"
- "DBA_HIST_SQL_PLAN"
- *Oracle Database PL/SQL Packages and Types Reference* for more information about the DBMS_APPLICATION_INFO package

6.68 DBA_HIST_SQLTEXT

DBA_HIST_SQLTEXT displays the text of SQL statements belonging to shared SQL cursors captured in the Workload Repository.

This view captures information from V$SQL and is used with the DBA_HIST_SQLSTAT view.

Column	Datatype	NULL	Description
DBID	NUMBER	NOT NULL	Database ID
SQL_ID	VARCHAR2(13)	NOT NULL	SQL identifier of the parent cursor in the library cache
SQL_TEXT	CLOB		Full text for the SQL statement exposed as a CLOB column
COMMAND_TYPE	NUMBER		Oracle command type definition
CON_DBID	NUMBER		The database ID of the PDB for the sampled session

Column	Datatype	NULL	Description
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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"
- "DBA_HIST_SQLSTAT"

6.69 DBA_HIST_STAT_NAME

DBA_HIST_STAT_NAME displays decoded statistic names for the statistics captured in the Workload Repository.

This includes OLAP statisitics and OLAP timed events. This view captures information from V$STATNAME and is used with DBA_HIST_SYSSTAT and DBA_HIST_SYS_TIME_MODEL.

Column	Datatype	NULL	Description
DBID	NUMBER	NOT NULL	Database ID
STAT_ID	NUMBER	NOT NULL	Statistic identifier
STAT_NAME	VARCHAR2(64)	NOT NULL	Statistic name
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$STATNAME"
- "DBA_HIST_SYSSTAT"
- "DBA_HIST_SYS_TIME_MODEL"

6.70 DBA_HIST_STREAMS_APPLY_SUM

DBA_HIST_STREAMS_APPLY_SUM displays information about each apply process and its activities.

This view contains a snapshot of V$STREAMS_APPLY_COORDINATOR, V$STREAMS_APPLY_READER, and V$STREAMS_APPLY_SERVER. This view is intended for use with Automatic Workload Repository (AWR).

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
APPLY_NAME	VARCHAR2(128)	NOT NULL	Name of the apply process
STARTUP_TIME	DATE	NOT NULL	Time that the apply process was last started
READER_TOTAL_MESSAGES_DEQUEUED	NUMBER		Total number of messages dequeued since the apply process was last started
READER_LAG	NUMBER		For captured messages, the delay (in seconds) between the creation of the last message and it being received by the apply process. For user enqueued messages, the delay between the message being enqueued in the local database and being received by the apply process.
COORD_TOTAL_RECEIVED	NUMBER		Total number of transactions received by the coordinator process since the apply process was last started
COORD_TOTAL_APPLIED	NUMBER		Total number of transactions applied by the apply process since the apply process was last started
COORD_TOTAL_ROLLBACKS	NUMBER		Number of transactions which were rolled back due to unexpected contention
COORD_TOTAL_WAIT_DEPS	NUMBER		Number of times since the apply process was last started that an apply server waited to apply a logical change record (LCR) in a transaction until another apply server applied a transaction because of a dependency between the transactions
COORD_TOTAL_WAIT_CMTS	NUMBER		Number of times since the apply process was last started that an apply server waited to commit a transaction until another apply server committed a transaction to serialize commits
COORD_LWM_LAG	NUMBER		For captured messages, the delay (in seconds) between the creation of the message corresponding to the low watermark and it being applied by the apply process. For user enqueued messages, the delay between the message being enqueued in the local database and being applied by the apply process.
SERVER_TOTAL_MESSAGES_APPLIED	NUMBER		Total number of messages applied by all the apply servers since the apply process was last started
SERVER_ELAPSED_DEQUEUE_TIME	NUMBER		Time elapsed (in hundredths of a second) dequeuing messages by all the apply servers since the apply process was last started

Column	Datatype	NULL	Description
SERVER_ELAPSED_APPLY_TIME	NUMBER		Time elapsed (in hundredths of a second) applying messages by all the apply servers since the apply process was last started
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$STREAMS_APPLY_COORDINATOR"
- "V$STREAMS_APPLY_READER"
- "V$STREAMS_APPLY_SERVER"

6.71 DBA_HIST_STREAMS_CAPTURE

DBA_HIST_STREAMS_CAPTURE displays information about each capture process.

This view is intended for use with Automatic Workload Repository (AWR).

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
CAPTURE_NAME	VARCHAR2(128)	NOT NULL	Name of the capture process
STARTUP_TIME	DATE	NOT NULL	Time that the capture process was last started
LAG	NUMBER		Delay (in seconds) between the creation and capture of the most recently captured message
TOTAL_MESSAGES_CAPTURED	NUMBER		Total changes captured since the capture process was last started
TOTAL_MESSAGES_ENQUEUED	NUMBER		Total number of messages enqueued since the capture process was last started
ELAPSED_RULE_TIME	NUMBER		Elapsed time (in hundredths of a second) evaluating rules since the capture process was last started
ELAPSED_ENQUEUE_TIME	NUMBER		Elapsed time (in hundredths of a second) enqueuing messages since the capture process was last started
ELAPSED_REDO_WAIT_TIME	NUMBER		Elapsed time (in hundredths of a second) spent by the capture process in the WAITING FOR REDO state

Column	Datatype	NULL	Description
ELAPSED_PAUSE_TIME	NUMBER		Elapsed pause time
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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

6.72 DBA_HIST_STREAMS_POOL_ADVICE

DBA_HIST_STREAMS_POOL_ADVICE displays historical information about the estimated count of spilled or unspilled messages and the associated time spent in the spill or unspill activity for different Streams pool sizes.

This view is intended for use with Automatic Workload Repository (AWR).

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID of the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number of the snapshot
SIZE_FOR_ESTIMATE	NUMBER	NOT NULL	Pool size for the estimate (in megabytes)
SIZE_FACTOR	NUMBER		Size factor with respect to the current pool size
ESTD_SPILL_COUNT	NUMBER		Estimated count of messages spilled from the Streams pool
ESTD_SPILL_TIME	NUMBER		Estimated elapsed time (in seconds) to spill
ESTD_UNSPILL_COUNT	NUMBER		Estimated count of unspills (read back from disk)
ESTD_UNSPILL_TIME	NUMBER		Estimated elapsed time (in seconds) to unspill
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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

6.73 DBA_HIST_SYS_TIME_MODEL

DBA_HIST_SYS_TIME_MODEL displays historical system time model statistics, including OLAP timed stastistics.

This view contains snapshots of V$SYS_TIME_MODEL.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
STAT_ID	NUMBER	NOT NULL	Statistic ID
STAT_NAME	VARCHAR2(64)	NOT NULL	Statistic name
VALUE	NUMBER		Statistic value
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$SYS_TIME_MODEL"
- "DBA_HIST_CON_SYS_TIME_MODEL"
- "V$CON_SYSMETRIC"

6.74 DBA_HIST_SYSMETRIC_HISTORY

DBA_HIST_SYSMETRIC_HISTORY externalizes all available history of the system metric values for the entire set of data kept in the database.

This view contains snapshots of V$SYSMETRIC_HISTORY.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
BEGIN_TIME	DATE	NOT NULL	Begin time of the interval
END_TIME	DATE	NOT NULL	End time of the interval
INTSIZE	NUMBER	NOT NULL	Interval size (in hundredths of a second)
GROUP_ID	NUMBER	NOT NULL	Group ID
METRIC_ID	NUMBER	NOT NULL	Metric ID
METRIC_NAME	VARCHAR2(64)	NOT NULL	Metric name
VALUE	NUMBER	NOT NULL	Metric Value
METRIC_UNIT	VARCHAR2(64)	NOT NULL	Unit of measurement

Column	Datatype	NULL	Description
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$SYSMETRIC_HISTORY"
- "DBA_HIST_CON_SYSMETRIC_HIST"
- "V$CON_SYSMETRIC_HISTORY"

6.75 DBA_HIST_SYSMETRIC_SUMMARY

DBA_HIST_SYSMETRIC_SUMMARY displays a history of statistical summary of all metric values in the System Metrics Long Duration group.

This view contains snapshots of V$SYSMETRIC_SUMMARY.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
BEGIN_TIME	DATE	NOT NULL	Begin time of the interval
END_TIME	DATE	NOT NULL	End time of the interval
INTSIZE	NUMBER	NOT NULL	Interval size (in hundredths of a second)
GROUP_ID	NUMBER	NOT NULL	Group ID
METRIC_ID	NUMBER	NOT NULL	Metric ID
METRIC_NAME	VARCHAR2(64)	NOT NULL	Metric name
METRIC_UNIT	VARCHAR2(64)	NOT NULL	Unit of measurement
NUM_INTERVAL	NUMBER	NOT NULL	Number of intervals observed
MINVAL	NUMBER	NOT NULL	Minimum value observed
MAXVAL	NUMBER	NOT NULL	Maximum value observed
AVERAGE	NUMBER	NOT NULL	Average over the period
STANDARD_DEVIATION	NUMBER	NOT NULL	One standard deviation
SUM_SQUARES	NUMBER		Sum of the squared deviations from the mean
CON_DBID	NUMBER		The database ID of the PDB for the sampled session

Column	Datatype	NULL	Description
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$SYSMETRIC_SUMMARY"
- "DBA_HIST_CON_SYSMETRIC_SUMM"
- "V$CON_SYSMETRIC_SUMMARY"

6.76 DBA_HIST_SYSSTAT

DBA_HIST_SYSSTAT displays historical system statistics information, including OLAP kernel statistics.

This view contains snapshots of V$SYSSTAT.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
STAT_ID	NUMBER	NOT NULL	Statistic identifier
STAT_NAME	VARCHAR2(64)	NOT NULL	Statistic name
VALUE	NUMBER		Statistic value
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$SYSSTAT"

6.77 DBA_HIST_SYSTEM_EVENT

DBA_HIST_SYSTEM_EVENT displays historical information on total waits for an event.

This view contains snapshots of V$SYSTEM_EVENT.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
EVENT_ID	NUMBER	NOT NULL	Identifier of the wait event
EVENT_NAME	VARCHAR2(64)	NOT NULL	Name of the wait event
WAIT_CLASS_ID	NUMBER		Identifier of the Class of the Wait Event
WAIT_CLASS	VARCHAR2(64)		Name of the Class of the Wait Event
TOTAL_WAITS	NUMBER		Total number of waits for the event
TOTAL_TIMEOUTS	NUMBER		Total number of timeouts for the event
TIME_WAITED_MICRO	NUMBER		Total amount of time waited for the event (in microseconds)
TOTAL_WAITS_FG	NUMBER		Total number of waits for the event, from foreground sessions
TOTAL_TIMEOUTS_FG	NUMBER		Total number of timeouts for the event, from foreground sessions
TIME_WAITED_MICRO_FG	NUMBER		Amount of time waited for the event (in microseconds), from foreground sessions
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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_EVENT"

6.78 DBA_HIST_TABLESPACE

DBA_HIST_TABLESPACE displays tablespace information contained in the Workload Repository.

Column	Datatype	NULL	Description
DBID	NUMBER	NOT NULL	Database ID
TS#	NUMBER	NOT NULL	Tablespace number
TSNAME	VARCHAR2(30)	NOT NULL	Tablespace name
CONTENTS	VARCHAR2(30)		Tablespace contents: • UNDO • PERMANENT • TEMPORARY
SEGMENT_SPACE_MANAGEMENT	VARCHAR2(30)		Indicates whether the free and used space in the tablespace is managed using free lists (MANUAL) or bitmaps (AUTO)
EXTENT_MANAGEMENT	VARCHAR2(30)		Indicates whether the extents in the tablespace are dictionary managed (DICTIONARY) or locally managed (LOCAL)
BLOCK_SIZE	NUMBER		Block size of the tablespace
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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

6.79 DBA_HIST_TABLESPACE_STAT

DBA_HIST_TABLESPACE_STAT displays tablespace information from the control file.

This view contains snapshots of V$TABLESPACE and DBA_TABLESPACES.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
TS#	NUMBER	NOT NULL	Tablespace number
TSNAME	VARCHAR2(30)		Tablespace name
CONTENTS	VARCHAR2(9)		Tablespace contents: • PERMANENT • TEMPORARY

Column	Datatype	NULL	Description
STATUS	VARCHAR2(9)		Tablespace status: • ONLINE • OFFLINE • READ ONLY
SEGMENT_SPACE_MANAGEMENT	VARCHAR2(6)		Indicates whether the free and used segment space in the tablespace is managed using free lists (MANUAL) or bitmaps (AUTO)
EXTENT_MANAGEMENT	VARCHAR2(10)		Indicates whether the extents in the tablespace are dictionary managed (DICTIONARY) or locally managed (LOCAL)
IS_BACKUP	VARCHAR2(5)		Indicates whether the tablespace is part of a backup
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$TABLESPACE"
- "DBA_TABLESPACES"

6.80 DBA_HIST_TBSPC_SPACE_USAGE

DBA_HIST_TBSPC_SPACE_USAGE displays historical tablespace usage statistics.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER		Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
TABLESPACE_ID	NUMBER		Tablespace ID
TABLESPACE_SIZE	NUMBER		Tablespace size (in database blocks)
TABLESPACE_MAXSIZE	NUMBER		Maximum size of the tablespace (in database blocks)
TABLESPACE_USEDSIZE	NUMBER		Used size of the tablespace (in database blocks)
RTIME	VARCHAR2(25)		Runtime
CON_DBID	NUMBER		The database ID of the PDB for the sampled session

Column	Datatype	NULL	Description
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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

6.81 DBA_HIST_TEMPFILE

DBA_HIST_TEMPFILE displays a history of the temp file information from the control file. This view contains snapshots of V$TEMPFILE.

Column	Datatype	NULL	Description
DBID	NUMBER	NOT NULL	Database ID
FILE#	NUMBER	NOT NULL	File identification number
CREATION_CHANGE#	NUMBER	NOT NULL	Change number at which the temp file was created
FILENAME	VARCHAR2(513)	NOT NULL	Name of the temp file
TS#	NUMBER	NOT NULL	Tablespace number
TSNAME	VARCHAR2(30)		Name of the tablespace
BLOCK_SIZE	NUMBER		Block size of the temp file
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$TEMPFILE"

6.82 DBA_HIST_TEMPSTATXS

DBA_HIST_TEMPSTATXS displays information about temporary file read/write statistics.

This view contains snapshots of V$TEMPSTAT.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
FILE#	NUMBER	NOT NULL	File identification number
CREATION_CHANGE#	NUMBER	NOT NULL	Change number at which the temp file was created
FILENAME	VARCHAR2(513)	NOT NULL	Name of the temp file
TS#	NUMBER	NOT NULL	Tablespace number
TSNAME	VARCHAR2(30)		Name of the tablespace
BLOCK_SIZE	NUMBER		Block size of the temp file
PHYRDS	NUMBER		Number of physical reads done
PHYWRTS	NUMBER		Number of times DBWR is required to write
SINGLEBLKRDS	NUMBER		Number of single block reads
READTIM	NUMBER		Time (in hundredths of a second) spent doing reads if the TIMED_STATISTICS parameter is true; 0 if false
WRITETIM	NUMBER		Time (in hundredths of a second) spent doing writes if the TIMED_STATISTICS parameter is true; 0 if false
SINGLEBLKRDTIM	NUMBER		Cumulative single block read time (in hundredths of a second)
PHYBLKRD	NUMBER		Number of physical blocks read
PHYBLKWRT	NUMBER		Number of blocks written to disk, which may be the same as PHYWRTS if all writes are single blocks
WAIT_COUNT	NUMBER		Shows the number of waits at the file level for contended buffers. This value includes the individual wait events that are included in the buffer busy waits wait event. **See Also:** "buffer busy waits"
TIME	NUMBER		Time spent waiting for the wait events in the WAIT_COUNT column
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$TEMPSTAT"

6.83 DBA_HIST_THREAD

DBA_HIST_THREAD displays historical thread information from the control file.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
THREAD#	NUMBER	NOT NULL	Thread number
THREAD_INSTANCE_NUMBER	NUMBER		Instance number of the thread
STATUS	VARCHAR2(6)		Thread status (OPEN) or (CLOSED)
OPEN_TIME	DATE		Last time the thread was opened
CURRENT_GROUP#	NUMBER		Current log group
SEQUENCE#	NUMBER		Sequence number of the current log
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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

6.84 DBA_HIST_TOPLEVELCALL_NAME

DBA_HIST_TOPLEVELCALL_NAME displays the mapping between Oracle top level calls and names.

Column	Datatype	NULL	Description
DBID	NUMBER	NOT NULL	Database ID
TOP_LEVEL_CALL#	NUMBER	NOT NULL	Oracle top level call number
TOP_LEVEL_CALL_NAME	VARCHAR2(64)		Oracle top level call name
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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

6.85 DBA_HIST_UNDOSTAT

DBA_HIST_UNDOSTAT displays the history of histograms of statistical data to show how well the system is working.

The available statistics include undo space consumption, transaction concurrency, and length of queries executed in the instance. This view contains snapshots of V$UNDOSTAT.

Column	Datatype	NULL	Description
BEGIN_TIME	DATE	NOT NULL	Identifies the beginning of the time interval
END_TIME	DATE	NOT NULL	Identifies the end of the time interval
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
UNDOTSN	NUMBER	NOT NULL	Represents the last active undo tablespace in the duration of time. The tablespace ID of the active undo tablespace is returned in this column. If more than one undo tablespace was active in that period, the active undo tablespace that was active at the end of the period is reported.
UNDOBLKS	NUMBER		Represents the total number of undo blocks consumed. You can use this column to obtain the consumption rate of undo blocks, and thereby estimate the size of the undo tablespace needed to handle the workload on your system.
TXNCOUNT	NUMBER		Identifies the total number of transactions executed within the period
MAXQUERYLEN	NUMBER		Identifies the length of the longest query (in number of seconds) executed in the instance during the period. You can use this statistic to estimate the proper setting of the UNDO_RETENTION initialization parameter. The length of a query is measured from the cursor open time to the last fetch/execute time of the cursor. Only the length of those cursors that have been fetched/ executed during the period are reflected in the view.
MAXQUERYSQLID	VARCHAR2(13)		SQL identifier of the longest running SQL statement in the period
MAXCONCURRENCY	NUMBER		Identifies the highest number of transactions executed concurrently within the period
UNXPSTEALCNT	NUMBER		Number of attempts to obtain undo space by stealing unexpired extents from other transactions
UNXPBLKRELCNT	NUMBER		Number of unexpired blocks removed from certain undo segments so they can be used by other transactions
UNXPBLKREUCNT	NUMBER		Number of unexpired undo blocks reused by transactions
EXPSTEALCNT	NUMBER		Number of attempts to steal expired undo blocks from other undo segments
EXPBLKRELCNT	NUMBER		Number of expired undo blocks stolen from other undo segments

Column	Datatype	NULL	Description
EXPBLKREUCNT	NUMBER		Number of expired undo blocks reused within the same undo segments
SSOLDERRCNT	NUMBER		Identifies the number of times the error ORA-01555 occurred. You can use this statistic to decide whether the UNDO_RETENTION initialization parameter is set properly given the size of the undo tablespace. Increasing the value of UNDO_RETENTION can reduce the occurrence of this error.
NOSPACEERRCNT	NUMBER		Identifies the number of times space was requested in the undo tablespace and there was no free space available. That is, all of the space in the undo tablespace was in use by active transactions. The corrective action is to add more space to the undo tablespace.
ACTIVEBLKS	NUMBER		Total number of blocks in the active extents of the undo tablespace for the instance at the sampled time in the period
UNEXPIREDBLKS	NUMBER		Total number of blocks in the unexpired extents of the undo tablespace for the instance at the sampled time in the period
EXPIREDBLKS	NUMBER		Total number of blocks in the expired extents of the undo tablespace for the instance at the sampled time in the period
TUNED_UNDORETENTION	NUMBER		System tuned value indicating the period for which undo is being retained The value of this column is not meaningful on an Oracle Active Data Guard standby database instance, because the system does not tune this value on such instances.
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$UNDOSTAT"

6.86 DBA_HIST_WAITCLASSMET_HISTORY

DBA_HIST_WAITCLASSMET_HISTORY displays the history of the wait event class metric data kept by the Workload Repository.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID of the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number of the snapshot
WAIT_CLASS_ID	NUMBER	NOT NULL	Identifier of the class of the wait event
WAIT_CLASS	VARCHAR2(64)		Name of the class of the wait event
BEGIN_TIME	DATE	NOT NULL	Begin time of the interval
END_TIME	DATE	NOT NULL	End time of the interval
INTSIZE	NUMBER	NOT NULL	Interval size (in hundredths of a second)
GROUP_ID	NUMBER	NOT NULL	Metric group ID
AVERAGE_WAITER_COUNT	NUMBER	NOT NULL	Average waiter count
DBTIME_IN_WAIT	NUMBER	NOT NULL	Percent of database time spent in the wait
TIME_WAITED	NUMBER	NOT NULL	Time waited during the interval (in microseconds)
WAIT_COUNT	NUMBER	NOT NULL	Number of times waited
TIME_WAITED_FG	NUMBER		Time waited (in hundredths of a second), from foreground sessions
WAIT_COUNT_FG	NUMBER		Number of times waited, from foreground sessions
CON_DBID	NUMBER		The database ID of the PDB for the sampled session
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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

6.87 DBA_HIST_WAITSTAT

DBA_HIST_WAITSTAT displays historical block contention statistics. This view contains snapshots of V$WAITSTAT.

Column	Datatype	NULL	Description
SNAP_ID	NUMBER	NOT NULL	Unique snapshot ID
DBID	NUMBER	NOT NULL	Database ID for the snapshot
INSTANCE_NUMBER	NUMBER	NOT NULL	Instance number for the snapshot
CLASS	VARCHAR2(18)	NOT NULL	Class of the block
WAIT_COUNT	NUMBER		Number of waits by the OPERATION for this CLASS of block
TIME	NUMBER		Sum of all wait times for all the waits by the OPERATION for this CLASS of block
CON_DBID	NUMBER		The database ID of the PDB for the sampled session

Column	Datatype	NULL	Description
CON_ID	NUMBER		The ID of the container that CON_DBID identifies. 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$WAITSTAT"

6.88 DBA_HIST_WR_CONTROL

DBA_HIST_WR_CONTROL displays the control information for the Workload Repository.

Column	Datatype	NULL	Description
DBID	NUMBER	NOT NULL	Database ID
SNAP_INTERVAL	INTERVAL DAY(5) TO SECOND(1)	NOT NULL	Snapshot interval; how often to automatically take snapshots
RETENTION	INTERVAL DAY(5) TO SECOND(1)	NOT NULL	Retention setting for the snapshots; amount of time to keep the snapshots
TOPNSQL	VARCHAR2(10)		The number of Top SQL flushed for each SQL criteria (elapsed time, CPU time, parse calls, sharable memory, version count)
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
SRC_DBID	NUMBER		Database ID of the non-CDB, CDB, or PDB where the AWR snapshot data was collected
SRC_DBNAME	VARCHAR2(128)		Database name of the non-CDB, CDB, or PDB where the AWR snapshot data was collected
TABLESPACE_NAME	VARCHAR2(128)		Name of the tablespace in which AWR snapshot data resides

6.89 DBA_HIST_WR_SETTINGS

DBA_HIST_WR_SETTINGS displays the settings and metadata for the Workload Repository.

Column	Datatype	NULL	Description
LOCAL_AWRDBID	NUMBER	NOT NULL	Database ID of the local database
VIEW_LOCATION	VARCHAR2(8)		Data source of the DBA_HIST dictionary views. Possible values include: • AWR_PDB: Views display AWR data stored in the PDB. • AWR_ROOT: Views display AWR data stored in the root.
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

6.90 DBA_HISTOGRAMS

DBA_HISTOGRAMS is a synonym for DBA_TAB_HISTOGRAMS.

> **See Also:**
>
> "DBA_TAB_HISTOGRAMS"

6.91 DBA_HIVE_COLUMNS

DBA_HIVE_COLUMNS describes all Hive columns in a Hive metastore. Its columns are the same as those in ALL_HIVE_COLUMNS.

> **See Also:**
>
> "ALL_HIVE_COLUMNS"

6.92 DBA_HIVE_DATABASES

DBA_HIVE_DATABASES describes all the Hive schemas in a Hadoop cluster. Its columns are the same as those in ALL_HIVE_DATABASES.

> **See Also:**
>
> "ALL_HIVE_DATABASES"

6.93 DBA_HIVE_PART_KEY_COLUMNS

DBA_HIVE_PART_KEY_COLUMNS provides information about all Hive table partition columns in the database. Its columns are the same as those in ALL_HIVE_PART_KEY_COLUMNS.

> **See Also:**
>
> "ALL_HIVE_PART_KEY_COLUMNS"

6.94 DBA_HIVE_TAB_PARTITIONS

DBA_HIVE_TAB_PARTITIONS provides information about all Hive table partitions in the database. Its columns are the same as those in ALL_HIVE_TAB_PARTITIONS.

> **See Also:**
>
> "ALL_HIVE_TAB_PARTITIONS"

6.95 DBA_HIVE_TABLES

DBA_HIVE_TABLES provides information about all the Hive tables in the Hive metastore. Its columns are the same as those in ALL_HIVE_TABLES.

> **See Also:**
>
> "ALL_HIVE_TABLES"

6.96 DBA_HOST_ACES

DBA_HOST_ACES describes access control entries defined in host access control lists.

Related View

USER_HOST_ACES describes the status of access control entries for the current user to access network hosts through PL/SQL network utility packages. This view does not display the ACE_ORDER, START_DATE, END_DATE, GRANT_TYPE, INVERTED_PRINCIPAL, PRINCIPAL, or PRINCIPAL_TYPE columns.

Column	Datatype	NULL	Description
HOST	VARCHAR2(1000)	NOT NULL	Network host
LOWER_PORT	NUMBER(5)		Lower bound of the port range
UPPER_PORT	NUMBER(5)		Upper bound of the port range
ACE_ORDER	NUMBER	NOT NULL	Order number of the access control entry
START_DATE	TIMESTAMP(6)		Start date of the access control entry
END_DATE	TIMESTAMP(6)		End date of the access control entry
GRANT_TYPE	VARCHAR2(5)		Indicates whether the access control entry grants or denies the privilege
INVERTED_PRINCIPAL	VARCHAR2(3)		Indicates whether the principal is inverted or not
PRINCIPAL	VARCHAR2(128)		Principal the privilege is applied to
PRINCIPAL_TYPE	VARCHAR2(16)		Type of the principal
PRIVILEGE	VARCHAR2(128)		Privilege

See Also:

"USER_HOST_ACES"

6.97 DBA_HOST_ACLS

DBA_HOST_ACLS describes access control lists assigned to restrict access to network hosts through PL/SQL network utility packages.

Column	Datatype	NULL	Description
HOST	VARCHAR2(1000)	NOT NULL	Network host
LOWER_PORT	NUMBER(5)		Lower bound of the port range
UPPER_PORT	NUMBER(5)		Upper bound of the port range
ACL	VARCHAR2(4000)		The name of the access control list
ACLID	RAW(8)		The object ID of the access control list
ACL_OWNER	VARCHAR2(128)		The owner of the access control list

6.98 DBA_IDENTIFIERS

DBA_IDENTIFIERS displays information about the identifiers in all stored objects in the database. Its columns are the same as those in ALL_IDENTIFIERS.

> **See Also:**
>
> "ALL_IDENTIFIERS"

6.99 DBA_ILMDATAMOVEMENTPOLICIES

DBA_ILMDATAMOVEMENTPOLICIES contains information specific to data movement-related attributes of an Automatic Data Optimization policy in a database.

Related View

USER_ILMDATAMOVEMENTPOLICIES contains information specific to data movement-related attributes of an Automatic Data Optimization policy for a user.

> **Note:**
>
> Automatic Data Optimization is supported in Oracle Database 12*c* Release 2 multitenant environments.

Column	Datatype	NULL	Description
POLICY_NAME	VARCHAR2(128)	NOT NULL	The Automatic Data Optimization policy name is autogenerated
ACTION_TYPE	VARCHAR2(11)		Type of data movement action performed by the Automatic Data Optimization policy: • ANNOTATE • COMPRESSION • EVICT • STORAGE
SCOPE	VARCHAR2(7)		Identifies the scope of the Automatic Data Optimization policy: • ROW • GROUP • SEGMENT
COMPRESSION_LEVEL	VARCHAR2(30)		Compression level to move selected rows or the entire segment to, for a compression Automatic Data Optimization policy
TIER_TABLESPACE	VARCHAR2(128)		Tablespace to move the object to, for a storage Automatic Data Optimization policy

Column	Datatype	NULL	Description
TIER_STATUS	VARCHAR2(9)		This column is only valid for storage tiering policies, and indicates whether the storage tiering policy was specified with a READ ONLY clause. This column takes READ ONLY as a potential value. In all other cases, it is blank. In other words, this column indicates whether the tablespace the object is being moved to will be made READ ONLY after the movement.
CONDITION_TYPE	VARCHAR2(22)		Column on which the condition for the policy is based. Possible values: • LAST ACCESS TIME • LAST MODIFICATION TIME • CREATION TIME • USER DEFINED • LOW ACCESS
CONDITION_DAYS	NUMBER		Number of days in the condition for the policy
CUSTOM_FUNCTION	VARCHAR2(128)		Optional function that evaluates the precondition on the policy
POLICY_SUBTYPE	VARCHAR2(10)		Storage tier on which the policy is specified
ACTION_CLAUSE	CLOB		Text of the action executed by the policy
TIER_TO	VARCHAR2(10)		Storage tier where the data is placed after the policy is executed

See Also:

"USER_ILMDATAMOVEMENTPOLICIES"

6.100 DBA_ILMEVALUATIONDETAILS

DBA_ILMEVALUATIONDETAILS displays details on evaluation of Automatic Data Optimization policies considered for Automatic Data Optimization tasks.

It also shows the job name that executes the policy, in case the policy was selected for execution. If the policy was not selected for execution, this view provides a reason.

Related View

USER_ILMEVALUATIONDETAILS displays details on evaluation of Automatic Data Optimization policies considered for Automatic Data Optimization tasks for a user. It also shows the job name that executes the policy, in case the policy was selected for execution. If the policy was not selected for execution, this view provides a reason.

Note:

Automatic Data Optimization is supported in Oracle Database 12*c* Release 2 multitenant environments.

Column	Datatype	NULL	Description
TASK_ID	NUMBER		Number that uniquely identifies a specific Automatic Data Optimization task
POLICY_NAME	VARCHAR2(128)	NOT NULL	Name of the Automatic Data Optimization policy
OBJECT_OWNER	VARCHAR2(128)	NOT NULL	Owner of the object associated with the Automatic Data Optimization policy
OBJECT_NAME	VARCHAR2(128)	NOT NULL	Name of the object associated with the Automatic Data Optimization policy
SUBOBJECT_NAME	VARCHAR2(128)		Name of the subobject associated with the Automatic Data Optimization policy
OBJECT_TYPE	VARCHAR2(18)		Object type. Valid values include TABLE, TABLE PARTITION, and TABLE SUBPARTITION.
SELECTED_FOR_EXECUTION	VARCHAR2(42)		Indicates whether the policy has been selected for execution on the object. If not, the reason for not being selected is listed. Possible values: • POLICY DISABLED • SELECTED FOR EXECUTION • POLICY OVERRULED • INHERITED POLICY OVERRULED • PRECONDITION NOT SATISFIED • JOB ALREADY EXISTS • NO OPERATION SINCE LAST ILM ACTION • TABLE HAS MATERIALIZED VIEW • TARGET COMPRESSION NOT HIGHER THAN CURRENT • STATISTICS NOT AVAILABLE The value SELECTED FOR EXECUTION means a job was created for this policy on the object. The other values state the reason why the policy on the object was selected for execution.
JOB_NAME	VARCHAR2(128)		Name of the job in the case where the policy is selected for execution on this object
COMMENTS	VARCHAR2(4000)		Reserved for future use

> **See Also:**
>
> "USER_ILMEVALUATIONDETAILS"

6.101 DBA_ILMOBJECTS

DBA_ILMOBJECTS displays all the Automatic Data Optimization policies and objects in the database.

Many objects inherit policies via their parent objects or because they were created in a particular tablespace. This view provides a mapping between the policies and objects and indicates whether a policy is inherited by an object or is directly specified on it.

Related View

USER_ILMOBJECTS displays all the Automatic Data Optimization policies and objects for a user.

> **Note:**
>
> Automatic Data Optimization is supported in Oracle Database 12*c* Release 2 multitenant environments.

Column	Datatype	NULL	Description
POLICY_NAME	VARCHAR2(128)	NOT NULL	Policy name is auto-generated
OBJECT_OWNER	VARCHAR2(128)	NOT NULL	Owner of the object associated with the Automatic Data Optimization policy
OBJECT_NAME	VARCHAR2(128)	NOT NULL	Name of the object associated with the Automatic Data Optimization policy
SUBOBJECT_NAME	VARCHAR2(128)		Name of the subobject associated with the Automatic Data Optimization policy
OBJECT_TYPE	VARCHAR2(18)		Object type. Valid values include: • INDEX • INDEX PARTITION • LOB • LOB PARTITION • TABLE • TABLE PARTITION • TABLE SUBPARTITION Direct policies on Index, Index Partition, LOB, and LOB Partition are not supported in Oracle Database 12*c*.
INHERITED_FROM	VARCHAR2(20)		Indicates if the policy is inherited by the object or subobject, or directly specified on the object or subobject. If the policy is inherited, the level from which the policy is inherited (TABLE, TABLE PARTITION, TABLESPACE) is identified.
TBS_INHERITED_FROM	VARCHAR2(30)		The tablespace name, if the policy is inherited from a tablespace
ENABLED	VARCHAR2(7)		Indicates if the Automatic Data Optimization policy is enabled for the object (YES or NO)
DELETED	VARCHAR2(7)		Possible values: • YES - Indicates that the policy is deleted for any objects that may be added in the future, but is active for those objects that are currently associated with that policy • NO - Indicates that the policy is active

> **See Also:**
>
> "USER_ILMOBJECTS"

6.102 DBA_ILMPARAMETERS

DBA_ILMPARAMETERS can be queried to provide information on the Automatic Data Optimization parameters in the database and their values.

> **Note:**
>
> Automatic Data Optimization is supported in Oracle Database 12*c* Release 2 multitenant environments.

Column	Datatype	NULL	Description
NAME	VARCHAR2(128)		Name of the Automatic Data Optimization environment parameter. The value is one of the constants defined in the DBMS_ILM_ADMIN package.
VALUE	NUMBER		Value of the parameter

> **See Also:**
>
> - *Oracle Database PL/SQL Packages and Types Reference* for more information about the API interface for implementing Automatic Data Optimization strategies
> - *Oracle Database PL/SQL Packages and Types Reference* for more information about the DBMS_ILM_ADMIN package

6.103 DBA_ILMPOLICIES

DBA_ILMPOLICIES displays details about Automatic Data Optimization policies in the database.

The view contains common details relevant to all types of Automatic Data Optimization policies, not just details relevant to the data movement-related Automatic Data Optimization policies.

Related View

USER_ILMPOLICIES displays details about Automatic Data Optimization policies owned by the user.

> **Note:**
>
> Automatic Data Optimization is supported in Oracle Database 12*c* Release 2 multitenant environments.

Column	Datatype	NULL	Description
POLICY_NAME	VARCHAR2(128)		The name of the Automatic Data Optimization policy is auto-generated
POLICY_TYPE	VARCHAR2(13)		Type of the policy. Valid values include DATAMOVEMENT.
TABLESPACE	VARCHAR2(30)		Tablespace name, in the case of a tablespace-level policy
ENABLED	VARCHAR2(6)		Indicates whether the policy is enabled or not
DELETED	VARCHAR2(7)		Possible values: • YES - Indicates that the policy associated has been deleted (but shall remain active for this object) • NO - Indicates that the policy is active

Note:

Automatic Data Optimization policies cannot be set on tables with object types or materialized views.

See Also:

"USER_ILMPOLICIES"

6.104 DBA_ILMRESULTS

DBA_ILMRESULTS displays information on data movement-related Automatic Data Optimization jobs in the database.

Related View

USER_ILMRESULTS displays information on data movement-related Automatic Data Optimization jobs for tasks created by the user.

Note:

Automatic Data Optimization is supported in Oracle Database 12*c* Release 2 multitenant environments.

Column	Datatype	NULL	Description
TASK_ID	NUMBER		Number that uniquely identifies a specific Automatic Data Optimization task
JOB_NAME	VARCHAR2(128)		Name of the Automatic Data Optimization job

Column	Datatype	NULL	Description
JOB_STATE	VARCHAR2(35)		State of the job. Possible values: • JOB CREATED • COMPLETED SUCCESSFULLY • FAILED • STOPPED • JOB CREATION FAILED • DEPENDENT OBJECTS BEING REBUILT • FAILED TO REBUILD DEPENDENT OBJECTS
START_TIME	TIMESTAMP(6)		Start time of the Automatic Data Optimization job
COMPLETION_TIME	TIMESTAMP(6)		Completion time of the Automatic Data Optimization job
COMMENTS	VARCHAR2(4000)		Additional information in cases where the Automatic Data Optimization job execution fails
STATISTICS	CLOB		Job specific statistics, such as space saved via compression. This column is in the form of comma separated name / value pairs, with each pair representing a particular statistic name and value.

See Also:

"USER_ILMRESULTS"

6.105 DBA_ILMTASKS

DBA_ILMTASKS displays information on Automatic Data Optimization execution.

Related View

USER_ILMTASKS displays information on Automatic Data Optimization tasks created by a user.

Note:

Automatic Data Optimization is supported in Oracle Database 12*c* Release 2 multitenant environments.

Column	Datatype	NULL	Description
TASK_ID	NUMBER		Number that uniquely identifies a specific Automatic Data Optimization task
TASK_OWNER	VARCHAR2(128)	NOT NULL	User who initiates the task
STATE	VARCHAR2(9)		Possible values: • INACTIVE: Indicates that the task was created for previewing • ACTIVE: Indicates that jobs have been created for the qualifying policies in the task • COMPLETE: Indicates that the task has completed

Column	Datatype	NULL	Description
CREATION_TIME	TIMESTAMP(6)		The time that the task was created
START_TIME	TIMESTAMP(6)		Start time of a specific task
COMPLETION_TIME	TIMESTAMP(6)		Completion time of a specific task

> **See Also:**
>
> "USER_ILMTASKS"

6.106 DBA_IM_EXPRESSIONS

DBA_IM_EXPRESSIONS provides information about the list of expressions (SYS_IME virtual columns) that are currently enabled for in-memory storage.

Typically, you can query this view after invoking the DBMS_INMEMORY_ADMIN.IME_CAPTURE_EXPRESSIONS PL/SQL procedure to see the list of hot expressions added to different tables across the database.

Based on this view, you can:

- Populate expressions on a particular table immediately
- Drop certain expressions that are marked for in-memory but not desired by users

Related View

USER_IM_EXPRESSIONS provides information about the list of expressions (SYS_IME virtual columns) that are currently enabled for in-memory storage in schemas owned by the current user. This view does not display the OWNER column.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(129)		Table owner
TABLE_NAME	VARCHAR2(129)		Table name
OBJECT_NUMBER	NUMBER		Object number of the table
COLUMN_NAME	VARCHAR2(128)	NOT NULL	Column name of the expression added to the table (with SYS_IME prefix)
SQL_EXPRESSION	LONG		SQL representation of the expression

> **See Also:**
>
> "USER_IM_EXPRESSIONS"

6.107 DBA_IMMUTABLE_ROW_VERSION_COLS

DBA_IMMUTABLE_ROW_VERSION_COLS displays information about row versioned columns in all immutable tables in the database. Its columns are the same as those in ALL_IMMUTABLE_ROW_VERSION_COLS.

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

> **See Also:**
>
> "ALL_IMMUTABLE_ROW_VERSION_COLS"

6.108 DBA_IMMUTABLE_ROW_VERSION_HISTORY

DBA_IMMUTABLE_ROW_VERSION_HISTORY provides a history of row versions in all immutable tables in the database. Its columns are the same as those in ALL_IMMUTABLE_ROW_VERSION_HISTORY.

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

> **See Also:**
>
> "ALL_IMMUTABLE_ROW_VERSION_HISTORY"

6.109 DBA_IMMUTABLE_TABLE_COLUMNS

DBA_IMMUTABLE_TABLE_COLUMNS displays information about columns valid in each epoch in all immutable tables in the database. Its columns are the same as those in ALL_IMMUTABLE_TABLE_COLUMNS.

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

> **See Also:**
>
> "ALL_IMMUTABLE_TABLE_COLUMNS"

6.110 DBA_IMMUTABLE_TABLE_EPOCHS

`DBA_IMMUTABLE_TABLE_EPOCHS` displays epoch information for all immutable tables in the database. Its columns are the same as those in `ALL_IMMUTABLE_TABLE_EPOCHS`.

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

> **See Also:**
>
> "ALL_IMMUTABLE_TABLE_EPOCHS"

6.111 DBA_IMMUTABLE_TABLES

`DBA_IMMUTABLE_TABLES` describes all immutable tables in the database. Its columns are the same as those in `ALL_IMMUTABLE_TABLES`.

> **See Also:**
>
> "ALL_IMMUTABLE_TABLES"

6.112 DBA_INCOMPLETE_SAGAS

`DBA_INCOMPLETE_SAGAS` describes all incomplete sagas in the database. Its columns are the same as those in `ALL_INCOMPLETE_SAGAS`.

This view displays sagas that were either initiated in the current PDB or joined by participants in the current PDB.

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

> **See Also:**
>
> "ALL_INCOMPLETE_SAGAS"

6.113 DBA_IND_COLUMNS

`DBA_IND_COLUMNS` describes the columns of all the indexes on all tables and clusters in the database. Its columns are the same as those in `ALL_IND_COLUMNS`.

> **See Also:**
>
> "ALL_IND_COLUMNS"

6.114 DBA_IND_EXPRESSIONS

`DBA_IND_EXPRESSIONS` lists expressions of function-based indexes on all tables and clusters in the database. Its columns are the same as those in `ALL_IND_EXPRESSIONS`.

> **See Also:**
>
> "ALL_IND_EXPRESSIONS"

6.115 DBA_IND_PARTITIONS

`DBA_IND_PARTITIONS` displays, for each index partition in the database, the partition-level partitioning information, the storage parameters for the partition, and various partition statistics generated by the `DBMS_STATS` package. Its columns are the same as those in `ALL_IND_PARTITIONS`.

> **See Also:**
>
> "ALL_IND_PARTITIONS"

6.116 DBA_IND_PENDING_STATS

DBA_IND_PENDING_STATS describes pending statistics for all tables, partitions, and subpartitions in the database collected using the DBMS_STATS package. Its columns are the same as those in ALL_IND_PENDING_STATS.

> **See Also:**
>
> "ALL_IND_PENDING_STATS"

6.117 DBA_IND_STATISTICS

DBA_IND_STATISTICS displays optimizer statistics for all indexes in the database collected using the DBMS_STATS package. Its columns are the same as those in ALL_IND_STATISTICS.

> **See Also:**
>
> "ALL_IND_STATISTICS"

6.118 DBA_IND_SUBPARTITIONS

DBA_IND_SUBPARTITIONS displays, for each index subpartition in the database, the subpartition-level partitioning information, the storage parameters for the subpartition, and various subpartition statistics generated by the DBMS_STATS package. Its columns are the same as those in ALL_IND_SUBPARTITIONS.

> **See Also:**
>
> "ALL_IND_SUBPARTITIONS"

6.119 DBA_INDEX_USAGE

DBA_INDEX_USAGE displays cumulative statistics for each index.

Column	Datatype	NULL	Description
OBJECT_ID	NUMBER	NOT NULL	Object ID for the index
NAME	VARCHAR2(128)	NOT NULL	Index name
OWNER	VARCHAR2(128)	NOT NULL	Index owner
TOTAL_ACCESS_COUNT	NUMBER		Total number of times the index has been accessed
TOTAL_EXEC_COUNT	NUMBER		Total executions the index has participated in

Column	Datatype	NULL	Description
TOTAL_ROWS_RETURNED	NUMBER		Total rows returned by the index. Index usage is categorized into buckets of different ranges. Each bucket has a range of values for access count and rows returned. An entry is placed into a bucket if the rows returned or access counts falls within the range of that bucket.
BUCKET_0_ACCESS_COUNT	NUMBER		The index has not been accessed
BUCKET_1_ACCESS_COUNT	NUMBER		The index has been accessed once
BUCKET_2_10_ACCESS_COUNT	NUMBER		The index has been accessed between 2 and 10 times
BUCKET_2_10_ROWS_RETURNED	NUMBER		The index has returned between 2 and 10 rows
BUCKET_11_100_ACCESS_COUNT	NUMBER		The index has been accessed between 11 and 100 times
BUCKET_11_100_ROWS_RETURNED	NUMBER		The index has returned between 11 and 100 rows
BUCKET_101_1000_ACCESS_COUNT	NUMBER		The index has been accessed between 101 and 1000 times
BUCKET_101_1000_ROWS_RETURNED	NUMBER		The index has returned between 101 and 1000 rows
BUCKET_1000_PLUS_ACCESS_COUNT	NUMBER		The index has been accessed more than 1000 times
BUCKET_1000_PLUS_ROWS_RETURNED	NUMBER		The index has returned more than 1000 rows
LAST_USED	DATE		Time that the index was last used

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

6.120 DBA_INDEXES

DBA_INDEXES describes all indexes in the database. To gather statistics for this view, use the DBMS_STATS package. This view supports parallel partitioned index scans. Its columns are the same as those in ALL_INDEXES.

> **See Also:**
>
> "ALL_INDEXES"

6.121 DBA_INDEXTYPE_ARRAYTYPES

DBA_INDEXTYPE_ARRAYTYPES displays information about the array types specified by all indextypes in the database. Its columns are the same as those in ALL_INDEXTYPE_ARRAYTYPES.

> **See Also:**
>
> "ALL_INDEXTYPE_ARRAYTYPES"

6.122 DBA_INDEXTYPE_COMMENTS

DBA_INDEXTYPE_COMMENTS displays comments for all user-defined indextypes in the database. Its columns are the same as those in ALL_INDEXTYPE_COMMENTS.

> **See Also:**
>
> "ALL_INDEXTYPE_COMMENTS"

6.123 DBA_INDEXTYPE_OPERATORS

DBA_INDEXTYPE_OPERATORS lists all the operators supported by indextypes in the database. Its columns are the same as those in ALL_INDEXTYPE_OPERATORS.

> **See Also:**
>
> "ALL_INDEXTYPE_OPERATORS"

6.124 DBA_INDEXTYPES

DBA_INDEXTYPES describes all indextypes in the database. Its columns are the same as those in ALL_INDEXTYPES.

> **See Also:**
>
> "ALL_INDEXTYPES"

6.125 DBA_INMEMORY_ADVISOR_RECOMMENDATION

DBA_INMEMORY_ADVISOR_RECOMMENDATION displays In-Memory Advisor recommendations.

In-Memory Advisor simulates various Oracle Database In-Memory sizes and estimates the database time for a reference workload for each size. The results are displayed in this view.

Column	Datatype	NULL	Description
TASK_ID	NUMBER		In-Memory Advisor task ID
INMEMORY_SIZE	NUMBER		Simulated In-Memory size (in bytes)
ESTIMATED_DB_TIME_LOW	NUMBER		Low end of the predicted database time range (in seconds)
ESTIMATED_DB_TIME_HIGH	NUMBER		High end of the predicted database time range (in seconds)
ESTIMATED_DB_TIME_ANALYTICS_LOW	NUMBER		Low end of the predicted database analytical time range (in seconds)
ESTIMATED_DB_TIME_ANALYTICS_HIGH	NUMBER		High end of the predicted database analytical time range (in seconds)
RECOMMENDED_OBJ_LIST	CLOB		List of recommended objects for the simulated In-Memory size (INMEMORY_SIZE)

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

6.126 DBA_INMEMORY_AIMTASKDETAILS

DBA_INMEMORY_AIMTASKDETAILS displays details for an Automatic In-Memory management task.

Column	Datatype	NULL	Description
TASK_ID	NUMBER		Number that uniquely identifies a specific automatic IM column store management task
OBJECT_OWNER	VARCHAR2(128)		Owner of the object subject to automatic IM column store management task action
OBJECT_NAME	VARCHAR2(128)		Name of the object subject to automatic IM column store management task action
SUBOBJECT_NAME	VARCHAR2(128)		Name of the subobject subject to automatic IM column store management task action
ACTION	VARCHAR2(16)		Action taken on the object
STATE	VARCHAR2(10)		Status of the action on the object

> **See Also:**
>
> *Oracle Database In-Memory Guide* for more information about configuring the Automatic In-Memory feature

6.127 DBA_INMEMORY_AIMTASKS

DBA_INMEMORY_AIMTASKS displays information about Automatic In-Memory management tasks.

Column	Datatype	NULL	Description
TASK_ID	NUMBER		Number that uniquely identifies a specific automatic IM column store management task
CREATION_TIME	TIMESTAMP(6)		Creation time of the task
STATE	VARCHAR2(7)		State of the task

> **See Also:**
>
> *Oracle Database In-Memory Guide* for more information about configuring the Automatic In-Memory feature

6.128 DBA_INTERNAL_TRIGGERS

DBA_INTERNAL_TRIGGERS describes internal triggers on all tables in the database. Its columns are the same as those in ALL_INTERNAL_TRIGGERS.

> **See Also:**
>
> "ALL_INTERNAL_TRIGGERS"

6.129 DBA_INVALID_OBJECTS

DBA_INVALID_OBJECTS describes all invalid objects in the database. You can use this view to identify invalid objects before and after a database upgrade.

This view eliminates old versions of object types. It only includes the object type it if is the latest version.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)		Owner of the object
OBJECT_NAME	VARCHAR2(128)		Name of the object
SUBOBJECT_NAME	VARCHAR2(128)		Name of the subobject (for example, partition)
OBJECT_ID	NUMBER		Dictionary object number of the object.

Column	Datatype	NULL	Description
DATA_OBJECT_ID	NUMBER		Dictionary object number of the segment that contains the object. **Note:** OBJECT_ID and DATA_OBJECT_ID display data dictionary metadata. Do not confuse these numbers with the unique 16-byte object identifier (*object ID*) that Oracle Database assigns to row objects in object tables in the system.
OBJECT_TYPE	VARCHAR2(23)		Type of the object (such as TABLE, INDEX). The current version of the type is shown only if it is invalid. In other words, if prior versions of the type are invalid but the most recent version of the type is valid, it will not be in this list.
CREATED	DATE		Timestamp for the creation of the object
LAST_DDL_TIME	DATE		Timestamp for the last modification of the object and dependent objects resulting from a DDL statement (including grants and revokes)
TIMESTAMP	VARCHAR2(19)		Timestamp for the specification of the object (character data)
STATUS	VARCHAR2(7)		Status of the object: • VALID • INVALID • N/A
TEMPORARY	VARCHAR2(1)		Indicates whether the object is temporary (the current session can see only data that it placed in this object itself) (Y) or not (N)
GENERATED	VARCHAR2(1)		Indicates whether the name of this object was system-generated (Y) or not (N)
SECONDARY	VARCHAR2(1)		Indicates whether this is a secondary object created by the ODCIIndexCreate method of the Oracle Data Cartridge (Y) or not (N)
NAMESPACE	NUMBER		Namespace for the object
EDITION_NAME	VARCHAR2(128)		Name of the edition in which the object is actual
SHARING	VARCHAR2(18)		Values: • DATA LINK - If the object is data-linked or a data link to an object in the root • METADATA LINK - If the object is metadata-linked or a metadata link to an object in the root • EXTENDED DATA LINK - If the object is extended-data-linked or an extended data link to an object in the root • NONE - If none of the above applies
EDITIONABLE	VARCHAR2(1)		Values: • Y - For objects marked EDITIONABLE • N - For objects marked NONEDITIONABLE • NULL - For objects whose type is not editionable in the database

Column	Datatype	NULL	Description
ORACLE_MAINTAINED	VARCHAR2(1)		Denotes whether the object was created, and is maintained, by Oracle-supplied scripts (such as catalog.sql or catproc.sql). An object for which this column has the value Y must not be changed in any way except by running an Oracle-supplied script.
APPLICATION	VARCHAR2(1)		Indicates whether the object is an Application common object (Y) or not (N)
DEFAULT_COLLATION	VARCHAR2(100)		Default collation for the object
DUPLICATED	VARCHAR2(1)		Indicates whether this object is duplicated on this shard (Y) or not (N)
SHARDED	VARCHAR2(1)		Indicates whether this object is sharded (Y) or not (N)
IMPORTED_OBJECT	VARCHAR2(1)		Indicates whether this object is imported (Y) or not (N)
SYNCHRONOUS_DUPLICATED	VARCHAR2(1)		Indicates whether this object is a synchronous duplicated table (Y) or not (N)
CREATED_APPID	NUMBER		ID of the Application that created the object
CREATED_VSNID	NUMBER		ID of the Application Version that created the object
MODIFIED_APPID	NUMBER		ID of the Application that last modified the object
MODIFIED_VSNID	NUMBER		ID of the Application Version that last modified the object

6.130 DBA_JAVA_ARGUMENTS

DBA_JAVA_ARGUMENTS displays argument information about all stored Java classes in the database. Its columns are the same as those in ALL_JAVA_ARGUMENTS.

> **See Also:**
>
> "ALL_JAVA_ARGUMENTS"

6.131 DBA_JAVA_CLASSES

DBA_JAVA_CLASSES displays class level information about all stored Java classes in the database. Its columns are the same as those in ALL_JAVA_CLASSES.

> **See Also:**
>
> "ALL_JAVA_CLASSES"

6.132 DBA_JAVA_COMPILER_OPTIONS

DBA_JAVA_COMPILER_OPTIONS displays information about all native compiler options in the database. Its columns are the same as those in ALL_JAVA_COMPILER_OPTIONS.

> **See Also:**
>
> "ALL_JAVA_COMPILER_OPTIONS"

6.133 DBA_JAVA_DERIVATIONS

DBA_JAVA_DERIVATIONS displays mapping information about Java source objects and their derived Java class objects and Java resource objects for all Java classes in the database. Its columns are the same as those in ALL_JAVA_DERIVATIONS.

> **See Also:**
>
> "ALL_JAVA_DERIVATIONS"

6.134 DBA_JAVA_FIELDS

DBA_JAVA_FIELDS displays field information about all stored Java classes in the database. Its columns are the same as those in ALL_JAVA_FIELDS.

> **See Also:**
>
> "ALL_JAVA_FIELDS"

6.135 DBA_JAVA_IMPLEMENTS

DBA_JAVA_IMPLEMENTS describes interfaces implemented by all stored Java classes in the database. Its columns are the same as those in ALL_JAVA_IMPLEMENTS.

> **See Also:**
>
> "ALL_JAVA_IMPLEMENTS"

6.136 DBA_JAVA_INNERS

DBA_JAVA_INNERS displays information about inner classes referred to by all stored Java classes in the database. Its columns are the same as those in ALL_JAVA_INNERS.

> **See Also:**
>
> "ALL_JAVA_INNERS"

6.137 DBA_JAVA_LAYOUTS

DBA_JAVA_LAYOUTS displays class layout information about all stored Java classes in the database. Its columns are the same as those in ALL_JAVA_LAYOUTS.

> **See Also:**
>
> "ALL_JAVA_LAYOUTS"

6.138 DBA_JAVA_METHODS

DBA_JAVA_METHODS displays method information about all stored Java classes in the database. Its columns are the same as those in ALL_JAVA_METHODS.

> **See Also:**
>
> "ALL_JAVA_METHODS"

6.139 DBA_JAVA_NCOMPS

DBA_JAVA_NCOMPS displays ncomp-related information about all Java classes in the database. Its columns are the same as those in ALL_JAVA_NCOMPS.

> **See Also:**
>
> "ALL_JAVA_NCOMPS"

6.140 DBA_JAVA_POLICY

DBA_JAVA_POLICY describes Java security permissions for all users in the database.

Related View

USER_JAVA_POLICY describes Java security permissions for the current user.

Column	Datatype	NULL	Description
KIND	VARCHAR2(8)		Indicates whether the permission is a positive (GRANT) or a limitation (RESTRICT)
GRANTEE	VARCHAR2(128)	NOT NULL	Name of the user, schema, or role to which the permission object is assigned
TYPE_SCHEMA	VARCHAR2(128)	NOT NULL	Schema in which the permission object is loaded
TYPE_NAME	VARCHAR2(4000)		Permission class type, which is designated by a string containing the full class name, such as, java.io.FilePermission
NAME	VARCHAR2(4000)		Target attribute (name) of the permission object. This name is used when defining the permission.
ACTION	VARCHAR2(4000)		Action attribute for this permission. Many permissions expect a null value if no action is appropriate for the permission.
ENABLED	VARCHAR2(8)		Indicates whether the permission is enabled (ENABLED) or disabled (DISABLED)
SEQ	NUMBER		Sequence number used to identify this row. This number should be supplied when disabling, enabling, or deleting the permission.

See Also:

"USER_JAVA_POLICY"

6.141 DBA_JAVA_RESOLVERS

DBA_JAVA_RESOLVERS displays information about resolvers of all Java classes in the database. Its columns are the same as those in ALL_JAVA_RESOLVERS.

See Also:

"ALL_JAVA_RESOLVERS"

6.142 DBA_JAVA_THROWS

DBA_JAVA_THROWS displays information about exceptions thrown from methods of all Java classes in the database. Its columns are the same as those in ALL_JAVA_THROWSa.

> **See Also:**
>
> "ALL_JAVA_THROWS"

6.143 DBA_JOBS

DBA_JOBS describes all jobs in the database.

Related View

USER_JOBS describes the jobs owned by the current user.

Column	Datatype	NULL	Description
JOB	NUMBER	NOT NULL	Identifier of job. Neither import/export nor repeated executions change this value.
LOG_USER	VARCHAR2(128)	NOT NULL	Login user when the job was submitted
PRIV_USER	VARCHAR2(128)	NOT NULL	User whose default privileges apply to this job
SCHEMA_USER	VARCHAR2(128)	NOT NULL	Default schema used to parse the job For example, if the SCHEMA_USER is SCOTT and you submit the procedure HIRE_EMP as a job, the Oracle Database looks for SCOTT.HIRE_EMP
LAST_DATE	DATE		Date on which this job last successfully executed
LAST_SEC	VARCHAR2(8)		Same as LAST_DATE. This is when the last successful execution started.
THIS_DATE	DATE		Date that this job started executing (usually null if not executing)
THIS_SEC	VARCHAR2(8)		Same as THIS_DATE. This is when the last successful execution started.
NEXT_DATE	DATE	NOT NULL	Date that this job will next be executed
NEXT_SEC	VARCHAR2(8)		Same as NEXT_DATE. The job becomes due for execution at this time.
TOTAL_TIME	NUMBER		Total wall clock time spent by the system on this job (in seconds) when it last executed
BROKEN	VARCHAR2(1)		Y: no attempt is made to run this job N: an attempt is made to run this job
INTERVAL	VARCHAR2(200)	NOT NULL	A date function, evaluated at the start of execution, becomes next NEXT_DATE
FAILURES	NUMBER		Number of times the job has started and failed since its last success
WHAT	VARCHAR2(4000)		Body of the anonymous PL/SQL block that the job executes

Column	Datatype	NULL	Description
NLS_ENV	VARCHAR2(4000)		Session parameters describing the NLS environment of the job
MISC_ENV	RAW(32)		Other session parameters of the session that created the job. The job is run using these parameters.
INSTANCE	NUMBER		ID of the instance that can execute or is executing the job. The default is 0.

> **See Also:**
>
> "USER_JOBS"

6.144 DBA_JOBS_RUNNING

DBA_JOBS_RUNNING lists all jobs that are currently running in the instance.

Column	Datatype	NULL	Description
SID	NUMBER		Identifier of process that is executing the job. See "V$LOCK".
JOB	NUMBER		Identifier of job. This job is currently executing.
FAILURES	NUMBER		Number of times this job started and failed since its last success.
LAST_DATE	DATE		Date that this job last successfully executed.
LAST_SEC	VARCHAR2(8)		Same as LAST_DATE. This is when the last successful execution started.
THIS_DATE	DATE		Date that this job started executing.
THIS_SEC	VARCHAR2(8)		Same as THIS_DATE. This is when the last successful execution started.
INSTANCE	NUMBER		Indicates which instance can execute or is executing the job; the default is 0.

6.145 DBA_JOIN_IND_COLUMNS

DBA_JOIN_IND_COLUMNS describes all join conditions in the database. Its columns are the same as those in ALL_JOIN_IND_COLUMNS.

> **See Also:**
>
> "ALL_JOIN_IND_COLUMNS"

6.146 DBA_JOINGROUPS

DBA_JOINGROUPS describes join groups in the database. A join group is a user-created object that consists of two or more columns that can be meaningfully joined. The maximum number of columns that can be included in a join group is 255.

In certain queries, join groups enable the database to eliminate the performance overhead of decompressing and hashing column values. Join groups require an In-Memory column store (IM column store).

Related View

USER_JOINGROUPS describes join groups belonging to the user. This view does not display the JOINGROUP_OWNER column.

Column	Datatype	NULL	Description
JOINGROUP_OWNER	VARCHAR2(128)	NOT NULL	Join group owner. This is the user that created the join group.
JOINGROUP_NAME	VARCHAR2(128)	NOT NULL	This is the user specified name of the join group. The join group name is specified when the join group is created as part of the CREATE INMEMORY JOIN GROUP statement.
TABLE_OWNER	VARCHAR2(128)	NOT NULL	Table owner
TABLE_NAME	VARCHAR2(128)	NOT NULL	Table name
COLUMN_NAME	VARCHAR2(128)	NOT NULL	Column name
FLAGS	VARCHAR2(6)		Possible values: • MASTER: Indicates which column in the join group is mastering the global dictionary. A join group is a group of columns sharing a global dictionary; the global dictionary is associated with one column and the other columns share the same dictionary. The column with which the global dictionary is associated is called the mastering column. • NULL: Indicates that the column is not mastering the global dictionary.
GD_ADDRESS	RAW(8)		The memory address of the global dictionary. Ideally, all the columns in one join group should have the same global dictionary address (that is, they share the same global structure). This might not always be the case (for example, a column might be added to a join group after it was populated into memory - in which case its GD_ADDRESS field will be NULL). In such cases, you should force re-populate the tables that are part of the join group and check the views after the repopulates complete.
AUTO_CREATED	VARCHAR2(3)		Indicates whether the join group was created by Automatic In-Memory (AIM) (YES) or whether the join group was user-created (NO)
CREATION_DATE	DATE	NOT NULL	Date and time at which the join group was created

See Also:

- "USER_JOINGROUPS"
- *Oracle Database In-Memory Guide* for an introduction to join groups
- *Oracle Database SQL Language Reference* for information about creating a join group using the `CREATE INMEMORY JOIN GROUP` statement

6.147 DBA_JSON_COLLECTION_TABLES

`DBA_JSON_COLLECTION_TABLES` describes all JSON collection tables in the database. Its columns are the same as those in `ALL_JSON_COLLECTION_TABLES`.

Note:

This view is available starting with Oracle Database 23ai.

See Also:

"ALL_JSON_COLLECTION_TABLES"

6.148 DBA_JSON_COLLECTION_VIEWS

`DBA_JSON_COLLECTION_VIEWS` describes all JSON collection views in the database. Its columns are the same as those in `ALL_JSON_COLLECTION_VIEWS`.

Note:

This view is available starting with Oracle Database 23ai, Release Update 23.6.

See Also:

"ALL_JSON_COLLECTION_VIEWS"

6.149 DBA_JSON_COLLECTIONS

DBA_JSON_COLLECTIONS describes all JSON collections in the database. Its columns are the same as those in ALL_JSON_COLLECTIONS.

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

> **See Also:**
>
> "ALL_JSON_COLLECTIONS"

6.150 DBA_JSON_COLUMNS

DBA_JSON_COLUMNS provides information on all JavaScript Object Notation (JSON) columns in the database. Its columns are the same as those in ALL_JSON_COLUMNS.

> **See Also:**
>
> - "ALL_JSON_COLUMNS"
> - *Oracle Database JSON Developer's Guide* for more information about using JSON with Oracle Database

6.151 DBA_JSON_DATAGUIDE_FIELDS

DBA_JSON_DATAGUIDE_FIELDS extracts the path and type information from all the data guides in the database, which are the data guides returned by the DBA_JSON_DATAGUIDE view. Its columns are the same as those in ALL_JSON_DATAGUIDE_FIELDS.

> **See Also:**
>
> "ALL_JSON_DATAGUIDE_FIELDS"

6.152 DBA_JSON_DATAGUIDES

DBA_JSON_DATAGUIDES provides information on all JavaScript Object Notation (JSON) columns in the database that have data guide enabled. Its columns are the same as those in ALL_JSON_DATAGUIDES.

> **See Also:**
>
> "ALL_JSON_DATAGUIDES"

6.153 DBA_JSON_DOMAIN_SCHEMA_COLUMNS

DBA_JSON_DOMAIN_SCHEMA_COLUMNS describes JSON schema constraints on columns of all data use case domains in the database. Its columns are the same as those in ALL_JSON_DOMAIN_SCHEMA_COLUMNS.

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

> **See Also:**
>
> "ALL_JSON_DOMAIN_SCHEMA_COLUMNS"

6.154 DBA_JSON_DUALITY_VIEW_LINKS

DBA_JSON_DUALITY_VIEW_LINKS describes the links associated with all JSON-relational duality views in the database. Its columns are the same as those in ALL_JSON_DUALITY_VIEW_LINKS.

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

> **See Also:**
>
> "ALL_JSON_DUALITY_VIEW_LINKS"

6.155 DBA_JSON_DUALITY_VIEW_TAB_COLS

DBA_JSON_DUALITY_VIEW_TAB_COLS describes the table columns associated with all JSON-relational duality views in the database. Its columns are the same as those in ALL_JSON_DUALITY_VIEW_TAB_COLS.

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

> **See Also:**
>
> "ALL_JSON_DUALITY_VIEW_TAB_COLS"

6.156 DBA_JSON_DUALITY_VIEW_TABS

DBA_JSON_DUALITY_VIEW_TABS describes the tables associated with all JSON-relational duality views in the database. Its columns are the same as those in ALL_JSON_DUALITY_VIEW_TABS.

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

> **See Also:**
>
> "ALL_JSON_DUALITY_VIEW_TABS"

6.157 DBA_JSON_DUALITY_VIEWS

DBA_JSON_DUALITY_VIEWS describes all JSON-relational duality views in the database. Its columns are the same as those in ALL_JSON_DUALITY_VIEWS.

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

> **See Also:**
>
> "ALL_JSON_DUALITY_VIEWS"

6.158 DBA_JSON_INDEXES

DBA_JSON_INDEXES describes all indexes on JSON data in the database. Its columns are the same as those in ALL_JSON_INDEXES.

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

> **See Also:**
>
> "ALL_JSON_INDEXES"

6.159 DBA_JSON_SCHEMA_COLUMNS

DBA_JSON_SCHEMA_COLUMNS describes JSON schema constraints on columns in all tables in the database. Its columns are the same as those in ALL_JSON_SCHEMA_COLUMNS.

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

> **See Also:**
>
> "ALL_JSON_SCHEMA_COLUMNS"

6.160 DBA_KAFKA_APPLICATIONS

DBA_KAFKA_APPLICATIONS describes all Oracle SQL Access to Kafka (OSAK) applications in the database.

Related View

USER_KAFKA_APPLICATIONS describes OSAK applications owned by the current user. This view does not display the OWNER or CLUSTER_ID columns.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)	NOT NULL	Owner of the application
CLUSTER_ID	VARCHAR2(40)	NOT NULL	ID of the OSAK cluster associated with the application
CLUSTER_NAME	VARCHAR2(30)	NOT NULL	Name of the OSAK cluster associated with the application
APPLICATION_NAME	VARCHAR2(30)	NOT NULL	Application name (also used as the Kafka group name)
TOPIC_NAME	VARCHAR2(249)	NOT NULL	Name of the Kafka topic associated with the application
FORMAT	VARCHAR2(4000)		Format of the Kafka topic value data: • AVRO • DSV • JSON
KEY_FORMAT	VARCHAR2(4000)		Format of the Kafka topic key data: • AVRO • DSV • JSON
NUM_VIEWS	NUMBER(38)	NOT NULL	Number of OSAK views for the application
NUM_PARTITIONS	NUMBER(38)	NOT NULL	Number of Kafka partitions associated with the OSAK views for the application
NUM_VIEWS_REQUESTED	NUMBER(38)	NOT NULL	Number of OSAK views requested for the application
APPLICATION_TYPE	VARCHAR2(30)	NOT NULL	Type of application: • LOAD • SEEKABLE • STREAMING
OPTIONS	BLOB	NOT NULL	User-supplied options (excluding AVRO-related schemas)
AVRO_SCHEMA	BLOB		User-supplied AVRO schema for the Kafka topic value data
KEY_AVRO_SCHEMA	BLOB		Reserved for future use

Note:

This view is available starting with Oracle Database 23ai.

See Also:

- "USER_KAFKA_APPLICATIONS"
- "DBMS_KAFKA_APPLICATIONS"

6.161 DBA_KAFKA_CLUSTERS

DBA_KAFKA_CLUSTERS describes the Oracle SQL Access to Kafka (OSAK) clusters for which any database user has READ access.

Related View

USER_KAFKA_CLUSTERS describes the OSAK clusters that are owned by the current user or for which the current user has READ access. This view does not display the OWNER or GRANTEE columns.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)	NOT NULL	Owner of the cluster The value of this column is always SYS.
CLUSTER_ID	VARCHAR2(40)	NOT NULL	System-generated unique cluster ID
STATE	NUMBER(38)	NOT NULL	Current state of the cluster: • 0 - CONNECTED • 1 - MAINTENANCE • 2 - BROKEN • 3 - DEREGISTERED
CLUSTER_NAME	VARCHAR2(30)	NOT NULL	User-supplied name of the cluster
BOOTSTRAP_SERVERS	VARCHAR2(4000)	NOT NULL	Comma-separated list of bootstrap servers for the cluster Each bootstrap server is represented in the following form: *hostname*:*port*
KAFKA_PROVIDER	VARCHAR2(128)	NOT NULL	Provider of the Kafka cluster software: • APACHE - Apache Kafka • OSS - Oracle Cloud Infrastructure Streaming Service
CONNECTION_TS	TIMESTAMP(6)		If GRANTEE is currently connected to the cluster, the date and time at which the connection was started; otherwise NULL
CLUSTER_DESCRIPTION	VARCHAR2(4000)		Optional user-supplied description of the Oracle SQL access to the cluster
GRANTEE	VARCHAR2(128)		User who is granted access to the cluster

Note:

This view is available starting with Oracle Database 23ai.

See Also:

- "USER_KAFKA_CLUSTERS"
- "DBMS_KAFKA_CLUSTERS" for information about viewing all registered OSAK clusters in the database

6.162 DBA_KAFKA_LOAD_METRICS

DBA_KAFKA_LOAD_METRICS displays metrics for DBMS_KAFKA.EXECUTE_LOAD_APP operations on all tables in the database.

Related View

USER_KAFKA_LOAD_METRICS displays metrics for DBMS_KAFKA.EXECUTE_LOAD_APP operations on tables owned by the current user. This view does not display the OWNER or CLUSTER_ID columns.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)	NOT NULL	Owner of the table into which the Kafka data was loaded (TARGET_TABLE)
CLUSTER_ID	VARCHAR2(40)	NOT NULL	ID of the OSAK cluster associated with the EXECUTE_LOAD_APP operation
CLUSTER_NAME	VARCHAR2(30)	NOT NULL	Name of the OSAK cluster associated with the OSAK cluster ID
APPLICATION_NAME	VARCHAR2(30)		Application name (also used as the Kafka group name)
TARGET_TABLE	VARCHAR2(128)	NOT NULL	Name of the table into which the Kafka data was loaded
TARGET_TABLE_SCHEMA	VARCHAR2(128)	NOT NULL	Schema of the table into which the Kafka data was loaded
INSERTED_ROWS	NUMBER(38)		Total number of rows inserted into the table by the load operation
KAFKA_RECORDS	NUMBER(38)		Total number of Kafka records fetched for the load operation
STARTED_TIME	TIMESTAMP(6) WITH TIME ZONE		Date and time at which the Kafka data started being loaded into the table
FINISHED_TIME	TIMESTAMP(6) WITH TIME ZONE		Date and time at which the Kafka data finished being loaded into the table

Note:

This view is available starting with Oracle Database 23ai.

See Also:

- "USER_KAFKA_LOAD_METRICS"
- "DBMS_KAFKA_LOAD_METRICS"
- *Oracle Database PL/SQL Packages and Types Reference* for more information about the DBMS_KAFKA.EXECUTE_LOAD_APP procedure

6.163 DBA_KAFKA_OPS

DBA_KAFKA_OPS describes operations for all Oracle SQL Access to Kafka (OSAK) views in the database.

Related View

USER_KAFKA_OPS describes operations for OSAK views owned by the current user. This view does not display the OWNER or CLUSTER_ID columns.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)	NOT NULL	Owner of the OSAK view
OP_KEY	VARCHAR2(30)	NOT NULL	Unique operation key (replacement for the location file)
OP_TYPE	VARCHAR2(30)	NOT NULL	Generic operation type
OP	VARCHAR2(30)	NOT NULL	Specific type of operation within OP_TYPE
CLUSTER_ID	VARCHAR2(40)	NOT NULL	ID of the OSAK cluster associated with the operation
CLUSTER_NAME	VARCHAR2(30)	NOT NULL	Name of the OSAK cluster associated with the operation
APPLICATION_NAME	VARCHAR2(30)		Name of the OSAK application associated with the operation (also used as the Kafka group name)
TOPIC_NAME	VARCHAR2(249)		Name of the Kafka topic associated with the operation
PARTITION_ID	NUMBER(38)		Partition ID of the Kafka topic associated with the operation
OFFSET	NUMBER(38)		Next offset from which an SEQ operation will start reading
RELATIVE_OFFSET	NUMBER(38)		Number of offsets, including the watermark, from which an SEQ operation will start reading
WATERMARK	VARCHAR2(5)		Basis for the start read offset for an SEQ operation (WMH or WML)
START_OFFSET_EPOCH_TIME	NUMBER(38)		Timestamp (in milliseconds) that determines the next offset from which the operation will start reading This value is used by the DBMS_KAFKA.INIT_OFFSET_TS and DBMS_KAFKA.SEEK_OFFSET_TS procedures.
END_OFFSET_EPOCH_TIME	NUMBER(38)		Timestamp (in milliseconds) that determines the end offset at which the operation will stop reading This value is used by the DBMS_KAFKA.SEEK_OFFSET_TS procedure.

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

> **See Also:**
>
> - "USER_KAFKA_OPS"
> - "DBMS_KAFKA_OPS"
> - *Oracle Database PL/SQL Packages and Types Reference* for more information about the `DBMS_KAFKA.INIT_OFFSET_TS` and `DBMS_KAFKA.SEEK_OFFSET_TS` procedures

6.164 DBA_KAFKA_OPS_RESULTS

`DBA_KAFKA_OPS_RESULTS` displays the results of operations for all Oracle SQL Access to Kafka (OSAK) views in the database.

Related View

`USER_KAFKA_OPS_RESULTS` displays the results of operations for Oracle SQL Access to Kafka (OSAK) views owned by the current user. This view does not display the `OWNER` column.

Column	Datatype	NULL	Description
`OWNER`	`VARCHAR2(128)`	`NOT NULL`	Owner of OSAK view
`OP_KEY`	`VARCHAR2(30)`		Unique operation key (replacement for the location file)
`ROWS_READ`	`NUMBER(38)`		Number of Kafka records read and returned by the operation
`LAST_OFFSET`	`NUMBER(38)`		Kafka offset of the last record read by the operation

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

> **See Also:**
>
> - "USER_KAFKA_OPS_RESULTS"
> - "DBMS_KAFKA_OPS_RESULTS"

6.165 DBA_KAFKA_PARTITIONS

`DBA_KAFKA_PARTITIONS` describes partitions for Kafka topics associated with all Oracle SQL Access to Kafka (OSAK) views in the database.

Related View

USER_KAFKA_PARTITIONS describes partitions for Kafka topics associated with OSAK views owned by the current user. This view does not display the OWNER or CLUSTER_ID columns.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)	NOT NULL	Owner of the OSAK view
CLUSTER_ID	VARCHAR2(40)	NOT NULL	ID of the OSAK cluster associated with the topic/partition
CLUSTER_NAME	VARCHAR2(30)	NOT NULL	User-supplied name of the associated Kafka server/cluster
APPLICATION_NAME	VARCHAR2(30)	NOT NULL	Application name (also used as the Kafka group name)
TOPIC_NAME	VARCHAR2(249)	NOT NULL	Kafka topic name associated with the topic/partition
PARTITION_ID	NUMBER(38)	NOT NULL	Partition ID of the Kafka topic
VIEW_ID	NUMBER(38)	NOT NULL	ID of the OSAK view associated with the topic/partition
VIEW_NAME	VARCHAR2(128)	NOT NULL	Name of the OSAK view associated with the topic/partition
TEMP_TABLE_NAME	VARCHAR2(128)		Name of the OSAK global temporary table associated with the topic/partition The value of this column is null if the application type is LOAD, that is, if the application was created using the DBMS_KAFKA.CREATE_LOAD_APP procedure. The application type is displayed in the APPLICATION_TYPE column of the DBA_KAFKA_APPLICATIONS view.

Note:

This view is available starting with Oracle Database 23ai.

See Also:

- "USER_KAFKA_PARTITIONS"
- "DBMS_KAFKA_PARTITIONS"

6.166 DBA_KGLLOCK

DBA_KGLLOCK lists all the locks and pins held on KGL objects (objects in the Kernel Generic Library cache).

Column	Datatype	NULL	Description
kgllkuse	RAW(4)		Address of the user session that holds the lock or pin
kgllkhdl	RAW(4)		Address of the handle for the KGL object

Column	Datatype	NULL	Description
kgllkmod	NUMBER		Current mode of the lock or pin
kgllkreq	NUMBER		Mode in which the lock or pin was requested
kgllktype	VARCHAR2(4)		Whether this is a lock or a pin

6.167 DBA_LIBRARIES

DBA_LIBRARIES describes all libraries in the database. Its columns are the same as those in ALL_LIBRARIES.

> **See Also:**
>
> "ALL_LIBRARIES"

6.168 DBA_LMT_FREE_SPACE

DBA_LMT_FREE_SPACE describes the free extents in all locally managed tablespaces in the database.

Column	Datatype	NULL	Description
TABLESPACE_ID	NUMBER		Identifier number of the tablespace containing the extent
FILE_ID	NUMBER		File identifier number of the file containing the extent
BLOCK_ID	NUMBER		Starting block number of the extent
BLOCKS	NUMBER		Size of the extent (in Oracle blocks)

6.169 DBA_LMT_USED_EXTENTS

DBA_LMT_USED_EXTENTS describes the extents comprising the segments in all locally managed tablespaces in the database.

Column	Datatype	NULL	Description
SEGMENT_FILEID	NUMBER		File number of the segment header of the extent
SEGMENT_BLOCK	NUMBER		Block number of the segment header of the extent
TABLESPACE_ID	NUMBER		Identifier number of the tablespace containing the extent
EXTENT_ID	NUMBER		Extent number in the segment
FILEID	NUMBER		File identifier number of the file containing the extent
BLOCK	NUMBER		Starting block number of the extent
LENGTH	NUMBER		Number of blocks in the extent

6.170 DBA_LOB_PARTITIONS

DBA_LOB_PARTITIONS displays all LOB partitions in the database. Its columns are the same as those in ALL_LOB_PARTITIONS.

> **See Also:**
>
> "ALL_LOB_PARTITIONS"

6.171 DBA_LOB_SUBPARTITIONS

DBA_LOB_SUBPARTITIONS displays partition-level attributes of all LOB data subpartitions in the database. Its columns are the same as those in ALL_LOB_SUBPARTITIONS.

> **See Also:**
>
> "ALL_LOB_SUBPARTITIONS"

6.172 DBA_LOB_TEMPLATES

DBA_LOB_TEMPLATES describes all LOB subpartition templates in the database. Its columns are the same as those in ALL_LOB_TEMPLATES.

> **See Also:**
>
> "ALL_LOB_TEMPLATES"

6.173 DBA_LOBS

DBA_LOBS displays the BLOBs and CLOBs contained in all tables in the database. BFILEs are stored outside the database, so they are not described by this view. This view's columns are the same as those in ALL_LOBS.

> **See Also:**
>
> "ALL_LOBS"

6.174 DBA_LOCK

DBA_LOCK lists all locks or latches held in the database, and all outstanding requests for a lock or latch.

Column	Datatype	NULL	Description
SESSION_ID	NUMBER		Session holding or acquiring the lock
LOCK_TYPE	VARCHAR2(26)		Lock type **See Also:** For a listing of lock types, see Oracle Enqueue Names
MODE_HELD	VARCHAR2(40)		Lock mode
MODE_REQUESTED	VARCHAR2(40)		Lock mode requested
LOCK_ID1	VARCHAR2(40)		Type-specific lock identifier, part 1
LOCK_ID2	VARCHAR2(40)		Type-specific lock identifier, part 2
LAST_CONVERT	NUMBER		The last convert
BLOCKING_OTHERS	VARCHAR2(40)		Whether the lock is currently blocking others
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

6.175 DBA_LOCK_INTERNAL

DBA_LOCK_INTERNAL displays a row for each lock or latch that is being held, and one row for each outstanding request for a lock or latch.

Column	Datatype	NULL	Description
SESSION_ID	NUMBER		Session holding or acquiring the lock
LOCK_TYPE	VARCHAR2(56)		Lock type **See Also:** For a listing of lock types, see Oracle Enqueue Names
MODE_HELD	VARCHAR2(40)		Lock mode
MODE_REQUESTED	VARCHAR2(40)		Lock mode requested
LOCK_ID1	VARCHAR2(1130)		Type-specific lock identifier, part 1
LOCK_ID2	VARCHAR2(40)		Type-specific lock identifier, part 2

Column	Datatype	NULL	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

6.176 DBA_LOCKDOWN_PROFILES

DBA_LOCKDOWN_PROFILES provides information about lockdown profiles.

The PRIVATE_DBAAS, PUBLIC_DBAAS, and SAAS lockdown profiles are empty placeholder profiles for the lockdown profiles of their corresponding deployment type. You can modify and add restrictions to these profiles based on their deployment purpose. For example, if you have a Software as a Service (SAAS) application, you can modify the SAAS lockdown profile and use it. You can also delete and re-create these profiles.

Column	Datatype	NULL	Description
PROFILE_NAME	VARCHAR2(128)	NOT NULL	Name of the lockdown profile
RULE_TYPE	VARCHAR2(128)		Rule type. A lockdown profile is used to restrict operations that can be performed by users connected to a given PDB. It provides the ability to add or remove different types of rules like STATEMENT, FEATURES or OPTIONS which will be restricted in the PDB.
RULE	VARCHAR2(128)		Rule to be enabled or disabled
CLAUSE	VARCHAR2(128)		Clause of the statement
CLAUSE_OPTION	VARCHAR2(128)		Option of the clause
OPTION_VALUE	VARCHAR2(4000)		Value of the option
MIN_VALUE	VARCHAR2(4000)		Minimum value allowed for the option
MAX_VALUE	VARCHAR2(4000)		Maximum value allowed for the option
LIST	VARCHAR2(4000)		List of allowed values for the option
STATUS	VARCHAR2(7)		Status of the lockdown profile: • ENABLE • DISABLE • EMPTY
USERS	VARCHAR2(6)		User type. Possible values: • COMMON • LOCAL • ALL
EXCEPT_USERS	CLOB		For internal use only

See Also:

- "PDB_LOCKDOWN"
- *Oracle Multitenant Administrator's Guide* for an introduction to PDB lockdown profiles
- *Oracle Database SQL Language Reference* for more information about creating lockdown profiles
- *Oracle Database SQL Language Reference* for more information about dropping lockdown profiles
- *Oracle Database SQL Language Reference* for more information about altering lockdown profiles

6.177 DBA_LOCKS

`DBA_LOCKS` is a synonym for `DBA_LOCK`.

See Also:

"DBA_LOCK"

6.178 DBA_LOG_GROUP_COLUMNS

`DBA_LOG_GROUP_COLUMNS` describes all columns in the database that are specified in log groups. Its columns are the same as those in `ALL_LOG_GROUP_COLUMNS`.

See Also:

"ALL_LOG_GROUP_COLUMNS"

6.179 DBA_LOG_GROUPS

`DBA_LOG_GROUPS` describes log group definitions on all tables in the database. Its columns are the same as those in `ALL_LOG_GROUPS`.

See Also:

"ALL_LOG_GROUPS"

6.180 DBA_LOGMNR_DICTIONARY_BUILDLOG

DBA_LOGMNR_DICTIONARY_BUILDLOG describes all successful LogMiner dictionary builds available for GoldenGate REGISTER EXTRACT.

When this view is queried from a PDB, it returns only LogMiner dictionary builds performed in that PDB.

Column	Datatype	NULL	Description
NAME	VARCHAR2(384)		User-supplied name of the LogMiner dictionary build
DATE_OF_BUILD	VARCHAR2(20)		Date and time at which the LogMiner dictionary build was performed
START_SCN	NUMBER		Begin SCN of the LogMiner dictionary build operation
END_SCN	NUMBER		End SCN of the LogMiner dictionary build operation
BUILD_TYPE	VARCHAR2(20)		Internal string that identifies how the LogMiner dictionary build was initiated
BUILD_OP	NUMBER		Internal number that identifies how the LogMiner dictionary build was initiated
CONTAINER_ID	NUMBER		ID of the container in which the LogMiner dictionary build was performed (the CDB root or a PDB)
CONTAINER_UID	NUMBER		Unique number that identifies the container in which the LogMiner dictionary build was performed (the CDB root or a PDB)
CONTAINER_NAME	VARCHAR2(384)		Name of the container in which the LogMiner dictionary build was performed (the CDB root or a PDB)
RESETLOGS_CHANGE#	NUMBER		SCN that identifies the redo branch of the LogMiner dictionary build

6.181 DBA_LOGMNR_LOG

DBA_LOGMNR_LOG displays all archived logs registered with active LogMiner persistent sessions in the database.

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.

Column	Datatype	NULL	Description
LOGMNR_SESSION_ID	NUMBER	NOT NULL	Unique identifier of the persistent session
NAME	VARCHAR2(513)		Name of the archived log
DBID	NUMBER	NOT NULL	Database identifier that produced the archived log
RESETLOGS_SCN	NUMBER	NOT NULL	SCN at which resetlogs operation was performed at the source database generating the archived log
RESETLOGS_TIME	NUMBER	NOT NULL	Timestamp at which resetlogs operation was performed at the source database generating the archived log
MODIFIED_TIME	DATE		Time at which the archived log was registered with LogMiner

Column	Datatype	NULL	Description
THREAD#	NUMBER	NOT NULL	Redo thread at the source database that generated the archived log
SEQUENCE#	NUMBER	NOT NULL	Logfile sequence number
FIRST_SCN	NUMBER	NOT NULL	Lowest SCN of the redo record contained in the logfile
NEXT_SCN	NUMBER		Highest possible SCN of the redo record contained in the logfile
FIRST_TIME	DATE		Time of the first redo record contained in the logfile
NEXT_TIME	DATE		Time of the last redo record contained in the logfile
DICTIONARY_BEGIN	VARCHAR2(3)		Indicates whether the archived log contains the beginning of a LogMiner dictionary (YES) or not (NO)
DICTIONARY_END	VARCHAR2(3)		Indicates whether the archived log contains the end of a LogMiner dictionary (YES) or not (NO)
KEEP	VARCHAR2(3)		Indicates whether the logfile is still required for this LogMiner session (YES) or not (NO)
SUSPECT	VARCHAR2(3)		Indicates whether the archived log content was deemed to be corrupt or the archived log is partially filled (YES) or not (NO)

6.182 DBA_LOGMNR_PURGED_LOG

DBA_LOGMNR_PURGED_LOG displays archived redo log files that have been applied to the logical standby database and can be deleted because they are no longer needed.

Files in this view are refreshed as a result of executing the DBMS_LOGSTDBY.PURGE_SESSION PL/SQL procedure for Oracle Data Guard SQL Apply:

Column	Datatype	NULL	Description
FILE_NAME	VARCHAR2(513)		Fully qualified names of the archived redo log files that are no longer needed by SQL Apply and can be deleted from the operating system

See Also:

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

6.183 DBA_LOGMNR_SESSION

DBA_LOGMNR_SESSION displays all active LogMiner persistent sessions in the database.

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.

Column	Datatype	NULL	Description
ID	NUMBER	NOT NULL	Unique session identifier

Column	Datatype	NULL	Description
NAME	VARCHAR2(128)	NOT NULL	Unique session name
SOURCE_DATABASE	VARCHAR2(128)		Global name of the source database whose archived logs are to be mined in this persistent LogMiner session
SOURCE_DBID	NUMBER		Database ID of the source database
SOURCE_RESETLOGS_SCN	NUMBER		Resetlogs SCN associated with the incarnation of the source database whose archived logs are mined
SOURCE_RESETLOGS_TIME	NUMBER		Resetlogs time associated with the incarnation of the source database whose archived logs are mined
FIRST_SCN	NUMBER		Only modifications that occurred on or after this SCN can be mined using this persistent session
END_SCN	NUMBER		No modifications that occurred on or after this SCN can be mined using this persistent session
RELOCATION_SCN	NUMBER		SCN below which GoldenGate capture cannot be altered This value is set when a PDB containing GoldenGate capture has been relocated to another CDB or switched to its Data Guard PDB, or when the CDB is upgraded using the DBMS_ROLLING package.
WAIT_FOR_LOG	VARCHAR2(3)		Indicates whether the persistent session waits for RFS to register new archived logs or to fill gaps (YES) or not (NO)
HOT_MINE	VARCHAR2(3)		Indicates whether real-time mining is on (YES) or not (NO)
SAFE_PURGE_SCN	NUMBER		Persistent session can safely be purged up to this SCN
CHECKPOINT_SCN	NUMBER		SCN at which the latest checkpoint is taken by the persistent LogMiner session
PURGE_SCN	NUMBER		The session has been purged up to this SCN

6.184 DBA_LOGSTDBY_EVENTS

DBA_LOGSTDBY_EVENTS displays information about the activity of the logical standby database system.

It can be used to determine the cause of failures that occur when applying redo data to logical standby databases. This view is for logical standby databases only.

Column	Datatype	NULL	Description
EVENT_TIME	DATE		Time when the event was logged
EVENT_TIMESTAMP	TIMESTAMP(6)	NOT NULL	Timestamp when the event was logged
START_SCN	NUMBER		The SCN at which the associated transaction started at the primary database. This SCN refers to the system change number at the primary database.

Column	Datatype	NULL	Description
CURRENT_SCN	NUMBER		SCN associated with the change at the primary database. If a failure occurred, then examine this column to determine which archived log file contains the source of the failure (for example, an unsupported record).
COMMIT_SCN	NUMBER		SCN value on which the change was committed at the primary database
XIDUSN	NUMBER		Transaction ID undo segment number at the primary database of the associated transaction
XIDSLT	NUMBER		Transaction ID slot number at the primary database of the associated transaction
XIDSQN	NUMBER		Transaction ID sequence number at the primary database of the associated transaction
EVENT	CLOB		Statement that was being processed when the failure occurred
STATUS_CODE	NUMBER		Status (or Oracle error code) belonging to the STATUS message
STATUS	VARCHAR2(2000)		Description of the current activity of the process or the reason why the apply operation stopped
SRC_CON_NAME	VARCHAR2(384)		Identifies the PDB name at the primary database where the transaction was executed
SRC_CON_ID	NUMBER		Contains the PDB ID (the PDB_ID column from the DBA_PDBS view) of the primary database where the associated change was generated.

> **Note:**
>
> In a CDB, this view shows data only when queried in the root.

6.185 DBA_LOGSTDBY_HISTORY

DBA_LOGSTDBY_HISTORY displays the history of switchovers and failovers in a Data Guard configuration.

It does this by showing the complete sequence of redo log streams processed or created on the local system, across all role transitions. (After a role transition, a new log stream is started and the log stream sequence number is incremented by the new primary database.). This view is for logical standby databases only.

Column	Datatype	NULL	Description
STREAM_SEQUENCE#	NUMBER		Lists the sequence numbers for all log streams created or applied on the local system. **Note**: A value of 0 indicates an unknown sequence order; this is reserved for future log streams.

Column	Datatype	NULL	Description
STATUS	VARCHAR2(16)		Description of the log stream processing: • Past - The log stream has already been processed • Immediate Past - This is the most recently processed log stream; its status is transitioning from Current to Past • Current - The log stream is currently being processed • Immediate Future - This is the next log stream to be processed; its status is transitioning from Future to Current • Future - The log stream will be processed
SOURCE	VARCHAR2(5)		Describes how the log stream was started: • RFS - The RFS process created the log stream • User - A user registered the initial log file for the log stream • Synch - A user issued the ALTER DATABASE START LOGICAL STANDBY APPLY NEW PRIMARY DDL statement • Redo - The log stream information was recorded in the redo log
DBID	NUMBER		Database identifier of the primary database that created the log stream
FIRST_CHANGE#	NUMBER		Lowest system change number (SCN) in the current log file
LAST_CHANGE#	NUMBER		Highest system change number (SCN) in the current log file
FIRST_TIME	DATE		Time of the first SCN entry (FIRST_CHANGE#) in the current log file
LAST_TIME	DATE		Time of the last SCN entry (LAST_CHANGE#) in the current log file
DGNAME	VARCHAR2(255)		Unique database name (DB_UNIQUE_NAME) of the primary database that produced the log stream. See V$DATAGUARD_CONFIG to display all database DB_UNIQUE_NAME values defined in the Data Guard configuration.
MERGE_CHANGE#	NUMBER		SCN that should be used to flashback a failed primary (that created the log stream) or to flashback a bystander logical standby database following a failover, in the context of the associated redo log stream. It is the SCN up to which redo for the associated log stream can be merged safely in all databases using local copies of archived logs received from the primary database. In order to apply changes beyond this following a failover, you will need to fetch and mine the redo logs from the failover target.

Column	Datatype	NULL	Description
PROCESSED_CHANGE#	NUMBER		Strict upper bound on the SCN up to which SQL Apply has applied redo records before it switched to a new log stream (either because it was activated and became the primary database, or in the case of a bystander logical standby database where it switched to a new log stream to accommodate a new primary database).

> **Note:**
>
> In a CDB, this view shows data only when queried in the root.

6.186 DBA_LOGSTDBY_LOG

DBA_LOGSTDBY_LOG displays information about the logs registered for a logical standby database.

This view is for logical standby databases only.

Column	Datatype	NULL	Description
THREAD#	NUMBER	NOT NULL	Thread ID of the archive log. The THREAD number is 1 for a single instance. For Real Application Clusters, this column will contain different numbers.
RESETLOGS_CHANGE#	NUMBER	NOT NULL	Start SCN of the branch
RESETLOGS_ID	NUMBER	NOT NULL	Resetlogs identifier (a numeric form of the timestamp of the branch)
SEQUENCE#	NUMBER	NOT NULL	Sequence number of the archive log file
FIRST_CHANGE#	NUMBER	NOT NULL	System change number (SCN) of the current archive log
NEXT_CHANGE#	NUMBER		SCN of the next archive log
FIRST_TIME	DATE		Date of the current archive log
NEXT_TIME	DATE		Date of the next archive log
FILE_NAME	VARCHAR2(513)		Name of the archive log
TIMESTAMP	DATE		Time when the archive log was registered
DICT_BEGIN	VARCHAR2(3)		Indicates whether the beginning of the dictionary build is in this archive log (YES) or not (NO)
DICT_END	VARCHAR2(3)		Indicates whether the end of the dictionary build is in this archive log (YES) or not (NO)

Column	Datatype	NULL	Description
APPLIED	VARCHAR2(8)		Indicates primarily whether a given foreign archived log has been applied fully by SQL Apply: • YES - SQL Apply has fully applied the foreign archived log and no longer needs it • CURRENT - SQL Apply is currently applying changes contained in the foreign archived log • NO - SQL Apply has not started applying any changes contained in the foreign archived log • FETCHING - SQL Apply encountered a corruption while reading redo records from this foreign archived log, and is currently using the automatic gap resolution to refetch a new copy of the log from the primary database • CORRUPT - SQL Apply encountered a corruption while reading redo records from this foreign archived log, and refetching a new copy of the archived log did not resolve the problem. SQL Apply will not refetch a new copy of this archived log automatically, and will require user intervention to manually register a new copy of the foreign archived log.
BLOCKS	NUMBER		Number of blocks in the log
BLOCK_SIZE	NUMBER		Size of each block in the log

Note:

The SCN values in this view correlate to the SCN values shown in the DBA_LOGSTDBY_PROGRESS view.

Note:

In a CDB, this view shows data only when queried in the root.

6.187 DBA_LOGSTDBY_NOT_UNIQUE

DBA_LOGSTDBY_NOT_UNIQUE displays all tables that have no primary and no non-null unique indexes.

Most of the tables displayed by this view are supported because their columns contain enough information to be maintained in a logical standby database. Some tables, however, cannot be supported because their columns do not contain the necessary information. Unsupported tables usually contain a column defined using an unsupported data type.

In a CDB, the data displayed pertains to the container in which the view is queried.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)		Schema name of the non-unique table

Column	Datatype	NULL	Description
TABLE_NAME	VARCHAR2(128)		Table name of the non-unique table
BAD_COLUMN	VARCHAR2(1)		• Y - Table column is defined using an unbounded data type, such as LONG or BLOB. If two rows in the table match except in their LOB columns, then the table cannot be maintained properly. Log apply services will attempt to maintain these tables, but you must ensure the application does not allow uniqueness only in the unbounded columns. • N - Enough column information is present to maintain the table in the logical standby database but the log transport services and log apply services would run more efficiently if you added a primary key. You should consider adding a disabled RELY constraint to these tables.

6.188 DBA_LOGSTDBY_PARAMETERS

DBA_LOGSTDBY_PARAMETERS displays the list of parameters used by SQL apply for logical standby databases.

This view is for logical standby databases only.

Column	Datatype	NULL	Description
NAME	VARCHAR2(64)		Name of the parameter: • MAX_SGA - System global area (SGA) allocated for the log apply services cache (in megabytes) • MAX_SERVERS - Number of processes used by SQL Apply services • PREPARE_SERVERS - Controls the number of parallel execution servers used to prepare changes • APPLY_SERVERS - Controls the number of parallel execution servers used to apply changes • MAX_EVENTS_RECORDED - Number of events stored in the DBA_LOGSTDBY_EVENTS view • RECORD_SKIP_ERRORS - Indicates records that are skipped • RECORD_SKIP_DDL - Indicates skipped DDL statements • RECORD_APPLIED_DDL - Indicates applied DDL statements • RECORD_UNSUPPORTED_OPERATIONS - Shows whether SQL Apply will capture information about transactions that did unsupported operations at the primary database in the DBA_LOGSTDBY_EVENTS view • EVENT_LOG_DEST - Indicates where SQL Apply records the occurrence of an interesting event • LOG_AUTO_DELETE - Shows whether SQL Apply will automatically delete remote archived logs received from the primary database, once the contents of the logs are applied at the logical standby database. • LOG_AUTO_DEL_RETENTION_TARGET - How many minutes a remote archived log received from the primary database will be retained at the logical standby database, once the contents of the log are applied by SQL Apply. • PRESERVE_COMMIT_ORDER - Shows whether transactions are committed at the logical standby database in the same order that they were committed at the primary database
VALUE	VARCHAR2(2000)		Value of the parameter
UNIT	VARCHAR2(64)		Unit of the value, if applicable
SETTING	VARCHAR2(64)		Possible values are as follows: • SYSTEM - Parameter value was not explicitly set by the user. However, the user can change it with an appropriate call to the APPLY_SET procedure. • USER - Parameter value was explicitly set by the user
DYNAMIC	VARCHAR2(64)		YES if the parameter can be set dynamically (that is, without having to stop SQL Apply) NO if setting the parameter requires that SQL Apply be stopped

> **Note:**
>
> In a CDB, this view shows data when queried in the root.

6.189 DBA_LOGSTDBY_PLSQL_MAP

DBA_LOGSTDBY_PLSQL_MAP shows the mapping between a supported user invokable (/external) PL/SQL procedure to the corresponding replicated internal PL/SQL procedure.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)	NOT NULL	Owner name of the procedure
PKG_NAME	VARCHAR2(128)		Package name of the user invokable procedure
PROC_NAME	VARCHAR2(128)		Procedure name of the user invokable procedure
INTERNAL_PKG_NAME	VARCHAR2(128)		Package name of the internal procedure
INTERNAL_PROC_NAME	VARCHAR2(128)		Procedure name of the internal procedure

> **Note:**
>
> In a CDB, this view shows data when queried in the root or a PDB.

6.190 DBA_LOGSTDBY_PLSQL_SUPPORT

DBA_LOGSTDBY_PLSQL_SUPPORT shows the PL/SQL packages that are only supported during rolling operations.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)	NOT NULL	Owner name of the package
PKG_NAME	VARCHAR2(128)		Package name of the user invokable procedure
SUPPORT_LEVEL	VARCHAR2(12)		Logical standby PL/SQL support level for the package: • ALWAYS - PL/SQL replication is always supported for this package, whether it is called inside or outside of DBMS_ROLLING • DBMS_ROLLING: PL/SQL replication is supported only when the procedure is called inside DBMS_ROLLING

> **Note:**
>
> In a CDB, this view shows data when queried in the root or a PDB.

> **See Also:**
>
> *Oracle Database PL/SQL Packages and Types Reference* for more information about the `DBMS_ROLLING` package

6.191 DBA_LOGSTDBY_PROGRESS

`DBA_LOGSTDBY_PROGRESS` is deprecated. The information that was provided in this view is now provided in the `V$LOGSTDBY_PROGRESS` view.

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

6.192 DBA_LOGSTDBY_SKIP

`DBA_LOGSTDBY_SKIP` displays the skip rules that are used by SQL Apply.

This view is for logical standby databases only.

Column	Datatype	NULL	Description
ERROR	VARCHAR2(1)		Indicates how the skip rule was created: • Y - For rules from DBMS_LOGSTDBY.SKIP_ERROR • N - For rules from DBMS_LOGSTDBY.SKIP
STATEMENT_OPT	VARCHAR2(128)		Specifies the type of statement that should be skipped
OWNER	VARCHAR2(128)		Name of the schema under which the skip option should be used
NAME	VARCHAR2(261)		Name of the object that is being skipped
USE_LIKE	VARCHAR2(1)		Indicates whether the statement should use a SQL wildcard search when matching names (Y) or not (N)
ESC	VARCHAR2(1)		Escape character used when performing wildcard matches
PROC	VARCHAR2(392)		Name of a stored procedure that will be executed when processing the skip option

> **See Also:**
>
> - *Oracle Database PL/SQL Packages and Types Reference* for more information about the `DBMS_LOGSTDBY.SKIP_ERROR` procedure
> - *Oracle Database PL/SQL Packages and Types Reference* for more information about the `DBMS_LOGSTDBY.SKIP` procedure

6.193 DBA_LOGSTDBY_SKIP_TRANSACTION

DBA_LOGSTDBY_SKIP_TRANSACTION displays the skip settings chosen. This view is for logical standby databases only.

Column	Datatype	NULL	Description
XIDUSN	NUMBER		Transaction ID undo segment number
XIDSLT	NUMBER		Transaction ID slot number
XIDSQN	NUMBER		Transaction ID sequence number
CON_NAME	VARCHAR2(384)		Container name

6.194 DBA_LOGSTDBY_SUPPORT_MODE

DBA_LOGSTDBY_SUPPORT_MODE displays information about whether tables in the database are supported for logical standby.

The query results for this view depend on whether the database is currently undergoing a manual rolling upgrade performed using transient logical standby databases. If the database *is* undergoing such an upgrade, then this view displays information about whether tables in the database are supported for transient logical standby. If the database *is not* undergoing such an upgrade, then this view displays information about whether tables in the database are supported for logical standby.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)		Table owner
TABLE_NAME	VARCHAR2(128)		Table name
SUPPORT_MODE	VARCHAR2(11)		If the database is currently undergoing a manual rolling upgrade performed using transient logical standby databases, then this column indicates whether the table is supported for transient logical standby. Otherwise, this column indicates whether the table is supported for logical standby. Possible values: • SUPPORTED - The table is fully supported • INTERNAL - The table is not supported because it contains data that should not be replicated. Such tables include mapping tables for index-organized tables, storage tables for nested tables, materialized view logs, secondary objects associated with domain indexes, and temporary tables. • UNSUPPORTED - The table is not supported because it contains a data type that is not supported or uses a feature that is not supported

Column	Datatype	NULL	Description
EXPLANATION	VARCHAR2(4000)		Reason the table is not fully supported for transient logical standby This column is populated only when all of the following conditions are met: • The database is currently undergoing a manual rolling upgrade using transient logical standby databases • The value of the COMPATIBLE initialization parameter is 20.0 or higher • The value of the SUPPORT_MODE column is INTERNAL or UNSUPPORTED

6.195 DBA_LOGSTDBY_UNSUPPORTED

DBA_LOGSTDBY_UNSUPPORTED displays the schemas, tables, and columns in those tables that contain data types that are unsupported by logical standby.

The query results for this view depend on whether the database is currently undergoing a manual rolling upgrade performed using transient logical standby databases. If the database *is* undergoing such an upgrade, then this view displays information about tables in the database that are unsupported for transient logical standby. If the database *is not* undergoing such an upgrade, then this view displays information about tables in the database are unsupported for logical standby.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)		Owner of the unsupported table
TABLE_NAME	VARCHAR2(128)		Name of the unsupported table
COLUMN_NAME	VARCHAR2(128)		Name of the unsupported column
ATTRIBUTES	VARCHAR2(4000)		If the database is currently undergoing a manual rolling upgrade performed using transient logical standby databases and the value of the COMPATIBLE initialization parameter is 20.0 or higher, then this column displays the reason the table is unsupported by SQL Apply.
DATA_TYPE	VARCHAR2(106)		Data type of the unsupported column

Note:

When the value of the COMPATIBLE initialization parameter is 20.0 or higher, this view displays all columns in unsupported tables. Otherwise, this view displays only the unsupported columns in unsupported tables.

> **Note:**
>
> A rolling upgrade performed using the `DBMS_ROLLING` PL/SQL package supports more object types than a manual rolling upgrade performed using transient logical standby databases.

> **See Also:**
>
> - "DBA_ROLLING_UNSUPPORTED" for more information about determining unsupported data types for a rolling upgrade using the `DBMS_ROLLING` package
> - *Oracle Data Guard Concepts and Administration* for more information about rolling operations
> - *Oracle Data Guard Concepts and Administration* for more information about unsupported tables for rolling upgrade operations
> - *Oracle Database PL/SQL Packages and Types Reference* for more information about the `DBMS_ROLLING` package

6.196 DBA_LOGSTDBY_UNSUPPORTED_TABLE

`DBA_LOGSTDBY_UNSUPPORTED_TABLE` displays the data tables that are unsupported by logical standby.

The query results for this view depend on whether the database is currently undergoing a manual rolling upgrade performed using transient logical standby databases. If the database *is* undergoing such an upgrade, then this view displays information about tables in the database that are unsupported for transient logical standby. If the database *is not* undergoing such an upgrade, then this view displays information about tables in the database are unsupported for logical standby.

The data displayed pertains to the container in which the view is queried.

Column	Datatype	NULL	Description
`OWNER`	`VARCHAR2(128)`		Owner of the unsupported table
`TABLE_NAME`	`VARCHAR2(128)`		Name of the unsupported table
`EXPLANATION`	`VARCHAR2(4000)`		If the database is currently undergoing a manual rolling upgrade performed using transient logical standby databases, then this column displays the reason the table is unsupported for transient logical standby. Otherwise, this column displays the reason the table is unsupported for logical standby. This column is populated only when the value of the `COMPATIBLE` initialization parameter is `20.0` or higher.

6.197 DBA_MEASURE_FOLDER_CONTENTS

DBA_MEASURE_FOLDER_CONTENTS describes the contents of all OLAP measure folders in the database. Its columns are the same as those in ALL_MEASURE_FOLDER_CONTENTS.

> **See Also:**
>
> "ALL_MEASURE_FOLDER_CONTENTS"

6.198 DBA_MEASURE_FOLDER_SUBFOLDERS

DBA_MEASURE_FOLDER_SUBFOLDERS describes the OLAP measure folders contained within the database OLAP measure folders. Its columns are the same as those in ALL_MEASURE_FOLDER_SUBFOLDERS.

> **See Also:**
>
> "ALL_MEASURE_FOLDER_SUBFOLDERS"

6.199 DBA_MEASURE_FOLDERS

DBA_MEASURE_FOLDERS describes all OLAP measure folders in the database. Its columns are the same as those in ALL_MEASURE_FOLDERS.

> **See Also:**
>
> "ALL_MEASURE_FOLDERS"

6.200 DBA_METADATA_PROPERTIES

DBA_METADATA_PROPERTIES describes OLAP metadata properties in the database. Its columns are the same as those in ALL_METADATA_PROPERTIES.

> **See Also:**
>
> "ALL_METADATA_PROPERTIES"

6.201 DBA_METHOD_PARAMS

DBA_METHOD_PARAMS describes the method parameters of all object types in the database. Its columns are the same as those in ALL_METHOD_PARAMS.

> **See Also:**
>
> "ALL_METHOD_PARAMS"

6.202 DBA_METHOD_RESULTS

DBA_METHOD_RESULTS describes the method results of all object types in the database. Its columns are the same as those in ALL_METHOD_RESULTS.

> **See Also:**
>
> "ALL_METHOD_RESULTS"

6.203 DBA_MINING_MODEL_ATTRIBUTES

DBA_MINING_MODEL_ATTRIBUTES describes all machine learning model attributes in the database. Its columns are the same as those in ALL_MINING_MODEL_ATTRIBUTES.

> **See Also:**
>
> "ALL_MINING_MODEL_ATTRIBUTES"

6.204 DBA_MINING_MODEL_PARTITIONS

DBA_MINING_MODEL_PARTITIONS describes all the model partitions accessible to the system. Its columns are the same as those in ALL_MINING_MODEL_PARTITIONS.

> **See Also:**
>
> "ALL_MINING_MODEL_PARTITIONS"

6.205 DBA_MINING_MODEL_SETTINGS

DBA_MINING_MODEL_SETTINGS describes all machine learning model settings in the database. Its columns are the same as those in ALL_MINING_MODEL_SETTINGS.

> **See Also:**
>
> "ALL_MINING_MODEL_SETTINGS"

6.206 DBA_MINING_MODEL_TABLES

DBA_MINING_MODEL_TABLES describes the tables that contain metadata about the machine learning models in the database.

Machine learning models are schema objects created by Oracle Machine Learning for SQL (OML4SQL).

Model tables reside in the schema of the machine learning model owner. The metadata stored in the tables is controlled by OML4SQL APIs. The tables are read-only. They should not be modified by users.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)	NOT NULL	Owner of the machine learning model
MODEL_NAME	VARCHAR2(128)	NOT NULL	Name of the machine learning model
TABLE_NAME	VARCHAR2(128)	NOT NULL	Name of the table
TABLE_TYPE	VARCHAR2(21)		The type of metadata stored in the table

6.207 DBA_MINING_MODEL_VIEWS

DBA_MINING_MODEL_VIEWS describes all model views in the database. Its columns are the same as those in ALL_MINING_MODEL_VIEWS.

> **See Also:**
>
> "ALL_MINING_MODEL_VIEWS"

6.208 DBA_MINING_MODEL_XFORMS

DBA_MINING_MODEL_XFORMS describes the user-specified transformations embedded in all models accessible in the system. Its columns are the same as those in ALL_MINING_MODEL_XFORMS.

> **See Also:**
>
> "ALL_MINING_MODEL_XFORMS"

6.209 DBA_MINING_MODELS

DBA_MINING_MODELS describes all machine learning models in the database. Its columns are the same as those in ALL_MINING_MODELS.

> **See Also:**
>
> "ALL_MINING_MODELS"

6.210 DBA_MLE_ENV_IMPORTS

DBA_MLE_ENV_IMPORTS describes import name to module mappings in all Oracle Database Multilingual Engine (MLE) environments in the database. Its columns are the same as those in ALL_MLE_ENV_IMPORTS.

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

> **See Also:**
>
> "ALL_MLE_ENV_IMPORTS"

6.211 DBA_MLE_ENVS

DBA_MLE_ENVS describes all Oracle Database Multilingual Engine (MLE) environments in the database. Its columns are the same as those in ALL_MLE_ENVS.

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

> **See Also:**
>
> "ALL_MLE_ENVS"

6.212 DBA_MLE_MODULES

DBA_MLE_MODULES describes all Oracle Database Multilingual Engine (MLE) modules in the database. Its columns are the same as those in ALL_MLE_MODULES.

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

> **See Also:**
>
> "ALL_MLE_MODULES"

6.213 DBA_MLE_PROCEDURES

DBA_MLE_PROCEDURES describes all Oracle Database Multilingual Engine (MLE) functions and procedures in the database. Its columns are the same as those in ALL_MLE_PROCEDURES.

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

> **See Also:**
>
> "ALL_MLE_PROCEDURES"

6.214 DBA_MVIEW_AGGREGATES

`DBA_MVIEW_AGGREGATES` describes the grouping functions (aggregated measures) that appear in the `SELECT` list of all aggregated materialized view in the database. Its columns are the same as those in `ALL_MVIEW_AGGREGATES`.

> **See Also:**
>
> "ALL_MVIEW_AGGREGATES"

6.215 DBA_MVIEW_ANALYSIS

`DBA_MVIEW_ANALYSIS` describes all materialized views in the database that potentially support query rewrite and that provide additional information for analysis by applications. Its columns are the same as those in `ALL_MVIEW_ANALYSIS`.

> **Note:**
>
> This view excludes materialized views that reference remote tables or that include references to non-static values such as `SYSDATE` or `USER`. This view also excludes materialized views that were created as snapshots before Oracle8*i* and that were never altered to enable query rewrite.

> **See Also:**
>
> "ALL_MVIEW_ANALYSIS"

6.216 DBA_MVIEW_COMMENTS

`DBA_MVIEW_COMMENTS` displays comments on all materialized views in the database. Its columns are the same as those in `ALL_MVIEW_COMMENTS`.

> **See Also:**
>
> "ALL_MVIEW_COMMENTS"

6.217 DBA_MVIEW_DETAIL_LOGICAL_PARTITION

DBA_MVIEW_DETAIL_LOGICAL_PARTITION displays freshness information, with respect to logical partition change tracking (LPCT) detail partitions, for all materialized views in the database. Its columns are the same as those in ALL_MVIEW_DETAIL_LOGICAL_PARTITION.

> **See Also:**
>
> "ALL_MVIEW_DETAIL_LOGICAL_PARTITION"

6.218 DBA_MVIEW_DETAIL_PARTITION

DBA_MVIEW_DETAIL_PARTITION displays freshness information, with respect to partition change tracking (PCT) detail partitions, for all materialized views in the database. Its columns are the same as those in ALL_MVIEW_DETAIL_PARTITION.

> **See Also:**
>
> "ALL_MVIEW_DETAIL_PARTITION".

6.219 DBA_MVIEW_DETAIL_RELATIONS

DBA_MVIEW_DETAIL_RELATIONS represents the named detail relations that are either in the FROM list of a materialized view, or that are indirectly referenced through views in the FROM list. Its columns are the same as those in ALL_MVIEW_DETAIL_RELATIONS.

> **See Also:**
>
> "ALL_MVIEW_DETAIL_RELATIONS"

6.220 DBA_MVIEW_DETAIL_SUBPARTITION

DBA_MVIEW_DETAIL_SUBPARTITION displays freshness information, with respect to partition change tracking (PCT) detail subpartitions, for all materialized views in the database. Its columns are the same as those in ALL_MVIEW_DETAIL_SUBPARTITION.

> **See Also:**
>
> "ALL_MVIEW_DETAIL_SUBPARTITION"

6.221 DBA_MVIEW_JOINS

DBA_MVIEW_JOINS describes a join between two columns in the WHERE clause of a subquery that defines a materialized view. Its columns are the same as those in ALL_MVIEW_JOINS.

> **See Also:**
>
> "ALL_MVIEW_JOINS"

6.222 DBA_MVIEW_KEYS

DBA_MVIEW_KEYS describes the columns or expressions in the SELECT list upon which materialized views in the database are based. Its columns are the same as those in ALL_MVIEW_KEYS.

> **See Also:**
>
> "ALL_MVIEW_KEYS"

6.223 DBA_MVIEW_LOG_FILTER_COLS

DBA_MVIEW_LOG_FILTER_COLS displays all columns (excluding primary key columns) being logged in the materialized view logs.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)		Owner of the master table being logged
NAME	VARCHAR2(128)		Name of the master table being logged
COLUMN_NAME	VARCHAR2(128)		Column being logged

6.224 DBA_MVIEW_LOGS

DBA_MVIEW_LOGS describes all materialized view logs in the database. Its columns are the same as those in ALL_MVIEW_LOGS.

> **See Also:**
>
> "ALL_MVIEW_LOGS"

6.225 DBA_MVIEW_REFRESH_TIMES

DBA_MVIEW_REFRESH_TIMES describes refresh times of all materialized views in the database. Its columns are the same as those in ALL_MVIEW_REFRESH_TIMES.

> **See Also:**
>
> "ALL_MVIEW_REFRESH_TIMES"

6.226 DBA_MVIEWS

DBA_MVIEWS describes all materialized views in the database. Its columns are the same as those in ALL_MVIEWS.

> **See Also:**
>
> "ALL_MVIEWS"

6.227 DBA_MVREF_CHANGE_STATS

DBA_MVREF_CHANGE_STATS displays the change data load information on the base tables associated with a refresh run for all the materialized views for the database.

Related View

USER_MVREF_CHANGE_STATS displays the change data load information on the master tables associated with a refresh run for all the materialized views in the database that are accessible to the current user.

Column	Datatype	NULL	Description
TBL_OWNER	VARCHAR2(128)	NOT NULL	Owner of the master table for the materialized view
TBL_NAME	VARCHAR2(128)	NOT NULL	Name of the master table for the materialized view
MV_OWNER	VARCHAR2(128)	NOT NULL	Owner of the materialized view
MV_NAME	VARCHAR2(128)	NOT NULL	Name of the materialized view
REFRESH_ID	NUMBER	NOT NULL	The refresh ID of the refresh run
NUM_ROWS_INS	NUMBER		The number of inserts in the materialized view log of the table (applicable only if the table has a materialized view log)
NUM_ROWS_UPD	NUMBER		The number of updates in the materialized view log of the table (applicable only if the table has a materialized view log)
NUM_ROWS_DEL	NUMBER		The number of deletes in the materialized view log of the table (applicable only if the table has a materialized view log)

Column	Datatype	NULL	Description
NUM_ROWS_DL_INS	NUMBER		The number of direct load inserts on the table
PMOPS_OCCURRED	CHAR(1)		Indicates whether a partition-maintenance operation (PMOP) occurred. Possible values: • Y • N • NULL: Indicates an unknown value
PMOP_DETAILS	VARCHAR2(4000)		Details of the PMOPs in the following format: • TRUNCATE (low_bound, high_bound) • EXECHANGE (low_bound, high_bound)
NUM_ROWS	NUMBER		The number of rows in the table at the start of the refresh operation

> **See Also:**
>
> "USER_MVREF_CHANGE_STATS"

6.228 DBA_MVREF_RUN_STATS

DBA_MVREF_RUN_STATS has information about each refresh run for all the materialized views for the database, with each run being identified by the REFRESH_ID. The information includes timing statistics related to the run and the parameters specified in that run.

Related View

USER_MVREF_RUN_STATS has information about each refresh run for the materialized views accessible for the current database user, with each run being identified by the REFRESH_ID. The information includes timing statistics related to the run and the parameters specified in that run. This view does not display the RUN_OWNER column.

Column	Datatype	NULL	Description
RUN_OWNER	VARCHAR2(128)	NOT NULL	Owner of the refresh operation (the user who launched the operation)
REFRESH_ID	NUMBER	NOT NULL	The refresh ID of the refresh run
NUM_MVS	NUMBER	NOT NULL	The number of materialized views being refreshed in the run
MVIEWS	VARCHAR2(4000)		Shows the list of comma separated parameters specified in the API for the materialized view refresh operation
BASE_TABLES	VARCHAR2(4000)		For internal use only
METHOD	VARCHAR2(4000)		The METHOD parameter specified by the API
ROLLBACK_SEG	VARCHAR2(4000)		The ROLLBACK_SEG parameter specified by the API
PUSH_DEFERRED_RPC	CHAR(1)		The PUSH_DEFERRED_RPC parameter specified by the API
REFRESH_AFTER_ERRORS	CHAR(1)		The REFRESH_AFTER_ERRORS parameter specified by the API

Column	Datatype	NULL	Description
PURGE_OPTION	NUMBER		The PURGE_OPTION parameter specified by the API
PARALLELISM	NUMBER		The PARALLELISM parameter specified by the API
HEAP_SIZE	NUMBER		The HEAP_SIZE parameter specified by the API
ATOMIC_REFRESH	CHAR(1)		The ATOMIC_REFRESH parameter specified by the API
NESTED	CHAR(1)		The NESTED parameter specified by the API
OUT_OF_PLACE	CHAR(1)		The OUT_OF_PLACE parameter specified by the API
NUMBER_OF_FAILURES	NUMBER		The number of failures that occurred in processing the API
START_TIME	TIMESTAMP(6)		Start time of the refresh run
END_TIME	TIMESTAMP(6)		End time of the refresh run
ELAPSED TIME	NUMBER		The length of time for the refresh run, in seconds
LOG_SETUP_TIME	NUMBER		Log setup time (in seconds) for the materialized view for a non-atomic refresh; NULL for an atomic refresh
LOG_PURGE_TIME	NUMBER		Log purge time (in seconds) for the materialized view in the case of atomic refresh; NULL in the case of non-atomic refresh
COMPLETE_STATS_AVAILABLE	CHAR(1)		Indicates whether all the complete refresh statistics are available for this run: • Y: All the statistics are available • N: All the statistics are not available

See Also:

"USER_MVREF_RUN_STATS"

6.229 DBA_MVREF_STATS

DBA_MVREF_STATS shows the REFRESH_ID associated with each refresh run of each materialized view for the database. It also provides some basic timing statistics related to that materialized view's refresh in that run.

Related View

USER_MVREF_STATS shows the REFRESH_ID associated with each refresh run of each materialized view for the database that is accessible to the current user. It also provides some basic timing statistics related to that materialized view's refresh in that run. This view does not display the MV_OWNER column.

Column	Datatype	NULL	Description
MV_OWNER	VARCHAR2(128)	NOT NULL	Owner of the materialized view
MV_NAME	VARCHAR2(128)	NOT NULL	Name of the materialized view
REFRESH_ID	NUMBER	NOT NULL	The refresh ID of the refresh run

Column	Datatype	NULL	Description
REFRESH_METHOD	VARCHAR2(30)		The refresh method used to refresh the materialized view: • FAST • PCT • COMPLETE • OUT OF PLACE FAST • OUT OF PLACE PCT • OUT OF PLACE COMPLETE
REFRESH_OPTIMIZATIONS	VARCHAR2(4000)		The refresh optimization, for example, a null refresh, or a primary key/foreign key that is applied during refresh of the materialize view
ADDITIONAL_EXECUTIONS	VARCHAR2(4000)		The additional executions, for example, an index rebuild, or log operations involved during refresh of the materialized view
START_TIME	TIMESTAMP(6)		Start time of the refresh run
END_TIME	TIMESTAMP(6)		End time of the refresh run
ELAPSED_TIME	NUMBER		The length of time for the refresh run, in seconds
LOG_SETUP_TIME	NUMBER		Log setup time (in seconds) for the materialized view for a non-atomic refresh; NULL for an atomic refresh
LOG_PURGE_TIME	NUMBER		Log purge time (in seconds) for the materialized view in the case of atomic refresh; NULL in the case of non-atomic refresh
INITIAL_NUM_ROWS	NUMBER		Initial number of rows in the materialized view (at the start of the refresh)
FINAL_NUM_ROWS	NUMBER		Final number of rows in the materialized view (at the end of the refresh)

See Also:

"USER_MVREF_STATS"

6.230 DBA_MVREF_STATS_PARAMS

DBA_MVREF_STATS_PARAMS displays the refresh statistics properties associated with each materialized view. These properties can be modified with the DBMS_MVIEW_STATS.SET_MVREF_STATS_PARAMS procedure.

Related View

USER_MVREF_STATS_PARAMS displays the refresh statistics properties associated with each materialized view accessible to the current user. These properties can be modified with the DBMS_MVIEW_STATS.SET_MVREF_STATS_PARAMS procedure.

Column	Datatype	NULL	Description
MV_OWNER	VARCHAR2(128)	NOT NULL	Owner of the materialized view
MV_NAME	VARCHAR2(128)	NOT NULL	Name of the materialized view

Column	Datatype	NULL	Description
COLLECTION_LEVEL	VARCHAR2(8)		The collection level for the materialized view
RETENTION_PERIOD	NUMBER		The retention period for the materialize view

> **See Also:**
>
> "USER_MVREF_STATS_PARAMS"

6.231 DBA_MVREF_STATS_SYS_DEFAULTS

DBA_MVREF_STATS_SYS_DEFAULTS displays the system-wide defaults for the refresh history statistics properties for materialized views. These values can be altered with the SET_SYSTEM_DEFAULTS procedure by a database administrator.

This view contains exactly two rows corresponding to the collection-level and retention-period properties; their initial values are TYPICAL and 31 respectively.

Related View

USER_MVREF_STATS_SYS_DEFAULTS displays the system-wide defaults for the refresh history statistics properties for materialized views accessible to the current user. These values can be altered with the SET_SYSTEM_DEFAULTS procedure by a database administrator.

Column	Datatype	NULL	Description
PARAMETER_NAME	CHAR(16)		Value of the parameter_name parameter: • COLLECTION_LEVEL • RETENTION_PERIOD
VALUE	VARCHAR2(40)		The system-wide default value for the parameter

> **See Also:**
>
> "USER_MVREF_STATS_SYS_DEFAULTS"

6.232 DBA_MVREF_STMT_STATS

DBA_MVREF_STMT_STATS shows information associated with each refresh statement of a materialized view in a refresh run.

Related View

USER_MVREF_STMT_STATS shows information associated with each refresh statement of a materialized view accessible to the current user in a refresh run.

Column	Datatype	NULL	Description
MV_OWNER	VARCHAR2(128)	NOT NULL	Owner of the materialized view
MV_NAME	VARCHAR2(128)	NOT NULL	Name of the materialized view
REFRESH_ID	NUMBER	NOT NULL	The refresh ID of the refresh run
STEP	NUMBER	NOT NULL	A number indicating the step in the refresh process in which the statement is executed for the materialized view. Steps are numbered consecutively starting at 1.
SQLID	VARCHAR2(14)	NOT NULL	The SQL ID of the statement
STMT	CLOB	NOT NULL	The text of the SQL statement
EXECUTION_TIME	NUMBER	NOT NULL	The time it took to execute the statement (in seconds)
EXECUTION_PLAN	XMLTYPE STORAGE NOT TRANSPORTABLE BINARY		For internal use only

See Also:

"USER_MVREF_STMT_STATS"

6.233 DBA_NESTED_TABLE_COLS

DBA_NESTED_TABLE_COLS describes the columns of all nested tables in the database. Its columns are the same as those in ALL_NESTED_TABLE_COLS.

To gather statistics for this view, use the DBMS_STATS package.

See Also:

"ALL_NESTED_TABLE_COLS"

6.234 DBA_NESTED_TABLES

DBA_NESTED_TABLES describes all nested tables contained in all tables in the database. Its columns are the same as those in ALL_NESTED_TABLES.

See Also:

"ALL_NESTED_TABLES"

6.235 DBA_NETWORK_ACL_PRIVILEGES

DBA_NETWORK_ACL_PRIVILEGES describes the network privileges defined in all access control lists that are currently assigned to network hosts.

> **Note:**
>
> This DBA_NETWORK_ACL_PRIVILEGES view is deprecated in Oracle Database 12*c* Release 1 (12.1). Oracle recommends that you use the DBA_HOST_ACES view, instead.

Column	Datatype	NULL	Description
ACL	VARCHAR2(4000)		Path of the access control list
ACLID	RAW(8)	NOT NULL	Object ID of the access control list
PRINCIPAL	VARCHAR2(128)		Principal (database user or role) whom the privilege is granted to or denied from
PRIVILEGE	VARCHAR2(128)		Network privilege
IS_GRANT	VARCHAR2(5)		Indicates whether the privilege is granted (true) or denied (false)
INVERT	VARCHAR2(5)		Indicates whether the access control entry contains invert principal (true) or not (false)
START_DATE	TIMESTAMP(6)		Start date of the access control entry
END_DATE	TIMESTAMP(6)		End date of the access control entry
ACL_OWNER	VARCHAR2(128)		Owner of the access control list

> **See Also:**
>
> "DBA_HOST_ACES"

6.236 DBA_NETWORK_ACLS

DBA_NETWORK_ACLS describes the access control list assignments to network hosts.

> **Note:**
>
> This DBA_NETWORK_ACLS view is deprecated in Oracle Database 12*c* Release 1 (12.1). Oracle recommends that you use the DBA_HOST_ACLS view, instead.

Column	Datatype	NULL	Description
HOST	VARCHAR2(1000)	NOT NULL	Network host
LOWER_PORT	NUMBER(5)		Lower bound of the port range

Column	Datatype	NULL	Description
UPPER_PORT	NUMBER(5)		Upper bound of the port range
ACL	VARCHAR2(4000)		Path of the access control list
ACLID	RAW(8)		Object ID of the access control list
ACL_OWNER	VARCHAR2(128)		Owner of the access control list

> **See Also:**
>
> "DBA_HOST_ACLS"

6.237 DBA_OBJ_AUDIT_OPTS

DBA_OBJ_AUDIT_OPTS describes auditing options on all objects.

> **Note:**
>
> This view is deprecated and applies only to traditional auditing. Traditional auditing is desupported starting in Oracle Database 23ai. Though traditional auditing is desupported, any current traditional audit settings that you have will still be honored and are viewable with this view. See *Oracle Database Security Guide* for more information about how this desupport works.

Related View

USER_OBJ_AUDIT_OPTS describes auditing options on all objects owned by the current user. This view does not display the OWNER column.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)		Owner of the object
OBJECT_NAME	VARCHAR2(128)		Name of the object
OBJECT_TYPE	VARCHAR2(23)		Type of the object
ALT	VARCHAR2(3)		Auditing ALTER WHENEVER SUCCESSFUL / UNSUCCESSFUL
AUD	VARCHAR2(3)		Auditing AUDIT WHENEVER SUCCESSFUL / UNSUCCESSFUL
COM	VARCHAR2(3)		Auditing COMMENT WHENEVER SUCCESSFUL / UNSUCCESSFUL
DEL	VARCHAR2(3)		Auditing DELETE WHENEVER SUCCESSFUL / UNSUCCESSFUL
GRA	VARCHAR2(3)		Auditing GRANT WHENEVER SUCCESSFUL / UNSUCCESSFUL
IND	VARCHAR2(3)		Auditing INDEX WHENEVER SUCCESSFUL / UNSUCCESSFUL
INS	VARCHAR2(3)		Auditing INSERT WHENEVER SUCCESSFUL / UNSUCCESSFUL
LOC	VARCHAR2(3)		Auditing LOCK WHENEVER SUCCESSFUL / UNSUCCESSFUL

Column	Datatype	NULL	Description
REN	VARCHAR2(3)		Auditing RENAME WHENEVER SUCCESSFUL / UNSUCCESSFUL
SEL	VARCHAR2(3)		Auditing SELECT WHENEVER SUCCESSFUL / UNSUCCESSFUL
UPD	VARCHAR2(3)		Auditing UPDATE WHENEVER SUCCESSFUL / UNSUCCESSFUL
EXE	VARCHAR2(3)		Auditing EXECUTE WHENEVER SUCCESSFUL / UNSUCCESSFUL
CRE	VARCHAR2(3)		Auditing CREATE WHENEVER SUCCESSFUL / UNSUCCESSFUL
REA	VARCHAR2(3)		Auditing READ WHENEVER SUCCESSFUL / UNSUCCESSFUL
WRI	VARCHAR2(3)		Auditing WRITE WHENEVER SUCCESSFUL / UNSUCCESSFUL
FBK	VARCHAR2(3)		Auditing FLASHBACK WHENEVER SUCCESSFUL / UNSUCCESSFUL

See Also:

- "USER_OBJ_AUDIT_OPTS"
- *Oracle Database SQL Language Reference* for more information about the SQL AUDIT statement for unified auditing
- *Oracle Database SQL Language Reference* for more information about the SQL AUDIT statement for traditional auditing
- *Oracle Database Security Guide* to learn how to find information about audited activities

6.238 DBA_OBJ_COLATTRS

DBA_OBJ_COLATTRS describes object columns and attributes contained in all tables in the database. Its columns are the same as those in ALL_OBJ_COLATTRS.

See Also:

"ALL_OBJ_COLATTRS"

6.239 DBA_OBJECT_SIZE

DBA_OBJECT_SIZE lists the sizes, in bytes, of various PL/SQL objects.

Related View

- USER_OBJECT_SIZE lists the size of PL/SQL objects owned by the current user.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)	NOT NULL	Owner of the object
NAME	VARCHAR2(128)	NOT NULL	Name of the object
TYPE	VARCHAR2(18)		Type of the object: TYPE, TYPE BODY, TABLE, VIEW, SYNONYM, SEQUENCE, PROCEDURE, FUNCTION, PACKAGE, PACKAGE BODY, JAVA SOURCE, JAVA CLASS or JAVA RESOURCE
SOURCE_SIZE	NUMBER		Size of the source in bytes. Must be in memory during compilation, or dynamic recompilation.
PARSED_SIZE	NUMBER		Size of the parsed form of the object, in bytes. Must be in memory when an object is being compiled that references this object.
CODE_SIZE	NUMBER		Code size, in bytes. Must be in memory when this object is executing.
ERROR_SIZE	NUMBER		Size of error messages, in bytes. In memory during the compilation of the object when there are compilation errors.

See Also:

"USER_OBJECT_SIZE"

6.240 DBA_OBJECT_TABLES

DBA_OBJECT_TABLES describes all object tables in the database. Its columns are the same as those in ALL_OBJECT_TABLES.

See Also:

"ALL_OBJECT_TABLES"

6.241 DBA_OBJECT_USAGE

DBA_OBJECT_USAGE displays statistics about index usage gathered from the database for all the indexes in the database.

You can use this view to monitor index usage. All indexes that have been used at least once can be monitored and displayed in this view.

Related View

- USER_OBJECT_USAGE displays statistics about index usage gathered from the database for the indexes owned by the current user.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)	NOT NULL	Index owner
INDEX_NAME	VARCHAR2(128)	NOT NULL	Index name in sys.obj$.name
TABLE_NAME	VARCHAR2(128)	NOT NULL	Table name in sys.obj$.name
MONITORING	VARCHAR2(3)		Indicates whether the monitoring feature is turned on. Possible values: • YES • NO
USED	VARCHAR2(3)		Indicates whether the index has been accessed. Possible values: • YES • NO
START_MONITORING	VARCHAR2(19)		Start monitoring time in sys.object_stats.start_monitoring
END_MONITORING	VARCHAR2(19)		End monitoring time in sys.object_stats.end_monitoring

See Also:

"USER_OBJECT_USAGE"

6.242 DBA_OBJECTS

DBA_OBJECTS describes all objects in the database. Its columns are the same as those in ALL_OBJECTS.

See Also:

"ALL_OBJECTS"

6.243 DBA_OBJECTS_AE

DBA_OBJECTS_AE describes all objects (across all editions) in the database. Its columns are the same as those in ALL_OBJECTS_AE.

See Also:

"ALL_OBJECTS_AE"

6.244 DBA_OGG_AUTO_CAPTURED_TABLES

DBA_OGG_AUTO_CAPTURED_TABLES describes all tables in the database that are enabled for Oracle GoldenGate automatic capture.

Related View

USER_OGG_AUTO_CAPTURED_TABLES describes the tables owned by the current user that are enabled for Oracle GoldenGate automatic capture. This view does not display the OWNER column.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)	NOT NULL	Owner of the table
NAME	VARCHAR2(128)	NOT NULL	Name of the table
ALLOW_NOVALIDATE_PK	VARCHAR2(3)		Indicates whether a primary key constraint in NOVALIDATE mode can be used as a unique identifier for the table (YES) or not (NO)

> **See Also:**
>
> "USER_OGG_AUTO_CAPTURED_TABLES"

6.245 DBA_OPANCILLARY

DBA_OPANCILLARY provides ancillary information for all operator bindings in the database. Its columns are the same as those in ALL_OPANCILLARY.

> **See Also:**
>
> "ALL_OPANCILLARY"

6.246 DBA_OPARGUMENTS

DBA_OPARGUMENTS provides argument information for all operator bindings in the database. Its columns are the same as those in ALL_OPARGUMENTS.

> **See Also:**
>
> "ALL_OPARGUMENTS"

6.247 DBA_OPBINDINGS

DBA_OPBINDINGS describes the binding functions and methods on all operators in the database. Its columns are the same as those in ALL_OPBINDINGS.

> **See Also:**
>
> "ALL_OPBINDINGS"

6.248 DBA_OPERATOR_COMMENTS

DBA_OPERATOR_COMMENTS displays comments for all user-defined operators in the database. Its columns are the same as those in ALL_OPERATOR_COMMENTS.

> **See Also:**
>
> "ALL_OPERATOR_COMMENTS"

6.249 DBA_OPERATORS

DBA_OPERATORS describes all operators in the database. Its columns are the same as those in ALL_OPERATORS.

> **See Also:**
>
> "ALL_OPERATORS"

6.250 DBA_OPTSTAT_OPERATION_TASKS

DBA_OPTSTAT_OPERATION_TASKS displays the history of tasks that are performed as part of statistics operations (recorded in DBA_OPTSTAT_OPERATIONS). Each task represents a target object to be processed in the corresponding parent operation.

Column	Datatype	NULL	Description
OPID	NUMBER		Internal identifier for the statistics operation that the task belongs to
TARGET	VARCHAR2(100)		Name of the object that this task operates on
TARGET_OBJN	NUMBER		Object number of the target object

Column	Datatype	NULL	Description
TARGET_TYPE	VARCHAR2(40)		Type of the target object. Possible values are: • TABLE • TABLE (GLOBAL STATS ONLY): Task is created to gather only global statistics of a partitioned table • TABLE (COORDINATOR JOB): Coordinator task for a partitioned table when concurrency is on • TABLE PARTITION • TABLE SUBPARTITION • INDEX • INDEX PARTITION • INDEX SUBPARTITION
TARGET_SIZE	NUMBER		Target size (in number of blocks) when the task started
START_TIME	TIMESTAMP(6) WITH TIME ZONE		Task start time
END_TIME	TIMESTAMP(6) WITH TIME ZONE		Task end time
STATUS	VARCHAR2(49)		Current task status. Possible values are: • PENDING: Task is queued for processing • IN PROGRESS: Task is currently running • COMPLETED: Task has completed successfully • FAILED: Task has failed • SKIPPED: Task has been skipped, as it does not exist any more, or its stats are not stale (applies only to only automatic statistics gathering) • TIMED OUT: Maintenance window was not enough to complete this task (applies only to automatic statistics gathering)
JOB_NAME	VARCHAR2(50)		Name of the scheduler job that executes this task (for example, when concurrency is on)
ESTIMATED_COST	NUMBER		Estimated cost of the task (measured as elapsed time in seconds). This column is populated only when concurrency is on.
BATCHING_COEFF	NUMBER		For internal use only
ACTIONS	NUMBER		For internal use only
PRIORITY	NUMBER		Rank of the task among all target objects for the parent operation
FLAGS	NUMBER		For internal use only
NOTES	VARCHAR2(4000)		Notes about the underlying task, such as the failure message for tasks with status FAILED.

See Also:

"DBA_OPTSTAT_OPERATIONS"

6.251 DBA_OPTSTAT_OPERATIONS

DBA_OPTSTAT_OPERATIONS contains a history of statistics operations performed at the schema and database level using the DBMS_STATS package.

Column	Datatype	NULL	Description
ID	NUMBER		Internal ID of the statistics operation
OPERATION	VARCHAR2(64)		Operation name
TARGET	VARCHAR2(64)		Target on which the operation was performed
START_TIME	TIMESTAMP(6) WITH TIME ZONE		Time at which the operation started
END_TIME	TIMESTAMP(6) WITH TIME ZONE		Time at which the operation ended
STATUS	VARCHAR2(49)		Current operation status. Possible values are: • IN PROGRESS: Operation is currently running • COMPLETED: Operation has completed successfully • FAILED: Operation has failed • TIMED OUT: Maintenance window was not enough to complete this operation (applies only to automatic statistics gathering)
JOB_NAME	VARCHAR2(32)		Name of the scheduler job that executes this operation (for example, a user scheduled statistics gathering job)
SESSION_ID	NUMBER		ID of the session in which this operation is invoked
NOTES	VARCHAR2(4000)		Notes about the operation, such as a failure message for operations with status FAILED

See Also:

- "DBA_OPTSTAT_OPERATION_TASKS"
- *Oracle Database PL/SQL Packages and Types Reference* for more information about the DBMS_STATS package
- *Oracle Database PL/SQL Packages and Types Reference* for more information about the DBMS_REPAIR.ADMIN_TABLES procedure
- *Oracle Database PL/SQL Packages and Types Reference* for more information about the DBMS_REPAIR.DUMP_ORPHAN_KEYS procedure

6.252 DBA_ORPHAN_KEY_TABLE

DBA_ORPHAN_KEY_TABLE reports key values from indexes where the underlying base table has block corruptions.

To create the view, run the DBMS_REPAIR.ADMIN_TABLES procedure. To populate the orphan key table for an index, run the DBMS_REPAIR.DUMP_ORPHAN_KEYS procedure on the index. For each key in the index that points to a corrupt data block, Oracle inserts a row into the orphan key table.

Column	Datatype	NULL	Description
SCHEMA_NAME	VARCHAR2(128)	NOT NULL	Schema name of the index
INDEX_NAME	VARCHAR2(128)	NOT NULL	Name of the index
IPART_NAME	VARCHAR2(128)	NULL	Name of the index partition or subpartition
INDEX_ID	NUMBER	NOT NULL	Dictionary object ID of the index
TABLE_NAME	VARCHAR2(128)	NOT NULL	Name of the base table of the index
PART_NAME	VARCHAR2(128)	NULL	Name of the base table partition or subpartition
TABLE_ID	NUMBER	NOT NULL	Dictionary object ID of the base table
KEYROWID	ROWID	NOT NULL	Physical rowid of the corrupt data row
KEY	ROWID	NOT NULL	Key values for the index entry
DUMP_TIMESTAMP	DATE	NOT NULL	Timestamp when the entry was made into the orphan key table

6.253 DBA_OUTLINE_HINTS

DBA_OUTLINE_HINTS describes the set of hints stored in all outlines in the database.

Related View

USER_OUTLINE_HINTS describes the set of hints stored in the outlines owned by the current user. This view does not display the OWNER column.

Column	Datatype	NULL	Description
NAME	VARCHAR2(128)		Name of the outline
OWNER	VARCHAR2(128)		Name of the user who created the outline
NODE	NUMBER		ID of the query or subquery to which the hint applies. The top-level query is labeled 1. Subqueries are assigned sequentially numbered labels, starting with 2.
STAGE	NUMBER		Outline hints can be applied at three different stages during the compilation process. This column indicates the stage at which this hint was applied.
JOIN_POS	NUMBER		Position of the table in the join order. The value is 0 for all hints except access method hints, which identify a table to which the hint and the join position apply.
HINT	CLOB		Text of the hint

See Also:

"USER_OUTLINE_HINTS"

6.254 DBA_OUTLINES

DBA_OUTLINES describes all stored outlines in the database.

Related View

USER_OUTLINES describes the stored outlines owned by the current user. This view does not display the OWNER column.

Column	Datatype	NULL	Description
NAME	VARCHAR2(128)		User-specified or generated name of the stored outline. The name must be of a form that can be expressed in SQL.
OWNER	VARCHAR2(128)		Name of the user who created the outline
CATEGORY	VARCHAR2(128)		User-defined name of the category to which the outline belongs
USED	VARCHAR2(6)		Indicates whether the outline has ever been used (USED) or not (UNUSED)
TIMESTAMP	DATE		Timestamp of outline creation
VERSION	VARCHAR2(64)		Oracle version that created the outline
SQL_TEXT	LONG		SQL text of the query, including any hints that were a part of the original statement. If bind variables are included, the variable names are stored as SQL text, not the values that are assigned to the variables. **Note:** This field may contain sensitive information about your database or application. Therefore, use discretion when granting SELECT or VIEW object privileges on these views.
SIGNATURE	RAW(16)		Signature uniquely identifying the outline SQL text
COMPATIBLE	VARCHAR2(12)		Indicates whether the outline hints were compatible across a migration (COMPATIBLE) or not (INCOMPATIBLE)
ENABLED	VARCHAR2(8)		Indicates whether the outline is enabled (ENABLED) or disabled (DISABLED)
FORMAT	VARCHAR2(6)		Hint format: • NORMAL • LOCAL
MIGRATED	VARCHAR2(12)		Indicates whether the outline has been migrated to a SQL plan baseline (MIGRATED) or not (NOT-MIGRATED)

See Also:

"USER_OUTLINES"

6.255 DBA_OUTSTANDING_ALERTS

DBA_OUTSTANDING_ALERTS describes alerts which the server considers to be outstanding.

Column	Datatype	NULL	Description
SEQUENCE_ID	NUMBER		Alert sequence number
REASON_ID	NUMBER	NOT NULL	ID of the alert reason
OWNER	VARCHAR2(128)		Owner of the object on which the alert was issued
OBJECT_NAME	VARCHAR2(513)		Name of the object
SUBOBJECT_NAME	VARCHAR2(128)		Name of the subobject (for example: partition)
OBJECT_TYPE	VARCHAR2(64)		Object type (for example: table, tablespace)
REASON	VARCHAR2(4000)		Reason for the alert
TIME_SUGGESTED	TIMESTAMP(6) WITH TIME ZONE		Time when the alert was last updated
CREATION_TIME	TIMESTAMP(6) WITH TIME ZONE		Time when the alert was first created
SUGGESTED_ACTION	VARCHAR2(4000)		Advice of the recommended action
ADVISOR_NAME	VARCHAR2(128)		Name of the advisor to be invoked for more information
METRIC_VALUE	NUMBER		Value of the related metrics
MESSAGE_TYPE	VARCHAR2(12)		Message type: • Notification • Warning
MESSAGE_GROUP	VARCHAR2(64)		Name of the message group to which the alert belongs
MESSAGE_LEVEL	NUMBER		Message severity level (1 to 32)
HOSTING_CLIENT_ID	VARCHAR2(64)		ID of the client or security group to which the alert relates
MODULE_ID	VARCHAR2(64)		ID of the module that originated the alert
PROCESS_ID	VARCHAR2(128)		Process ID
HOST_ID	VARCHAR2(256)		DNS host name of the originating host
HOST_NW_ADDR	VARCHAR2(256)		IP or other network address of the originating host
INSTANCE_NAME	VARCHAR2(16)		Originating instance name
INSTANCE_NUMBER	NUMBER		Originating instance number
USER_ID	VARCHAR2(128)		User ID
EXECUTION_CONTEXT_ID	VARCHAR2(128)		Execution Context ID
ERROR_INSTANCE_ID	VARCHAR2(142)		ID of an error instance plus a sequence number
STATE_TRANSITION_NUMBER	NUMBER		Sequence number of the state transition for the alert
PDB_NAME	VARCHAR2(128)		PDB 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

6.256 DBA_PARALLEL_EXECUTE_CHUNKS

DBA_PARALLEL_EXECUTE_CHUNKS displays the chunks for all tasks in the database.

Related View

USER_PARALLEL_EXECUTE_CHUNKS displays the chunks for tasks created by the current user. This view does not display the TASK_OWNER column.

Column	Datatype	NULL	Description
CHUNK_ID	NUMBER	NOT NULL	Unique ID for the chunk
TASK_OWNER	VARCHAR2(128)	NOT NULL	Owner of the task
TASK_NAME	VARCHAR2(128)	NOT NULL	Name of the task
STATUS	VARCHAR2(20)		Status of the chunk: • UNASSIGNED • ASSIGNED • PROCESSED • PROCESSED_WITH_ERROR
START_ROWID	ROWID		Rowid for the first row in the chunk
END_ROWID	ROWID		Rowid for the last row in the chunk
START_ID	NUMBER		Number column value of the first row in the chunk
END_ID	NUMBER		Number column value of the last row in the chunk
JOB_NAME	VARCHAR2(128)		Name of the job which processed this chunk
START_TS	TIMESTAMP(6)		Processing start time for the chunk
END_TS	TIMESTAMP(6)		Processing end time for the chunk
ERROR_CODE	NUMBER		Error code returned during the execution of the chunk if the STATUS column is PROCESSED_WITH_ERROR
ERROR_MESSAGE	VARCHAR2(4000)		Error message returned during the execution of the chunk if the STATUS column is PROCESSED_WITH_ERROR

See Also:

"USER_PARALLEL_EXECUTE_CHUNKS"

6.257 DBA_PARALLEL_EXECUTE_TASKS

DBA_PARALLEL_EXECUTE_TASKS displays all tasks in the database.

Related View

USER_PARALLEL_EXECUTE_TASKS displays the tasks created by the current user. This view does not display the TASK_OWNER column.

Column	Datatype	NULL	Description
TASK_OWNER	VARCHAR2(128)	NOT NULL	Owner of the task

Column	Datatype	NULL	Description
TASK_NAME	VARCHAR2(128)	NOT NULL	Name of the task
CHUNK_TYPE	VARCHAR2(12)		Type of parallel update: • UNDELARED • ROWID_RANGE • NUMBER_RANGE
STATUS	VARCHAR2(19)		Status of the task: • CREATED • CHUNKING • CHUNKING_FAILED • CHUNKED • PROCESSING • FINISHED • FINISHED_WITH_ERROR • CRASHED
TABLE_OWNER	VARCHAR2(128)		Owner of the table to be chunked
TABLE_NAME	VARCHAR2(128)		Name of the table to be chunked
NUMBER_COLUMN	VARCHAR2(128)		Name of the column holding IDs (only applicable to NUMBER_RANGE chunking type)
TASK_COMMENT	VARCHAR2(4000)		Comment field
JOB_PREFIX	VARCHAR2(128)		Prefix of the job name executing this task
SQL_STMT	CLOB		Argument used in the previous DBMS_PARALLEL_EXECUTE.RUN_TASK
LANGUAGE_FLAG	NUMBER		Argument used in the previous DBMS_PARALLEL_EXECUTE.RUN_TASK
EDITION	VARCHAR2(130)		Argument used in the previous DBMS_PARALLEL_EXECUTE.RUN_TASK
APPLY_CROSSEDITION_TRIGGER	VARCHAR2(130)		Argument used in the previous DBMS_PARALLEL_EXECUTE.RUN_TASK
FIRE_APPLY_TRIGGER	VARCHAR2(10)		Argument used in the previous DBMS_PARALLEL_EXECUTE.RUN_TASK
PARALLEL_LEVEL	NUMBER		Argument used in the previous DBMS_PARALLEL_EXECUTE.RUN_TASK
JOB_CLASS	VARCHAR2(128)		Argument used in the previous DBMS_PARALLEL_EXECUTE.RUN_TASK

See Also:

- "USER_PARALLEL_EXECUTE_TASKS"
- *Oracle Database PL/SQL Packages and Types Reference* for more information about the DBMS_PARALLEL_EXECUTE.RUN_TASK procedure

6.258 DBA_PART_COL_STATISTICS

DBA_PART_COL_STATISTICS displays column statistics and histogram information for all table partitions in the database. Its columns are the same as those in ALL_PART_COL_STATISTICS.

> **See Also:**
>
> "ALL_PART_COL_STATISTICS"

6.259 DBA_PART_HISTOGRAMS

DBA_PART_HISTOGRAMS displays the histogram data (endpoints per histogram) for the histograms on all table partitions in the database. Its columns are the same as those in ALL_PART_HISTOGRAMS.

> **See Also:**
>
> "ALL_PART_HISTOGRAMS"

6.260 DBA_PART_INDEXES

DBA_PART_INDEXES displays the object-level partitioning information for all partitioned indexes in the database. Its columns are the same as those in ALL_PART_INDEXES.

> **See Also:**
>
> "ALL_PART_INDEXES"

6.261 DBA_PART_KEY_COLUMNS

DBA_PART_KEY_COLUMNS describes the partitioning key columns for all partitioned objects in the database. Its columns are the same as those in ALL_PART_KEY_COLUMNS.

> **See Also:**
>
> "ALL_PART_KEY_COLUMNS"

6.262 DBA_PART_LOBS

DBA_PART_LOBS displays table-level information for all partitioned LOBs in the database, including default attributes for LOB data partitions. Its columns are the same as those in ALL_PART_LOBS.

> **See Also:**
>
> "ALL_PART_LOBS"

6.263 DBA_PART_TABLES

DBA_PART_TABLES displays the object-level partitioning information for all partitioned tables in the database. Its columns are the same as those in ALL_PART_TABLES.

> **See Also:**
>
> "ALL_PART_TABLES"

6.264 DBA_PARTIAL_DROP_TABS

DBA_PARTIAL_DROP_TABS describes all tables in the database that have partially completed DROP COLUMN operations. Its columns are the same as those in ALL_PARTIAL_DROP_TABS.

> **See Also:**
>
> "ALL_PARTIAL_DROP_TABS"

6.265 DBA_PDB_HISTORY

DBA_PDB_HISTORY describes the lineage of the PDB to which it belongs.

Column	Datatype	NULL	Description
PDB_NAME	VARCHAR2(128)	NOT NULL	Name of this PDB in one of its incarnations
PDB_ID	NUMBER	NOT NULL	Container ID of this PDB in one of its incarnations.
PDB_DBID	NUMBER	NOT NULL	Database ID of this PDB in one of its incarnations
PDB_GUID	RAW(16)	NOT NULL	Globally unique ID of this PDB in one of its incarnations
OP_SCNBAS	NUMBER	NOT NULL	SCN base when an operation was performed on one of the incarnations of this PDB

Column	Datatype	NULL	Description
OP_SCNWRP	NUMBER	NOT NULL	SCN wrap when an operation was performed on one of incarnations of this PDB
OP_TIMESTAMP	DATE	NOT NULL	Timestamp of an operation performed on one of the incarnations of this PDB
OPERATION	VARCHAR2(16)	NOT NULL	Operation that was performed on one of the incarnations of this PDB
DB_VERSION	NUMBER	NOT NULL	Database version
CLONED_FROM_PDB_NAME	VARCHAR2(128)		Name of a PDB from which one of the incarnations of this PDB was cloned
CLONED_FROM_PDB_DBID	NUMBER		Database ID of a PDB from which one of the incarnations of this PDB was cloned
CLONED_FROM_PDB_GUID	RAW(16)		Globally unique ID of a PDB from which one of the incarnations of this PDB was cloned
DB_NAME	VARCHAR2(128)		Name of a CDB in which one of the incarnations of this PDB was created
DB_UNIQUE_NAME	VARCHAR2(128)		Unique name of a CDB in which one of the incarnations of this PDB was created
DB_DBID	NUMBER		Database ID of a CDB in which one of the incarnations of this PDB was created
CLONETAG	VARCHAR2(128)		Clone tag name for the PDB if the PDB was cloned using the snapshot copy mechanism
DB_VERSION_STRING	VARCHAR2(204)		Database version string

6.266 DBA_PDB_SAVED_STATES

DBA_PDB_SAVED_STATES shows information about the current saved PDB states in the CDB.

This view is a data link, so the data is also available within the PDB.

Column	Datatype	NULL	Description
CON_ID	NUMBER	NOT NULL	The ID of the PDB
CON_NAME	VARCHAR2(128)	NOT NULL	Name of the PDB
INSTANCE_NAME	VARCHAR2(128)	NOT NULL	Name of the instance for which the state is saved
CON_UID	NUMBER	NOT NULL	Unique ID assigned to the PDB at creation time
GUID	RAW(16)		Globally unique immutable ID assigned to the PDB at creation time
STATE	VARCHAR2(14)		Open state of the PDB
RESTRICTED	VARCHAR2(3)		Restricted mode of the PDB

See Also:

Oracle Database SQL Language Reference for more information about preserving a PDB's open mode across an instance restart

6.267 DBA_PDB_SNAPSHOTFILE

DBA_PDB_SNAPSHOTFILE displays the files associated with snapshots taken of pluggable databases (PDBs).

You can use this view in conjunction with the DBA_PDB_SNAPSHOT view. Join the SNAPSHOT_SCN column in this view with the SNAPSHOT_SCN column in DBA_PDB_SNAPSHOT to determine the files associated with a particular PDB snapshot. A PDB snapshot consists of an archive log file, one or more data files, and one or more XML files. A row is added to this view for each file associated with a PDB snapshot.

Column	Datatype	NULL	Description
CON_ID	NUMBER	NOT NULL	The ID of the PDB
SNAPSHOT_SCN	NUMBER	NOT NULL	SCN at which the snapshot was taken
SNAPSHOT_FILENAME	VARCHAR2(513)	NOT NULL	Snapshot file name
SNAPSHOT_FILETYPE	VARCHAR2(8)	NOT NULL	Snapshot file type. Possible values: • ARCH: Archive log file • DATA: Data file • XML: XML file

See Also:

"DBA_PDB_SNAPSHOTS"

6.268 DBA_PDB_SNAPSHOTS

DBA_PDB_SNAPSHOTS describes the snapshots taken of pluggable databases (PDBs).

Rows are added to this view when a snapshot of a PDB is taken by using the ALTER PLUGGABLE DATABASE SNAPSHOT SQL statement.

Column	Datatype	NULL	Description
CON_ID	NUMBER	NOT NULL	The ID of the PDB
CON_UID	NUMBER	NOT NULL	Unique ID assigned to the PDB at creation time
CON_NAME	VARCHAR2(128)	NOT NULL	Name of the PDB
SNAPSHOT_NAME	VARCHAR2(128)	NOT NULL	Snapshot name of the PDB
SNAPSHOT_SCN	NUMBER	NOT NULL	SCN at which the snapshot was taken
PREVIOUS_SNAPSHOT_SCN	NUMBER	NOT NULL	SCN at which the previous snapshot for the PDB was taken
SNAPSHOT_TIME	NUMBER	NOT NULL	Timestamp at which the snapshot was taken
PREVIOUS_SNAPSHOT_TIME	NUMBER	NOT NULL	Timestamp of the previous snapshot for this PDB
FULL_SNAPSHOT_PATH	VARCHAR2(4000)	NOT NULL	Full path for the snapshot

> **Note:**
>
> This view does not display snapshot copy PDBs, which are created by using the `CREATE PLUGGABLE DATABASE ... SNAPSHOT COPY` SQL statement.

> **See Also:**
>
> "DBA_PDB_SNAPSHOTFILE" for information about the files associated with a particular PDB snapshot

6.269 DBA_PDBS

`DBA_PDBS` describes PDBs belonging to a given CDB.

When queried from a CDB root, this view describes all PDBs that belong to the CDB. When queried from an application root, it describes all PDBs that belong to the application container. When queried from a regular PDB or from an application PDB, it describes the regular PDB or the application PDB.

Column	Datatype	NULL	Description
PDB_ID	NUMBER	NOT NULL	Container ID of the PDB
PDB_NAME	VARCHAR2(128)	NOT NULL	Name of the PDB
DBID	NUMBER	NOT NULL	PDB identifier calculated when the PDB is created and stored in all file headers associated with the PDB
CON_UID	NUMBER	NOT NULL	Unique identifier associated with the container
GUID	RAW(16)		Globally unique immutable ID assigned to the PDB at creation time

Column	Datatype	NULL	Description
STATUS	VARCHAR2(10)		State of the PDB. Possible values: • NEW - The PDB has never been opened since it was created. It must be opened in READ WRITE mode for Oracle to perform processing needed to complete the integration of the PDB into the CDB and mark it NORMAL. An error will be thrown if an attempt is made to open the PDB read only. • NORMAL - The PDB is ready to be used. • UNPLUGGED - The PDB has been unplugged. The only operation that can be performed on it is DROP PLUGGABLE DATABASE. • RELOCATING: The PDB is in the process of being relocated to a different CDB. • RELOCATED: The PDB has been relocated to a different CDB. • REFRESHING: The PDB is a refresh PDB. • STANDBY: The PDB is a Data Guard standby PDB. • UNDEFINED: The PDB is in an undefined state. • UNUSABLE - The PDB is being created or an unrecoverable error was encountered during its creation. The PDB cannot be opened while its state is set to UNUSABLE. If the PDB remains in this state because of an error encountered during its creation, it can only be dropped. The alert log can be checked to determine if there was an error during PDB creation.
CREATION_SCN	NUMBER		Creation SCN
VSN	NUMBER		The version number of the PDB
LOGGING	VARCHAR2(9)		Shows the current logging mode for the PDB. Possible values: • LOGGING • NOLOGGING
FORCE_LOGGING	VARCHAR2(3)		Specifies whether force logging is turned on for the PDB. Possible values: • NO • YES
FORCE_NOLOGGING	VARCHAR2(3)		Specifies whether force nologging is turned on for the PDB. Possible values: • NO • YES
APPLICATION_ROOT	VARCHAR2(3)		Indicates whether the PDB is an application root.
APPLICATION_PDB	VARCHAR2(3)		Indicates whether a PDB is an application PDB
APPLICATION_SEED	VARCHAR2(3)		Indicates whether a 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.

Column	Datatype	NULL	Description
IS_PROXY_PDB	VARCHAR2(3)		Indicates whether this PDB is a proxy PDB
CON_ID	NUMBER	NOT NULL	The ID of the container that CON_DBID identifies. 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
UPGRADE_PRIORITY	NUMBER		The upgrade priority of the PDB.
APPLICATION_CLONE	VARCHAR2(3)		Indicates whether this PDB is an application root clone (YES) or not (NO)
FOREIGN_CDB_DBID	NUMBER		The foreign CDB's DBID
UNPLUG_SCN	NUMBER		SCN at which the PDB was unplugged
FOREIGN_PDB_ID	NUMBER		The foreign PDB ID
CREATION_TIME	DATE	NOT NULL	PDB creation timestamp
REFRESH_MODE	VARCHAR2(6)		PDB refresh mode. Possible values: • MANUAL • AUTO
REFRESH_INTERVAL	NUMBER		PDB refresh interval. This is applicable only when REFRESH_MODE is AUTO.
TEMPLATE	VARCHAR2(3)		For internal use only
LAST_REFRESH_SCN	NUMBER		System change number (SCN) of the last refresh operation
TENANT_ID	VARCHAR2(255)		Pluggable database tenant key
SNAPSHOT_MODE	VARCHAR2(6)		Pluggable database snapshot mode
SNAPSHOT_INTERVAL	NUMBER		Pluggable database snapshot interval, in minutes
CREDENTIAL_NAME	VARCHAR2(262)		Credential object name associated with the PDB
LAST_REFRESH_TIME	DATE		Time of the last refresh operation
CLOUD_IDENTITY	VARCHAR2(32767)		Cloud identifier for the PDB
SOURCE_PDB_NAME	VARCHAR2(128)		Name of the source PDB
SOURCE_DB_LINK	VARCHAR2(128)		Name of the source database link
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.

6.270 DBA_PENDING_CONV_TABLES

DBA_PENDING_CONV_TABLES describes all pending conversion tables in the database. Its columns are the same as those in ALL_PENDING_CONV_TABLES.

> **See Also:**
>
> "ALL_PENDING_CONV_TABLES"

6.271 DBA_PENDING_TRANSACTIONS

DBA_PENDING_TRANSACTIONS describes unresolved transactions (either due to failure or if the coordinator has not sent a commit/rollback).

Column	Datatype	NULL	Description
FORMATID	NUMBER		The format identifier of the transaction identifier
GLOBALID	RAW(64)		The global part (gtrid) of the transaction identifier
BRANCHID	RAW(64)		The branch qualifier (bqual) of the transaction identifier

6.272 DBA_PG_EDGE_RELATIONSHIPS

DBA_PG_EDGE_RELATIONSHIPS describes edge relationships in all property graphs in the database. Its columns are the same as those in ALL_PG_EDGE_RELATIONSHIPS.

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

> **See Also:**
>
> "ALL_PG_EDGE_RELATIONSHIPS"

6.273 DBA_PG_ELEMENT_LABELS

DBA_PG_ELEMENT_LABELS describes labels for the element tables in all property graphs in the database. Its columns are the same as those in ALL_PG_ELEMENT_LABELS.

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

> **See Also:**
>
> "ALL_PG_ELEMENT_LABELS"

6.274 DBA_PG_ELEMENTS

`DBA_PG_ELEMENTS` describes element tables in all property graphs in the database. Its columns are the same as those in `ALL_PG_ELEMENTS`.

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

> **See Also:**
>
> "ALL_PG_ELEMENTS"

6.275 DBA_PG_KEYS

`DBA_PG_KEYS` describes key columns in all property graphs in the database. Its columns are the same as those in `ALL_PG_KEYS`.

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

> **See Also:**
>
> "ALL_PG_KEYS"

6.276 DBA_PG_LABEL_PROPERTIES

`DBA_PG_LABEL_PROPERTIES` describes properties in all property graphs in the database. Its columns are the same as those in `ALL_PG_LABEL_PROPERTIES`.

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

> **See Also:**
>
> "ALL_PG_LABEL_PROPERTIES"

6.277 DBA_PG_LABELS

`DBA_PG_LABELS` describes labels in all property graphs in the database. Its columns are the same as those in `ALL_PG_LABELS`.

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

> **See Also:**
>
> "ALL_PG_LABELS"

6.278 DBA_PG_PROP_DEFINITIONS

`DBA_PG_PROP_DEFINITIONS` describes columns and expressions exposed as properties in all property graphs in the database. Its columns are the same as those in `ALL_PG_PROP_DEFINITIONS`.

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

> **See Also:**
>
> "ALL_PG_PROP_DEFINITIONS"

6.279 DBA_PLSQL_COLL_TYPES

DBA_PLSQL_COLL_TYPES describes all named PL/SQL collection types in the database. Its columns (except for CHAR_USED) are the same as those in ALL_PLSQL_COLL_TYPES.

> **See Also:**
>
> "ALL_PLSQL_COLL_TYPES"

6.280 DBA_PLSQL_OBJECT_SETTINGS

DBA_PLSQL_OBJECT_SETTINGS displays information about the compiler settings for all stored objects in the database. Its columns are the same as those in ALL_PLSQL_OBJECT_SETTINGS.

> **See Also:**
>
> "ALL_PLSQL_OBJECT_SETTINGS"

6.281 DBA_PLSQL_TYPE_ATTRS

DBA_PLSQL_TYPE_ATTRS describes the attributes of all PL/SQL types in the database. Its columns are the same as those in ALL_PLSQL_TYPE_ATTRS.

> **See Also:**
>
> "ALL_PLSQL_TYPE_ATTRS"

6.282 DBA_PLSQL_TYPES

DBA_PLSQL_TYPES describes all PL/SQL types in the database. Its columns are the same as those in ALL_PLSQL_TYPES.

> **See Also:**
>
> "ALL_PLSQL_TYPES"

6.283 DBA_POLICIES

DBA_POLICIES describes all Oracle Virtual Private Database (VPD) security policies in the database. Its columns are the same as those in ALL_POLICIES.

A security policy is a list of security requirements and rules that regulate row level access to database objects.

> **See Also:**
>
> "ALL_POLICIES"

6.284 DBA_POLICY_ATTRIBUTES

DBA_POLICY_ATTRIBUTES lists the attribute associations {Namespaces, Attributes} of all context-sensitive and shared context-sensitive Oracle Virtual Private Database (VPD) policies in the database. Its columns are the same as those in ALL_POLICY_ATTRIBUTES.

> **See Also:**
>
> "ALL_POLICY_ATTRIBUTES"

6.285 DBA_POLICY_CONTEXTS

DBA_POLICY_CONTEXTS describes all driving contexts in the database. Its columns are the same as those in ALL_POLICY_CONTEXTS.

> **See Also:**
>
> "ALL_POLICY_CONTEXTS"

6.286 DBA_POLICY_GROUPS

DBA_POLICY_GROUPS describes all policy groups in the database. Its columns are the same as those in ALL_POLICY_GROUPS.

> **See Also:**
>
> "ALL_POLICY_GROUPS"

6.287 DBA_PRIV_AUDIT_OPTS

DBA_PRIV_AUDIT_OPTS describes current system privileges being audited across the system and by user.

> **Note:**
>
> This view is deprecated and applies only to traditional auditing. Traditional auditing is desupported starting in Oracle Database 23ai. Though traditional auditing is desupported, any current traditional audit settings that you have will still be honored and are viewable with this view. See *Oracle Database Security Guide* for more information about how this desupport works.

Column	Datatype	NULL	Description
USER_NAME	VARCHAR2(128)		User name if by user auditing; ANY CLIENT if access by a proxy on behalf of a client is being audited; NULL for systemwide auditing
PROXY_NAME	VARCHAR2(128)		Name of the proxy user which is performing an operation for the client; NULL if the client is performing the operation directly
PRIVILEGE	VARCHAR2(40)	NOT NULL	Name of the system privilege being audited
SUCCESS	VARCHAR2(10)		Mode for WHENEVER SUCCESSFUL system auditing
FAILURE	VARCHAR2(10)		Mode for WHENEVER NOT SUCCESSFUL system auditing

6.288 DBA_PRIV_CAPTURES

DBA_PRIV_CAPTURES lists the privilege analysis policies in the database.

Column	Datatype	NULL	Description
NAME	VARCHAR2(128)	NOT NULL	Name of the privilege analysis policy
DESCRIPTION	VARCHAR2(1024)		Description of the privilege analysis
TYPE	VARCHAR2(16)		Type of the privilege analysis policy. Possible values: • G_DATABASE: Database wide privilege analysis • G_ROLE: Role privilege analysis • G_CONTEXT: Context privilege analysis • G_ROLE_AND_CONTEXT: Role and context privilege analysis
ENABLED	VARCHAR2(1)		Enabling status of the privilege analysis
ROLES	ROLE_ID_LIST		List of roles whose privileges to analyze if the privilege analysis type is G_ROLE or G_ROLE_AND_CONTEXT
CONTEXT	VARCHAR2(4000)		Context condition if the privilege analysis type is G_CONTEXT or G_ROLE_AND_CONTEXT
RUN_NAME	VARCHAR2(128)		Displays run name information for each run

> **See Also:**
>
> *Oracle Database Security Guide* for more information about privilege analysis

6.289 DBA_PRIVATE_TEMP_TABLES

`DBA_PRIVATE_TEMP_TABLES` describes all of the private temporary tables in the database.

Related View

`USER_PRIVATE_TEMP_TABLES` describes the private temporary tables in the current session. This view does not display the `INST_ID` column.

Column	Datatype	NULL	Description
`SID`	`NUMBER`		Session ID of the session that created the private temporary table
`SERIAL#`	`NUMBER`		Session serial number of the session that created the private temporary table
`INST_ID`	`NUMBER`		Instance ID of the session that created the private temporary table
`OWNER`	`VARCHAR2(128)`		Owner name of the private temporary table
`TABLE_NAME`	`VARCHAR2(128)`		Private temporary table name
`TABLESPACE_NAME`	`VARCHAR2(128)`		Private temporary table's tablespace name
`DURATION`	`VARCHAR2(128)`		Private temporary table's duration (for example, `SESSION` or `TRANSACTION`)
`NUM_ROWS`	`NUMBER`		Number of rows in the private temporary table when analyzed
`BLOCKS`	`NUMBER`		Number of blocks used by private temporary table
`AVG_ROW_LEN`	`NUMBER`		Average row length
`LAST_ANALYZED`	`DATE`		Timestamp of last analyze
`TXN_ID`	`RAW(8)`		Transaction ID of the transaction duration private temporary table
`SAVE_POINT_NUM`	`NUMBER`		Save point number of the transaction duration private temporary table

> **See Also:**
>
> - "USER_PRIVATE_TEMP_TABLES"
> - "PRIVATE_TEMP_TABLE_PREFIX"
> - *Oracle Database Administrator's Guide* for an introduction to private temporary tables

6.290 DBA_PROCEDURES

DBA_PROCEDURES lists all functions and procedures that are available in the database, along with their associated properties. Its columns are the same as those in ALL_PROCEDURES.

See Also:

- "ALL_PROCEDURES"
- "DBA_ARGUMENTS" for information about the arguments of all of the functions and procedures that are available in the database

6.291 DBA_PROFILES

DBA_PROFILES displays all profiles and their limits.

Column	Datatype	NULL	Description
PROFILE	VARCHAR2(128)	NOT NULL	Profile name
RESOURCE_NAME	VARCHAR2(32)	NOT NULL	Resource name
RESOURCE_TYPE	VARCHAR2(8)		Indicates whether the resource profile is a KERNEL or a PASSWORD parameter
LIMIT	VARCHAR2(128)		Limit placed on this resource for the profile
COMMON	VARCHAR2(3)		Indicates whether the profile is common. Possible values: • YES if a profile is common • NO if a profile is local (not common)
INHERITED	VARCHAR2(3)		Indicates whether the profile definition was inherited from another container (YES) or not (NO)
IMPLICIT	VARCHAR2(3)		Indicates whether the profile was created by an implicit application (YES) or not (NO)
ORACLE_MAINTAINED	VARCHAR2(3)		Indicates whether the profile was created by Oracle-supplied scripts (YES) or not (NO) Note that a profile created by Oracle-supplied scripts is local to all PDBs and can be changed inside a PDB to accommodate individual application requirements.
MANDATORY	VARCHAR2(3)		Indicates whether the profile is a mandatory user profile (YES) or not (NO) **See Also:** The MANDATORY_USER_PROFILE initialization parameter

6.292 DBA_PROPERTY_GRAPHS

DBA_PROPERTY_GRAPHS describes all property graphs in the database. Its columns are the same as those in ALL_PROPERTY_GRAPHS.

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

> **See Also:**
>
> "ALL_PROPERTY_GRAPHS"

6.293 DBA_PROPAGATION

DBA_PROPAGATION displays information about all propagations in the database. Its columns are the same as those in ALL_PROPAGATION.

> **See Also:**
>
> "ALL_PROPAGATION"

6.294 DBA_PROXIES

DBA_PROXIES displays information about all proxy connections in the database.

Related View

USER_PROXIES displays information about connections the current user is allowed to proxy. This view does not display the PROXY or PROXY_AUTHORITY columns.

Column	Datatype	NULL	Description
PROXY	VARCHAR2(128)		Name of the proxy user
CLIENT	VARCHAR2(128)	NOT NULL	Name of the client user who the proxy user can act on behalf of
AUTHENTICATION	VARCHAR2(3)		Indicates whether the proxy is required to supply the client's authentication credentials (YES) or not (NO)
AUTHORIZATION_CONSTRAINT	VARCHAR2(35)		Indicates the proxy's authority to exercise roles on the client's behalf: • NO CLIENT ROLES MAY BE ACTIVATED • PROXY MAY ACTIVATE ROLE • PROXY MAY ACTIVATE ALL CLIENT ROLES • PROXY MAY NOT ACTIVATE ROLE

Column	Datatype	NULL	Description
ROLE	VARCHAR2(128)		Name of the role referenced in AUTHORIZATION_CONSTRAINT
PROXY_AUTHORITY	VARCHAR2(9)		Value is either: • DIRECTORY if an EUS proxy is enabled for the database user • DATABASE if this row describes a local proxy permission

See Also:

- "USER_PROXIES"
- "PROXY_USERS" for a list of proxy users and the clients on whose behalf they can act

6.295 DBA_QUARANTINED_TRANSACTIONS

DBA_QUARANTINED_TRANSACTIONS lists all quarantined transactions in the current container.

Column	Datatype	NULL	Description
USN	NUMBER	NOT NULL	Undo segment number of the quarantined transaction
SLT	NUMBER	NOT NULL	Slot number of the quarantined transaction
SQN	NUMBER	NOT NULL	Sequence number of the quarantined transaction
REASON	VARCHAR2(256)		Reason the transaction failed to recover
TRACE_FILE_NAME	VARCHAR2(4096)		Name of the trace file containing the reason for the transaction's recovery failure, along with diagnostic information about the failure
UNDO_TSN	NUMBER		Undo tablespace number for the quarantined transaction
UNDO_RECORD_OBJN	NUMBER		Dictionary object number of the object (OBJN)
UNDO_RECORD_OBJD	NUMBER		Dictionary object number of the segment that contains the object (OBJD)
DATA_BLOCK_TSN	NUMBER		Tablespace ID for the object
PREV_UNDO_BLOCK_DBA	NUMBER		Previous undo block address, which was used for rollback
UBA_RDBA	NUMBER		Relative data block address of the current undo block being applied for rollback
UBA_SQN	NUMBER		Undo block sequence number of the current undo block being applied for rollback
UBA_RECORD_NUMBER	NUMBER		Undo record number of the current undo block being applied for rollback
TXN_START_SCN	NUMBER		Start SCN of the quarantined transaction
INCIDENT_TIME	VARCHAR2(64)		Time at which the incident occurred

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

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

6.296 DBA_QUEUE_EVENT_STREAMS

`DBA_QUEUE_EVENT_STREAMS` describes all Transactional Event Queue (TxEventQ) event streams in the database. Its columns are the same as those in `ALL_QUEUE_EVENT_STREAMS`.

> **See Also:**
>
> "ALL_QUEUE_EVENT_STREAMS"

6.297 DBA_QUEUE_SCHEDULES

`DBA_QUEUE_SCHEDULES` describes all propagation schedules in the database. Its columns are the same as those in `ALL_QUEUE_SCHEDULES`.

> **See Also:**
>
> "ALL_QUEUE_SCHEDULES"

6.298 DBA_QUEUE_SUBSCRIBERS

`DBA_QUEUE_SUBSCRIBERS` displays all subscribers on all queues in the database. Its columns are the same as those in `ALL_QUEUE_SUBSCRIBERS`.

> **See Also:**
>
> "ALL_QUEUE_SUBSCRIBERS"

6.299 DBA_QUEUE_TABLES

DBA_QUEUE_TABLES contains information about the owner instance for a queue table.

A queue table can contain multiple queues. In this case, each queue in a queue table has the same owner instance as the queue table. Its columns are the same as those in ALL_QUEUE_TABLES.

> **See Also:**
>
> "ALL_QUEUE_TABLES"

6.300 DBA_QUEUES

DBA_QUEUES describes the operational characteristics of every queue in a database. Its columns are the same as those in ALL_QUEUES.

> **See Also:**
>
> - "ALL_QUEUES"
> - *Oracle Database Advanced Queuing User's Guide* for more information about Advanced Queuing

6.301 DBA_RAT_CAPTURE_SCHEMA_INFO

DBA_RAT_CAPTURE_SCHEMA_INFO displays the login schema and current schema that were in effect when SQL statements were recorded in a workload capture.

This view is useful when you perform a workload replay in extended PL/SQL mode. This type of replay may include SQL statements that perform table operations such as SELECT, UPDATE, and DELETE. If the current schema was different from the login schema at the time of the workload capture, then those table operations may have been performed with the privileges of the current user, not the login user. During workload replay, all operations are performed with the privileges of the login user. Therefore, errors can occur during replay if the login user does not have the necessary privileges to perform the table operations.

To resolve this issue, you can use this view in conjunction with the DBA_WORKLOAD_CAPTURE_SQLTEXT view. Join the CAPTURE_ID column in this view with the CAPTURE_ID column in DBA_WORKLOAD_CAPTURE_SQLTEXT to determine the login schema and current schema that were in effect when each SQL statement in DBA_WORKLOAD_CAPTURE_SQLTEXT was captured. Examine the SQL_TEXT column in DBA_WORKLOAD_CAPTURE_SQLTEXT to determine whether the SQL statement involved any table operations, and whether those table operations were performed with the privileges of the current user or the login user. You can then grant to the login user the necessary privileges for performing those table operations before performing a workload replay.

Column	Datatype	NULL	Description
CAPTURE_ID	NUMBER(38)		Internal key for the workload capture
SQL_ID	VARCHAR2(13)		SQL identifier for the parent cursor in the library cache
LOGIN_SCHEMA	VARCHAR2(128)		The schema of the user who logged on to the session in which the SQL statement was recorded in the workload capture. This value does not change during a session.
CURRENT_SCHEMA	VARCHAR2(128)		The currently active default schema for the session when the SQL statement was recorded in the workload capture. This value may change during a session through use of an ALTER SESSION SET CURRENT_SCHEMA statement. It may also change during a session to reflect the owner of any active definer's rights object.

See Also:

"DBA_WORKLOAD_CAPTURE_SQLTEXT"

6.302 DBA_RCHILD

DBA_RCHILD displays all the children in any refresh group.

Column	Datatype	NULL	Description
REFGROUP	NUMBER		Internal identifier of the refresh group
OWNER	VARCHAR2(128)	NOT NULL	Owner of the object in the refresh group
NAME	VARCHAR2(128)	NOT NULL	Name of the object in the refresh group
TYPE#	VARCHAR2(128)		Type of the object in the refresh group

6.303 DBA_RECOVERABLE_SCRIPT

DBA_RECOVERABLE_SCRIPT provides details about recoverable operations.

Column	Datatype	NULL	Description
SCRIPT_ID	RAW(16)		Unique ID of the operation
CREATION_TIME	DATE		Time the operation was invoked
INVOKING_PACKAGE_OWNER	VARCHAR2(128)		Invoking package owner of the operation
INVOKING_PACKAGE	VARCHAR2(128)		Invoking package of the operation
INVOKING_PROCEDURE	VARCHAR2(128)		Invoking procedure of the operation
INVOKING_USER	VARCHAR2(128)		Script owner
STATUS	VARCHAR2(12)		State of the recoverable script: GENERATING, NOT EXECUTED, EXECUTING, EXECUTED, or ERROR
TOTAL_BLOCKS	NUMBER		Total number of blocks for the recoverable script to be executed

Column	Datatype	NULL	Description
DONE_BLOCK_NUM	NUMBER		Last block executed, thus far
SCRIPT_COMMENT	VARCHAR2(4000)		Comment for the recoverable script

6.304 DBA_RECOVERABLE_SCRIPT_BLOCKS

DBA_RECOVERABLE_SCRIPT_BLOCKS provides details about recoverable script blocks.

Column	Datatype	NULL	Description
SCRIPT_ID	RAW(16)		Global unique ID of the recoverable script to which this block belongs
BLOCK_NUM	NUMBER		The *n*th block in the recoverable script to be executed
FORWARD_BLOCK	CLOB		Forward block to be executed
FORWARD_BLOCK_DBLINK	VARCHAR2(128)		Database where the forward block is executed
UNDO_BLOCK	CLOB		Block to roll back the forward operation
UNDO_BLOCK_DBLINK	VARCHAR2(128)		Database where the undo block is executed
STATUS	VARCHAR2(12)		Status of the block execution: GENERATING, NOT EXECUTED, EXECUTING, EXECUTED, or ERROR
BLOCK_COMMENT	VARCHAR2(4000)		Comment for the block

6.305 DBA_RECOVERABLE_SCRIPT_ERRORS

DBA_RECOVERABLE_SCRIPT_ERRORS provides details about errors that occurred during script execution.

Column	Datatype	NULL	Description
SCRIPT_ID	RAW(16)		Global unique ID of the recoverable script
BLOCK_NUM	NUMBER		The *n*th block that failed
ERROR_NUMBER	NUMBER		Number of the error encountered while executing the block
ERROR_MESSAGE	VARCHAR2(4000)		Error message encountered while executing the block
ERROR_CREATION_TIME	DATE		Time that the error was created

6.306 DBA_RECOVERABLE_SCRIPT_HIST

DBA_RECOVERABLE_SCRIPT_HIST displays details about executed or purged recoverable operations.

Column	Datatype	NULL	Description
SCRIPT_ID	RAW(16)		Unique id of the operation
CREATION_TIME	DATE		Time the operation was invoked
INVOKING_PACKAGE_OWNER	VARCHAR2(128)		Invoking package owner of the operation
INVOKING_PACKAGE	VARCHAR2(128)		Invoking package of the operation

Column	Datatype	NULL	Description
INVOKING_PROCEDURE	VARCHAR2(128)		Invoking procedure of the operation
INVOKING_USER	VARCHAR2(128)		Script owner
STATUS	VARCHAR2(8)		state of the recoverable script: EXECUTED, PURGED
TOTAL_BLOCKS	NUMBER		total number of blocks for the recoverable script to be executed
DONE_BLOCK_NUM	NUMBER		last block so far executed
SCRIPT_COMMENT	VARCHAR2(4000)		comment for the recoverable script

6.307 DBA_RECOVERABLE_SCRIPT_PARAMS

DBA_RECOVERABLE_SCRIPT_PARAMS provides details about recoverable operation parameters.

Column	Datatype	NULL	Description
SCRIPT_ID	RAW(16)		Unique ID of the operation
PARAMETER	VARCHAR2(128)		Name of the parameter
PARAM_INDEX	NUMBER		Index for multi-valued parameter
VALUE	VARCHAR2(4000)		Value of the parameter

6.308 DBA_RECYCLEBIN

DBA_RECYCLEBIN displays information about all recycle bins in the database.

Related View

USER_RECYCLEBIN displays information about the recycle bin owned by the current user. This view does not display the OWNER column.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)	NOT NULL	Name of the original owner of the object
OBJECT_NAME	VARCHAR2(128)	NOT NULL	New name of the object
ORIGINAL_NAME	VARCHAR2(128)		Original name of the object
OPERATION	VARCHAR2(9)		Operation carried out on the object: • DROP - Object was dropped • TRUNCATE - Object was truncated **Note:** The Oracle Database currently only supports recovering dropped objects from the recycle bin. The truncated objects cannot be recovered.

Column	Datatype	NULL	Description
TYPE	VARCHAR2(25)		Type of the object: • TABLE • NORMAL INDEX • BITMAP INDEX • NESTED TABLE • LOB • LOB INDEX • DOMAIN INDEX • IOT TOP INDEX • IOT OVERFLOW SEGMENT • IOT MAPPING TABLE • TRIGGER • Table Partition • Table Composite Partition • Index Partition • Index Composite Partition • LOB Partition • LOB Composite Partition
TS_NAME	VARCHAR2(30)		Name of the tablespace to which the object belongs
CREATETIME	VARCHAR2(19)		Timestamp for the creation of the object
DROPTIME	VARCHAR2(19)		Timestamp for the dropping of the object
DROPSCN	NUMBER		System change number (SCN) of the transaction which moved the object to the recycle bin
PARTITION_NAME	VARCHAR2(128)		Name of the partition which was dropped
CAN_UNDROP	VARCHAR2(3)		Indicates whether the object can be undropped (YES) or not (NO)
CAN_PURGE	VARCHAR2(3)		Indicates whether the object can be purged (YES) or not (NO)
RELATED	NUMBER	NOT NULL	Object number of the parent object
BASE_OBJECT	NUMBER	NOT NULL	Object number of the base object
PURGE_OBJECT	NUMBER	NOT NULL	Object number for the object which gets purged
SPACE	NUMBER		Number of blocks used by the object

See Also:

"USER_RECYCLEBIN"

6.309 DBA_REDEFINITION_ERRORS

DBA_REDEFINITION_ERRORS is an online redefinition view. It displays the dependent objects for which errors were raised while attempting to create similar objects on the interim table of the redefinition.

Column	Datatype	NULL	Description
OBJECT_TYPE	VARCHAR2(12)		Type of the redefinition object: • TABLE • INDEX • CONSTRAINT • TRIGGER • NESTED TABLE • PARTITION • MV LOG
OBJECT_OWNER	VARCHAR2(4000)		Owner of the redefinition object
OBJECT_NAME	VARCHAR2(128)	NOT NULL	Name of the redefinition object
BASE_TABLE_OWNER	VARCHAR2(128)		Owner of the base table of the redefinition object
BASE_TABLE_NAME	VARCHAR2(128)		Name of the base table of the redefinition object
DDL_TXT	CLOB		DDL used to create the corresponding interim redefinition object
EDITION_NAME	VARCHAR2(128)		Reserved for future use
ERR_NO	NUMBER(38)		Oracle error number corresponding to this error
ERR_TXT	VARCHAR2(1000)		Oracle error text corresponding to this error

See Also:

Oracle Database Administrator's Guide for more information about online redefinition

6.310 DBA_REDEFINITION_OBJECTS

DBA_REDEFINITION_OBJECTS is an online redefinition view. It displays the objects involved in the current redefinitions.

Column	Datatype	NULL	Description
OBJECT_TYPE	VARCHAR2(12)		Type of the redefinition object: • TABLE • INDEX • CONSTRAINT • TRIGGER • NESTED TABLE • PARTITION • MV LOG
OBJECT_OWNER	VARCHAR2(4000)		Owner of the redefinition object
OBJECT_NAME	VARCHAR2(128)	NOT NULL	Name of the redefinition object
BASE_TABLE_OWNER	VARCHAR2(128)		Owner of the base table of the redefinition object
BASE_TABLE_NAME	VARCHAR2(128)		Name of the base table of the redefinition object
INTERIM_OBJECT_OWNER	VARCHAR2(4000)		Owner of the corresponding interim redefinition object
INTERIM_OBJECT_NAME	VARCHAR2(128)		Name of the corresponding interim redefinition object
EDITION_NAME	VARCHAR2(128)		Reserved for future use

> **See Also:**
>
> *Oracle Database Administrator's Guide* for more information about online redefinition

6.311 DBA_REDEFINITION_STATUS

DBA_REDEFINITION_STATUS is an online redefinition view. It provides information about the online redefinition status.

Column	Datatype	NULL	Description
REDEFINITION_ID	NUMBER(38)	NOT NULL	ID for the redefinition object
BASE_TABLE_OWNER	VARCHAR2(128)		Owner of the base table of the redefinition object
BASE_TABLE_NAME	VARCHAR2(128)		Name of the base table of the redefinition object
BASE_OBJECT_NAME	VARCHAR2(128)	NOT NULL	Name of the base object of the redefinition object
BASE_OBJECT_TYPE	VARCHAR2(9)		Type of the base object of the redefinition object
INTERIM_OBJECT_OWNER	VARCHAR2(128)		Owner of the interim object of the redefinition object
INTERIM_OBJECT_NAME	VARCHAR2(128)		Name of the interim object of the redefinition object
OPERATION	VARCHAR2(128)	NOT NULL	The current redefinition operation: • START_REDEF_TABLE • SYNC_INTERIM_TABLE • COPY_TABLE_DEPENDENTS • FINISH_REDEF_TABLE
STATUS	VARCHAR2(128)	NOT NULL	Status of the previous redefinition operation: • FAILURE • SUCCESS
RESTARTABLE	VARCHAR2(1)	NOT NULL	Indicates whether the previous operation can be restarted
ERR_TXT	VARCHAR2(1000)		The error message raised from the previous operation
ACTION	VARCHAR2(400)		The suggested action
REFRESH_DEP_MVIEWS	VARCHAR2(1)		Indicates whether the online redefinition will also refresh dependent materialized views when syncing the interim table (Y) or not (N)

> **See Also:**
>
> *Oracle Database Administrator's Guide* for more information about online redefinition

6.312 DBA_REFRESH

DBA_REFRESH describes all refresh groups in the database. Its columns are the same as those in ALL_REFRESH.

> **See Also:**
>
> "ALL_REFRESH"

6.313 DBA_REFRESH_CHILDREN

DBA_REFRESH_CHILDREN lists all of the objects in all refresh groups in the database. Its columns are the same as those in ALL_REFRESH_CHILDREN.

> **See Also:**
>
> "ALL_REFRESH_CHILDREN"

6.314 DBA_REFS

DBA_REFS describes the REF columns and REF attributes in object type columns of all the objects in the database. Its columns are the same as those in ALL_REFS.

> **See Also:**
>
> "ALL_REFS"

6.315 DBA_REGISTERED_ARCHIVED_LOG

DBA_REGISTERED_ARCHIVED_LOG displays information about all registered archived logfiles in the database.

Column	Datatype	NULL	Description
CONSUMER_NAME	VARCHAR2(128)	NOT NULL	Consumer name of the archived logs
SOURCE_DATABASE	VARCHAR2(128)		Name of the database which generated the redo logs
THREAD#	NUMBER	NOT NULL	Thread number of the archived redo log. The thread number is 1 for a single instance. For Real Application Clusters, this column will contain different numbers.
SEQUENCE#	NUMBER	NOT NULL	Sequence number of the archived redo log file
FIRST_SCN	NUMBER	NOT NULL	System change number (SCN) of the current archived redo log

Column	Datatype	NULL	Description
NEXT_SCN	NUMBER		System change number (SCN) of the next archived redo log
FIRST_TIME	DATE		Date and time of the current archived redo log
NEXT_TIME	DATE		Date and time of the next archived redo log
NAME	VARCHAR2(513)		Name of the archived redo log
MODIFIED_TIME	DATE		Time when the archived redo log was registered
DICTIONARY_BEGIN	VARCHAR2(3)		Indicates whether the beginning of the dictionary build is in the archived redo log (YES) or not (NO)
DICTIONARY_END	VARCHAR2(3)		Indicates whether the end of the dictionary build is in the archived redo log (YES) or not (NO)
PURGEABLE	VARCHAR2(3)		Indicates whether the redo log can be permanently removed (YES) or not (NO)
RESET_LOGS_CHANGE#	NUMBER	NOT NULL	Resetlogs change number of the database when the log was written
RESET_TIMESTAMP	NUMBER	NOT NULL	Resetlogs time of the database when the log was written

6.316 DBA_REGISTERED_MVIEWS

DBA_REGISTERED_MVIEWS describes all registered materialized views (registered at a master site or a master materialized view site) in the database. Its columns are the same as those in ALL_REGISTERED_MVIEWS.

> **See Also:**
>
> "ALL_REGISTERED_MVIEWS"

6.317 DBA_REGISTRY

DBA_REGISTRY displays information about all components in the database that are loaded into the component registry.

The component registry tracks components that can be separately loaded into the Oracle Database. When a SQL script loads the PL/SQL packages and other database objects for a component into the database, the script records the component name, status, and version. If scripts are used to upgrade/downgrade the dictionary elements for the component, then those scripts also record status and version information.

Related View

USER_REGISTRY displays information about the components owned by the current user that are loaded into the component registry.

Column	Datatype	NULL	Description
COMP_ID	VARCHAR2(30)	NOT NULL	Component identifier

Column	Datatype	NULL	Description
COMP_NAME	VARCHAR2(255)		Component name
VERSION	VARCHAR2(30)		Component version loaded
VERSION_FULL	VARCHAR2(30)		Component full version
STATUS	VARCHAR2(11)		Component status: • INVALID • VALID • LOADING • LOADED • UPGRADING • UPGRADED • DOWNGRADING • DOWNGRADED • REMOVING • REMOVED
MODIFIED	VARCHAR2(20)		Time when the component was last modified
NAMESPACE	VARCHAR2(30)	NOT NULL	Component namespace
CONTROL	VARCHAR2(128)	NOT NULL	User that created the component entry
SCHEMA	VARCHAR2(128)	NOT NULL	User that contains the objects for the component
PROCEDURE	VARCHAR2(61)		Validation procedure
STARTUP	VARCHAR2(8)		Indicates whether the component requires a startup after the upgrade (REQUIRED) or not
PARENT_ID	VARCHAR2(30)		Parent component identifier
OTHER_SCHEMAS	VARCHAR2(4000)		A list of ancillary schema names associated with the component

See Also:

"USER_REGISTRY"

6.318 DBA_REGISTRY_BACKPORTS

DBA_REGISTRY_BACKPORTS displays backported bug fixes that were applied to the database. This view displays only bug fixes that changed the data dictionary of the database.

Column	Datatype	NULL	Description
BUGNO	NUMBER	NOT NULL	Bug number
VERSION_FULL	VARCHAR2(30)	NOT NULL	Component full version
COMP_ID	VARCHAR2(30)	NOT NULL	Component identifier
NAMESPACE	VARCHAR2(30)	NOT NULL	Component namespace
BACKPORT_TYPE	VARCHAR2(30)	NOT NULL	Type of backported bug fix The value of this columns is always DICTIONARY, which indicates that the bug fix changed the data dictionary of the database.

Column	Datatype	NULL	Description
BACKPORT_TIME	TIMESTAMP(6)	NOT NULL	Date and time at which the backported bug fix was applied

6.319 DBA_REGISTRY_HIERARCHY

DBA_REGISTRY_HIERARCHY displays information about the components loaded into the database, grouped by owner and organized in the component hierarchy.

Column	Datatype	NULL	Description
NAMESPACE	VARCHAR2(30)	NOT NULL	Component namespace
COMP_ID	VARCHAR2(4000)		Component identifier
VERSION	VARCHAR2(30)		Component version loaded
VERSION_FULL	VARCHAR2(30)		Component full version
STATUS	VARCHAR2(11)		Component status: • INVALID • VALID • LOADING • LOADED • UPGRADING • UPGRADED • DOWNGRADING • DOWNGRADED • REMOVING • REMOVED
MODIFIED	VARCHAR2(20)		Time when the component was last modified

6.320 DBA_REGISTRY_HISTORY

DBA_REGISTRY_HISTORY provides information about upgrades, downgrades, and critical patch updates that have been performed on the database.

Column	Datatype	NULL	Description
ACTION_TIME	TIMESTAMP(6)		The time the upgrade, downgrade, or patch action was completed
ACTION	VARCHAR2(30)		The specific action (for example, UPGRADE or DOWNGRADE)
NAMESPACE	VARCHAR2(30)		The namespace of the components affected (for example, SERVER)
VERSION	VARCHAR2(30)		The version number of the server (for example, 10.2.0.1.0)
ID	NUMBER		Bundle ID
COMMENTS	VARCHAR2(255)		Additional comments about the action taken
BUNDLE_SERIES	VARCHAR2(30)		If a bundle patch, the series (for example, PSU or DBBP)

6.321 DBA_REGISTRY_LOG

DBA_REGISTRY_LOG displays operating information about components loaded into the database.

Column	Datatype	NULL	Description
OPTIME	VARCHAR2(20)		Operation time
NAMESPACE	VARCHAR2(30)		Component namespace
COMP_ID	VARCHAR2(30)		Component identifier
OPERATION	VARCHAR2(11)		Operation name
MESSAGE	VARCHAR2(1000)		Message

6.322 DBA_REGISTRY_SCHEMAS

DBA_REGISTRY_SCHEMAS lists the primary and ancillary schemas included in the component registry. The ancillary schemas that are listed in this view are the same schemas that would be included in the OTHER_SCHEMAS column of the DBA_REGISTRY view.

Column	Datatype	NULL	Description
NAMESPACE	VARCHAR2(30)		Component namespace
COMP_ID	VARCHAR2(30)		Component identifier
SCHEMA	VARCHAR2(128)		User that contains the objects for the component

See Also:

- "DBA_REGISTRY"
- "USER_REGISTRY"

6.323 DBA_REGISTRY_SQLPATCH

DBA_REGISTRY_SQLPATCH contains information about the SQL patches that have been installed in the database.

A SQL patch is a patch that contains SQL scripts which need to be run after OPatch completes. DBA_REGISTRY_SQLPATCH is updated by the datapatch utility. Each row contains information about an installation attempt (apply or roll back) for a given patch.

Column	Datatype	NULL	Description
INSTALL_ID	NUMBER	NOT NULL	Unique numeric identifier for this datapatch session. All patches installed in the same invocation of datapatch will have the same value for INSTALL_ID.
PATCH_ID	NUMBER	NOT NULL	ID associated with the patch
PATCH_UID	NUMBER	NOT NULL	UPI (Universal Patch ID) associated with the patch

Column	Datatype	NULL	Description
PATCH_TYPE	VARCHAR2(10)	NOT NULL	Type of the patch. Possible values: • INTERIM: Interim patch • RU: Release Update • RUI: Release Update Increment • RUR: Release Update Revision • CU: Cumulative Update
ACTION	VARCHAR2(15)	NOT NULL	APPLY or ROLLBACK
STATUS	VARCHAR2(25)	NOT NULL	Possible values: • SUCCESS: Patch application has completed with no errors • WITH ERRORS: Patch application finished with errors
ACTION_TIME	TIMESTAMP(6)	NOT NULL	Timestamp when the install was performed
DESCRIPTION	VARCHAR2(100)		Description of this patch from OPatch metadata
LOGFILE	VARCHAR2(500)	NOT NULL	Location of the logfile for this apply or rollback attempt
RU_LOGFILE	VARCHAR2(500)		Logfile location for RU specific commands
FLAGS	VARCHAR2(10)		One or more of the following: • J: Patch is a JVM patch • M: Patch installation was merged with another patch • N: Patch requires normal mode • R: Patch installation has been retried • U: Patch requires upgrade mode
PATCH_DESCRIPTOR	XMLTYPE	NOT NULL	Contents of the XML descriptor for the patch
PATCH_DIRECTORY	BLOB		Contents of the patch directory under ORACLE_HOME/sqlpatch
SOURCE_VERSION	VARCHAR2(15)		5 digit version (for example, 18.3.2.0.0) for the version on which the patch was applied
SOURCE_BUILD_DESCRIPTION	VARCHAR2(80)		Build description (for example, Release_Update or Release_Update_Revision) for the version on which the patch was applied
SOURCE_BUILD_TIMESTAMP	TIMESTAMP(6)		Build timestamp for the version on which the patch was applied
TARGET_VERSION	VARCHAR2(15)		5 digit version (for example, 18.4.0.0.0) for the version to be installed
TARGET_BUILD_DESCRIPTION	VARCHAR2(80)		Build description (for example, Release_Update or Release_Update_Revision) for the version to be installed
TARGET_BUILD_TIMESTAMP	TIMESTAMP(6)		Build timestamp for the version to be installed

See Also:

- *Oracle OPatch User's Guide for Windows and UNIX* for more information about OPatch and related patching utilities
- My Oracle Support note 1585822.1 "Datapatch: Database 12c Post Patch SQL Automation" at the following URL for more information about datapatch:

  https://support.oracle.com/rs?type=doc&id=1585822.1

6.324 DBA_REPAIR_TABLE

DBA_REPAIR_TABLE describes any corruptions found by the DBMS_REPAIR.CHECK_OBJECT procedure.

This information is used by the DBMS_REPAIR.FIX_CORRUPT_BLOCKS procedure on execution. To create this view, first run the DBMS_REPAIR.ADMIN_TABLES procedure. To populate the resulting repair table for an object, run the DBMS_REPAIR.CHECK_OBJECT procedure on the object.

Note:

The table created by the DBMS_REPAIR.ADMIN_TABLES procedure is called REPAIR TABLE by default. If you specify a different name, this view will have the name you specify, preceded by "DBA_REPAIR_".

Column	Datatype	NULL	Description
OBJECT_ID	NUMBER	NOT NULL	Dictionary object number of the object with the corruption
TABLESPACE_ID	NUMBER	NOT NULL	Tablespace number of the corrupt object
RELATIVE_FILE_ID	NUMBER)	NOT NULL	Relative file number of the corrupt object
BLOCK_ID	NUMBER	NOT NULL	Block number of the corruption
CORRUPT_TYPE	NUMBER	NOT NULL	Type of corruption encountered
SCHEMA_NAME	VARCHAR2(128)	NOT NULL	Schema of the corrupt object
OBJECT_NAME	VARCHAR2(128)	NOT NULL	Name of the corrupt object
BASEOBJECT_NAME	VARCHAR2(128)	NULL	If the object is an index, the name of its base table
PARTITION_NAME	VARCHAR2(128)	NULL	Partition or subpartition name, if applicable
CORRUPT_DESCRIPTION	VARCHAR2(2000)	NULL	Description of corruption
REPAIR_DESCRIPTION	VARCHAR2(200)	NULL	Description of repair action
MARKED_CORRUPT	VARCHAR2(10)	NOT NULL	Whether the block is marked corrupt (TRUE | FALSE)
CHECK_TIMESTAMP	DATE	NOT NULL	Date and time when this row was inserted into the repair table
FIX_TIMESTAMP	DATE	NULL	Date and time when the block was modified by the FIX_CORRUPT_BLOCKS procedure, if applicable
REFORMAT_TIMESTAMP	DATE	NULL	Reserved for future use

> **See Also:**
>
> *Oracle Database PL/SQL Packages and Types Reference* for more information about the `DBMS_REPAIR` package

6.325 DBA_REPL_DBNAME_MAPPING

`DBA_REPL_DBNAME_MAPPING` provides details about the database name mapping in replication. Its columns are the same as those in `ALL_REPL_DBNAME_MAPPING`.

> **See Also:**
>
> "ALL_REPL_DBNAME_MAPPING"

6.326 DBA_REPLICATION_PROCESS_EVENTS

`DBA_REPLICATION_PROCESS_EVENTS` provides information about the replication processes events in the database. Its columns are the same as those in `ALL_REPLICATION_PROCESS_EVENTS`.

> **See Also:**
>
> "ALL_REPLICATION_PROCESS_EVENTS"

6.327 DBA_RESOURCE_INCARNATIONS

`DBA_RESOURCE_INCARNATIONS` lists all resource incarnations that are running or eligible for HA status notification.

Column	Datatype	NULL	Description
RESOURCE_TYPE	VARCHAR2(30)	NOT NULL	Type of resource
RESOURCE_NAME	VARCHAR2(256)		Name of resource
DB_UNIQUE_NAME	VARCHAR2(30)	NOT NULL	Database unique name
DB_DOMAIN	VARCHAR2(128)	NOT NULL	Database domain
INSTANCE_NAME	VARCHAR2(30)	NOT NULL	Name of instance at which resource is located
HOST_NAME	VARCHAR2(512)		Name of host at which resource is located
STARTUP_TIME	TIMESTAMP(9) WITH TIME ZONE		Resource startup date and time

6.328 DBA_RESUMABLE

DBA_RESUMABLE displays all resumable statements executed in the system.

Related View

USER_RESUMABLE displays the resumable statements executed by the current user. This view does not display the USER_ID column.

Column	Datatype	NULL	Description
USER_ID	NUMBER		User ID Number of the Resumable Statement Owner
SESSION_ID	NUMBER		Session Identifier of the Resumable Statement
INSTANCE_ID	NUMBER		Instance Number of the Resumable Statement
COORD_INSTANCE_ID	NUMBER		Instance Number on which the Parallel Coordinator is Running
COORD_SESSION_ID	NUMBER		Session Identifier of the Parallel Coordinator
STATUS	VARCHAR2(9)		Status of the resumable statement: • RUNNING • SUSPENDED • TIMEOUT • ERROR • ABORTED
TIMEOUT	NUMBER		Timeout of the resumable statement
START_TIME	VARCHAR2(20)		Start time of the resumable statement
SUSPEND_TIME	VARCHAR2(20)		Last time the resumable statement was suspended (initialized to NULL)
RESUME_TIME	VARCHAR2(20)		Last time the suspended resumable statement was resumed (initialized to NULL)
NAME	VARCHAR2(4000)		Name given in the resumable clause of the resumable statement
SQL_TEXT	VARCHAR2(1000)		Resumable statement, selected from the V$SQL view
ERROR_NUMBER	NUMBER		Error code of the last correctable error. When STATUS is set to RUNNING, its value will be 0.
ERROR_PARAMETER1	VARCHAR2(80)		First parameter for the error message (NULL if no error)
ERROR_PARAMETER2	VARCHAR2(80)		Second parameter for the error message (NULL if no error)
ERROR_PARAMETER3	VARCHAR2(80)		Third parameter for the error message (NULL if no error)
ERROR_PARAMETER4	VARCHAR2(80)		Forth parameter for the error message (NULL if no error)
ERROR_PARAMETER5	VARCHAR2(80)		Fifth parameter for the error message (NULL if no error)
ERROR_MSG	VARCHAR2(4000)		Error message corresponding to ERROR_NUMBER. It will be NULL when ERROR_NUMBER is 0.

> **See Also:**
>
> "USER_RESUMABLE"

6.329 DBA_REWRITE_EQUIVALENCES

DBA_REWRITE_EQUIVALENCES describes all rewrite equivalences in the database. Its columns are the same as those in ALL_REWRITE_EQUIVALENCES.

> **See Also:**
>
> "ALL_REWRITE_EQUIVALENCES"

6.330 DBA_RGROUP

DBA_RGROUP displays all refresh groups.

Column	Datatype	NULL	Description
REFGROUP	NUMBER		Internal identifier of the refresh group
OWNER	VARCHAR2(128)	NOT NULL	Owner of the object in the refresh group
NAME	VARCHAR2(128)	NOT NULL	Name of the object in the refresh group
IMPLICIT_DESTROY	VARCHAR2(1)		Indicates whether the refresh group is destroyed when its last item is removed (Y) or not (N)
PUSH_DEFERRED_RPC	VARCHAR2(1)		Indicates whether changes are pushed from the snapshot to the master before refresh (Y) or not (N)
REFRESH_AFTER_ERRORS	VARCHAR2(1)		Indicates whether to proceed with refresh despite errors when pushing deferred RPCs (Y) or not (N)
ROLLBACK_SEG	VARCHAR2(128)		Name of the rollback segment to use while refreshing
JOB	NUMBER	NOT NULL	Identifier of the job used to refresh the group automatically
PURGE_OPTION	NUMBER(38)		Method for purging the transaction queue after each push. 1 indicates quick purge option; 2 indicates precise purge option
PARALLELISM	NUMBER(38)		Level of parallelism for transaction propagation
HEAP_SIZE	NUMBER(38)		Size of the heap
JOB_NAME	VARCHAR2(128)		The name of the job used to automatically refresh the group

6.331 DBA_ROLE_PRIVS

DBA_ROLE_PRIVS describes the roles granted to all users and roles in the database.

Related View

USER_ROLE_PRIVS describes the roles granted to the current user.

Column	Datatype	NULL	Description
GRANTEE	VARCHAR2(128)		Name of the user or role receiving the grant
GRANTED_ROLE	VARCHAR2(128)		Granted role name
ADMIN_OPTION	VARCHAR2(3)		Indicates whether the grant was with the ADMIN OPTION (YES) or not (NO)
DELEGATE_OPTION	VARCHAR2(3)		Indicates whether the grant was with the DELEGATE OPTION (YES) or not (NO)
DEFAULT_ROLE	VARCHAR2(3)		Indicates whether the role is designated as a DEFAULT ROLE for the user (YES) or not (NO)
COMMON	VARCHAR2(3)		Indicates how the grant was made. Possible values: • YES if the role was granted commonly (CONTAINER=ALL was used) • NO if the role was granted locally (CONTAINER=ALL was not used)
INHERITED	VARCHAR2(3)		Indicates whether the role grant was inherited from another container (YES) or not (NO)

See Also:

"USER_ROLE_PRIVS"

6.332 DBA_ROLES

DBA_ROLES describes all roles in the database.

Column	Datatype	NULL	Description
ROLE	VARCHAR2(128)	NOT NULL	Name of the role
ROLE_ID	NUMBER	NOT NULL	ID number of the role
PASSWORD_REQUIRED	VARCHAR2(8)		This column is deprecated in favor of the AUTHENTICATION_TYPE column

Column	Datatype	NULL	Description
AUTHENTICATION_TYPE	VARCHAR2(11)		Indicates the authentication mechanism for the role: • NONE - CREATE ROLE *role1*; • EXTERNAL - CREATE ROLE *role2* IDENTIFIED EXTERNALLY; • GLOBAL - CREATE ROLE *role3* IDENTIFIED GLOBALLY; • APPLICATION - CREATE ROLE *role4* IDENTIFIED USING *schema.package*; • PASSWORD - CREATE ROLE *role5* IDENTIFIED BY *role5*;
COMMON	VARCHAR2(3)		Indicates whether a given role is common. Possible values: • YES if the role is common • NO if the role is local (not common)
ORACLE_MAINTAINED	VARCHAR2(1)		Denotes whether the role was created, and is maintained, by Oracle-supplied scripts (such as catalog.sql or catproc.sql). A role for which this column has the value Y must not be changed in any way except by running an Oracle-supplied script.
INHERITED	VARCHAR2(3)		Indicates whether the role was inherited from another container (YES) or not (NO)
IMPLICIT	VARCHAR2(3)		Indicates whether the role is a common role created by an implicit application (YES) or not (NO)
EXTERNAL_NAME	VARCHAR2(4000)		For a global role, the external name refers to the DN of a group from a directory service that is mapped to the global role. This is not applicable to a local role.

6.333 DBA_ROLLBACK_SEGS

DBA_ROLLBACK_SEGS describes rollback segments.

Column	Datatype	NULL	Description
SEGMENT_NAME	VARCHAR2(30)	NOT NULL	Name of the rollback segment
OWNER	VARCHAR2(6)		Owner of the rollback segment: • PUBLIC • SYS
TABLESPACE_NAME	VARCHAR2(30)	NOT NULL	Name of the tablespace containing the rollback segment
SEGMENT_ID	NUMBER	NOT NULL	ID number of the rollback segment
FILE_ID	NUMBER	NOT NULL	Absolute file number of the data file containing the segment header
BLOCK_ID	NUMBER	NOT NULL	ID number of the block containing the segment header
INITIAL_EXTENT	NUMBER		Initial extent size in bytes
NEXT_EXTENT	NUMBER		Secondary extent size in bytes
MIN_EXTENTS	NUMBER	NOT NULL	Minimum number of extents
MAX_EXTENTS	NUMBER	NOT NULL	Maximum number of extent
PCT_INCREASE	NUMBER		Percent increase for extent size

Column	Datatype	NULL	Description
STATUS	VARCHAR2(16)		Rollback segment status: • OFFLINE • ONLINE • NEEDS RECOVERY • PARTLY AVAILABLE • UNDEFINED
INSTANCE_NUM	VARCHAR2(40)		Rollback segment owning Oracle Real Application Clusters instance number
RELATIVE_FNO	NUMBER	NOT NULL	Relative file number of the segment header

6.334 DBA_ROLLING_DATABASES

DBA_ROLLING_DATABASES lists all the databases eligible for configuration with rolling operations.

Column	Datatype	NULL	Description
RDBID	NUMBER		Rolling operation database identifier
DBID	NUMBER		Oracle database identifier
DBUN	VARCHAR2(128)		Database unique name
ROLE	VARCHAR2(8)		Database role
OPEN_MODE	VARCHAR2(15)		Open mode information
PARTICIPANT	VARCHAR2(3)		Indicates whether the database is participating in the rolling operation (YES) or not (NO)
VERSION	VARCHAR2(128)		RDBMS version number
ENGINE_STATUS	VARCHAR2(14)		Running status of the MRP-recovery or LSP-apply process
RAC	VARCHAR2(3)		Indicates whether the database is an Oracle Real Application Clusters (Oracle RAC) database
UPDATE_PROGRESS	VARCHAR2(11)		Upgrade status of the system catalog
PROD_RSCN	VARCHAR(40)		Resetlogs SCN at which redo is currently being produced
PROD_RID	VARCHAR(40)		Resetlogs ID at which redo is currently being produced
PROD_SCN	VARCHAR(40)		Last SCN at which redo was produced
REDO_SOURCE	VARCHAR2(128)		Database unique name of the producer of redo being consumed
CONS_RSCN	VARCHAR(40)		Resetlogs SCN at which redo is currently being consumed
CONS_RID	VARCHAR(40)		Resetlogs ID at which redo is currently being consumed
CONS_SCN	VARCHAR(40)		Last SCN at which redo was consumed
UPDATE_TIME	TIMESTAMP(6)		Time of the last record update

> **See Also:**
>
> *Oracle Data Guard Concepts and Administration* for more information about rolling operations.

6.335 DBA_ROLLING_EVENTS

DBA_ROLLING_EVENTS lists all the events reported from the DBMS_ROLLING PL/SQL package.

Column	Datatype	NULL	Description
EVENTID	NUMBER		Event identifier which identifies event order
EVENT_TIME	TIMESTAMP(6)		Time associated with the event
TYPE	VARCHAR2(7)		Type of event: INFO, NOTICE, WARNING, or ERROR
MESSAGE	VARCHAR2(256)		Text describing the event details
STATUS	NUMBER		Status code associated with an event
INSTID	NUMBER		Instruction ID associated with an event
REVISION	NUMBER		Plan revision number associated with an event

> **See Also:**
>
> - *Oracle Data Guard Concepts and Administration* for more information about rolling operations.
> - *Oracle Database PL/SQL Packages and Types Reference* for more information about the DBMS_ROLLING package

6.336 DBA_ROLLING_PARAMETERS

DBA_ROLLING_PARAMETERS lists the available parameters of the DBMS_ROLLING PL/SQL package.

Column	Datatype	NULL	Description
SCOPE	VARCHAR2(128)		Database unique name associated with a parameter
TYPE	VARCHAR2(7)		Type of parameter
NAME	VARCHAR2(32)		Name of the parameter
DESCRIPTION	VARCHAR2(256)		Description of the parameter
CURVAL	VARCHAR2(256)		Current value of the parameter
LSTVAL	VARCHAR2(256)		Prior value of the parameter
DEFVAL	VARCHAR2(256)		Default value of the parameter
MINVAL	NUMBER		Minimum value of the parameter
MAXVAL	NUMBER		Maximum value of the parameter

See Also:

- *Oracle Data Guard Concepts and Administration* for more information about rolling operations.
- *Oracle Database PL/SQL Packages and Types Reference* for more information about the `DBMS_ROLLING` package

6.337 DBA_ROLLING_PLAN

`DBA_ROLLING_PLAN` displays the instructions which constitute the active upgrade plan.

Each row in `DBA_ROLLING_PLAN` identifies a specific instruction scheduled to execute at a specific database. Instructions are created as a result of successful calls to the `DBMS_ROLLING.BUILD_PLAN` procedure.

During execution, groups of instructions are scheduled in batches to execute at remote databases. Groups of instructions are guaranteed to complete in `BATCHID` order.

Column	Datatype	NULL	Description
REVISION	NUMBER		Plan revision number associated with an instruction
BATCHID	NUMBER		Identifier for a batch of instructions which are requested together
INSTID	NUMBER		Identifier for a single instruction
SOURCE	VARCHAR2(128)		Database unique name where an instruction executes
TARGET	VARCHAR2(128)		The site where a given instruction will execute
PHASE	VARCHAR2(14)		rolling operation phase in which an instruction executes
STATUS	VARCHAR2(7)		Scheduling status of the instruction
PROGRESS	VARCHAR2(10)		Execution progress of the instruction
DESCRIPTION	VARCHAR2(256)		Description of the instruction
EXEC_STATUS	NUMBER		Status code returned from instruction execution
EXEC_INFO	VARCHAR2(256)		Supplemental information obtained during instruction execution
EXEC_TIME	TIMESTAMP(6)		Time of instruction execution
FINISH_TIME	TIMESTAMP(6)		Time of instruction completion

See Also:

- *Oracle Data Guard Concepts and Administration* for more information about rolling operations.
- *Oracle Database PL/SQL Packages and Types Reference* for more information about the `DBMS_ROLLING` package

6.338 DBA_ROLLING_STATISTICS

DBA_ROLLING_STATISTICS provides a list of rolling operation statistics.

Column	Datatype	NULL	Description
NAME	VARCHAR2(256)		Name of the statistic. Possible values: • DBMS_ROLLING execution time • logical to primary switchover finish time • primary services offline for • primary to logical switchover start time • rolling upgrade finish time • rolling upgrade start time • time for former primary to fully recover upgrade redo • time for former primary to start upgrade redo recovery • total time former primary in physical role • transient logical creation finish time • transient logical creation start time • transient logical protection finish time • transient logical protection start time • user upgrade time
VALUE	VARCHAR2(256)		Value of the statistic
UPDATE_TIME	TIMESTAMP(6)		Time of last update

See Also:

Oracle Data Guard Concepts and Administration for more information about rolling operations.

6.339 DBA_ROLLING_STATUS

DBA_ROLLING_STATUS displays the overall status of the rolling operation.

Column	Datatype	NULL	Description
REVISION	NUMBER		Revision number of the current upgrade plan
STATUS	VARCHAR2(12)		Readiness of the facility to begin or resume the rolling operation
PHASE	VARCHAR2(14)		Current phase of the plan
NEXT_INSTRUCTION	NUMBER		Instruction ID of the next pending instruction
REMAINING_INSTRUCTIONS	NUMBER		Number of remaining instructions to execute in the plan
COORDINATOR_INSTANCE	NUMBER		Instance number from which the rolling operation is being coordinated
COORDINATOR_PID	NUMBER		Process PID in which the rolling operation is being coordinated

Column	Datatype	NULL	Description
ORIGINAL_PRIMARY	VARCHAR2(128)		Database unique name of the original primary
FUTURE_PRIMARY	VARCHAR2(128)		Database unique name of the future primary
TOTAL_DATABASES	NUMBER		Number of total databases eligible to participate in the rolling operation
PARTICIPATING_DATABASES	NUMBER		Number of databases configured to participate in the rolling operation
INIT_TIME	TIMESTAMP(6)		Time of the last call to DBMS_ROLLING.INIT_PLAN
BUILD_TIME	TIMESTAMP(6)		Time of the last call to DBMS_ROLLING.BUILD
START_TIME	TIMESTAMP(6)		Time of the last call to DBMS_ROLLING.START_UPGRADE
SWITCH_TIME	TIMESTAMP(6)		Time of the last call to DBMS_ROLLING.SWITCHOVER
FINISH_TIME	TIMESTAMP(6)		Time of the last call to DBMS_ROLLING.FINISH

See Also:

- *Oracle Data Guard Concepts and Administration* for more information about rolling operations.
- *Oracle Database PL/SQL Packages and Types Reference* for more information about the DBMS_ROLLING package

6.340 DBA_ROLLING_SUPPORT_MODE

DBA_ROLLING_SUPPORT_MODE displays information about whether tables in the database are supported for rolling upgrades performed using the DBMS_ROLLING package.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)		Table owner
TABLE_NAME	VARCHAR2(128)		Table name
SUPPORT_MODE	VARCHAR2(11)		Indicates whether the table is supported Possible values: • SUPPORTED - The table is fully supported • INTERNAL - The table is not supported because it contains data that should not be replicated. Such tables include mapping tables for index-organized tables, storage tables for nested tables, materialized view logs, secondary objects associated with domain indexes, and temporary tables. • UNSUPPORTED - The table is not supported because it contains a data type that is not supported or uses a feature that is not supported

Column	Datatype	NULL	Description
EXPLANATION	VARCHAR2(4000)		Reason why the table is not fully supported This column is populated only when both of the following conditions are met: • The value of the COMPATIBLE initialization parameter is 20.0 or higher • The value of the SUPPORT_MODE column is INTERNAL or UNSUPPORTED

6.341 DBA_ROLLING_UNSUPPORTED

DBA_ROLLING_UNSUPPORTED displays the schemas, tables, and columns in those tables that contain unsupported data types for a rolling upgrade operation for a logical standby database using the DBMS_ROLLING PL/SQL package.

Use this view before you perform a rolling upgrade using DBMS_ROLLING to determine what is unsupported.

The data pertains to the container in which the view is queried.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)		Schema name of the unsupported column
TABLE_NAME	VARCHAR2(128)		Name of the table that the unsupported column belongs to
COLUMN_NAME	VARCHAR2(128)		Name of the unsupported column
ATTRIBUTES	VARCHAR2(4000)		When the value of the COMPATIBLE initialization parameter is 20.0 or higher, this column displays the reason the table is unsupported. Otherwise, this column displays the reason the table is unsupported only if the reason is not a data type issue.
DATA_TYPE	VARCHAR2(106)		Data type of the unsupported column

Note:

When the value of the COMPATIBLE initialization parameter is 20.0 or higher, this view displays all columns in unsupported tables. Otherwise, this view displays only the unsupported columns in unsupported tables.

Note:

A rolling upgrade using DBMS_ROLLING supports more object types than a manual rolling upgrade using transient logical standby databases

See Also:

- "DBA_LOGSTDBY_UNSUPPORTED" for more information about determining unsupported data types for a manual rolling upgrade operation using transient logical standby databases
- *Oracle Data Guard Concepts and Administration* for more information about rolling operations
- *Oracle Data Guard Concepts and Administration* for more information about unsupported tables for rolling upgrade operations
- *Oracle Database PL/SQL Packages and Types Reference* for more information about the DBMS_ROLLING package

6.342 DBA_RSRC_CATEGORIES

DBA_RSRC_CATEGORIES displays all resource consumer group categories.

Column	Datatype	NULL	Description
NAME	VARCHAR2(128)		Name of the consumer group category
COMMENTS	VARCHAR2(2000)		Text comment on the consumer group category
STATUS	VARCHAR2(128)		Indicates whether the consumer group category is part of the pending area (PENDING) or not (NULL)
MANDATORY	VARCHAR2(3)		Indicates whether the consumer group category is mandatory (YES) or not (NO)

6.343 DBA_RSRC_CONSUMER_GROUP_PRIVS

DBA_RSRC_CONSUMER_GROUP_PRIVS displays information about all resource consumer groups and the users and roles assigned to them.

The grant referred to in this view and the related view is the grant of the SWITCH_CONSUMER_GROUP object privilege, which is granted using the DBMS_RESOURCE_MANAGER_PRIVS package. This privilege is not granted through the GRANT SQL statement.

Related View

USER_RSRC_CONSUMER_GROUP_PRIVS displays information about the resource consumer groups to which the current user is assigned. This view does not display the GRANTEE column.

Column	Datatype	NULL	Description
GRANTEE	VARCHAR2(128)	NOT NULL	User or role receiving the grant
GRANTED_GROUP	VARCHAR2(128)		Granted consumer group name
GRANT_OPTION	VARCHAR2(3)		Indicates whether the grant was with the GRANT option (YES) or not (NO)
INITIAL_GROUP	VARCHAR2(3)		Indicates whether the consumer group is designated as the default for this user or role (YES) or not (NO)

> **See Also:**
>
> - "USER_RSRC_CONSUMER_GROUP_PRIVS"
> - *Oracle Database PL/SQL Packages and Types Reference* for more information about the DBMS_RESOURCE_MANAGER_PRIVS package

6.344 DBA_RSRC_CONSUMER_GROUPS

DBA_RSRC_CONSUMER_GROUPS displays information about all resource consumer groups in the database.

Column	Datatype	NULL	Description
CONSUMER_GROUP_ID	NUMBER	NOT NULL	ID of the consumer group
CONSUMER_GROUP	VARCHAR2(128)		Name of the consumer group
CPU_METHOD	VARCHAR2(128)		CPU resource allocation method for the consumer group
MGMT_METHOD	VARCHAR2(128)		Resource allocation method for the consumer group
INTERNAL_USE	VARCHAR2(3)		Indicates whether the consumer group is for internal use only (YES) or not (NO)
COMMENTS	VARCHAR2(2000)		Text comment on the consumer group
CATEGORY	VARCHAR2(128)		Category of the consumer group
STATUS	VARCHAR2(128)		Indicates whether the consumer group is part of the pending area (PENDING) or not (NULL)
MANDATORY	VARCHAR2(3)		Indicates whether the consumer group is mandatory (YES) or not (NO)

6.345 DBA_RSRC_GROUP_MAPPINGS

DBA_RSRC_GROUP_MAPPINGS displays the mapping between session attributes and consumer groups in the database.

Column	Datatype	NULL	Description
ATTRIBUTE	VARCHAR2(128)		Session attribute to match
VALUE	VARCHAR2(128)		Attribute value
CONSUMER_GROUP	VARCHAR2(128)		Target consumer group name
STATUS	VARCHAR2(128)		Indicates whether the consumer group is part of the pending area (PENDING) or not (NULL)

6.346 DBA_RSRC_IO_CALIBRATE

DBA_RSRC_IO_CALIBRATE displays I/O calibration results for the latest calibration run.

Column	Datatype	NULL	Description
START_TIME	TIMESTAMP(6)		Start time of the most recent I/O calibration
END_TIME	TIMESTAMP(6)		End time of the most recent I/O calibration
MAX_IOPS	NUMBER		Maximum number of data block read requests that can be sustained per second
MAX_MBPS	NUMBER		Maximum megabytes per second of maximum-sized read requests that can be sustained
MAX_PMBPS	NUMBER		Maximum megabytes per second of large I/O requests that can be sustained by a single process
LATENCY	NUMBER		Latency for data block read requests
NUM_PHYSICAL_DISKS	NUMBER		Number of physical disks in the storage subsystem (as specified by the user)
ADDITIONAL_INFO	VARCHAR2(1024)		Additional information about the most recent calibration run

6.347 DBA_RSRC_MANAGER_SYSTEM_PRIVS

DBA_RSRC_MANAGER_SYSTEM_PRIVS displays information about all the users and roles that have been granted the ADMINISTER_RESOURCE_MANAGER system privilege, which is granted using the DBMS_RESOURCE_MANAGER_PRIVS package.

This privilege is not granted through the GRANT SQL statement.

Related View

USER_RSRC_MANAGER_SYSTEM_PRIVS displays information about the users who are granted system privileges for the DBMS_RESOURCE_MANAGER package. This view does not display the GRANTEE column.

Column	Datatype	NULL	Description
GRANTEE	VARCHAR2(128)	NOT NULL	User or role receiving the grant
PRIVILEGE	VARCHAR2(40)	NOT NULL	Name of the system privilege
ADMIN_OPTION	VARCHAR2(3)		Indicates whether the grant was with the ADMIN option (YES) or not (NO)

See Also:

- "USER_RSRC_MANAGER_SYSTEM_PRIVS"
- *Oracle Database PL/SQL Packages and Types Reference* for more information about the DBMS_RESOURCE_MANAGER package

6.348 DBA_RSRC_MAPPING_PRIORITY

DBA_RSRC_MAPPING_PRIORITY displays information about all consumer group mapping attribute priorities.

Column	Datatype	NULL	Description
ATTRIBUTE	VARCHAR2(128)		Session attribute
PRIORITY	NUMBER		Priority (1 is the highest)
STATUS	VARCHAR2(128)		Indicates whether the consumer group is part of the pending area (PENDING) or not (NULL)

6.349 DBA_RSRC_PLAN_DIRECTIVES

DBA_RSRC_PLAN_DIRECTIVES displays information about all resource plan directives in the database.

Column	Datatype	NULL	Description
PLAN	VARCHAR2(128)		Name of the plan to which the directive belongs
GROUP_OR_SUBPLAN	VARCHAR2(128)		Name of the consumer group or subplan referred to
TYPE	VARCHAR2(14)		Indicates whether GROUP_OR_SUBPLAN refers to a consumer group (CONSUMER_GROUP) or a plan (PLAN)
CPU_P1	NUMBER		This column is deprecated. Use the MGMT_P1 column instead.
CPU_P2	NUMBER		This column is deprecated. Use the MGMT_P2 column instead.
CPU_P3	NUMBER		This column is deprecated. Use the MGMT_P3 column instead.
CPU_P4	NUMBER		This column is deprecated. Use the MGMT_P4 column instead.
CPU_P5	NUMBER		This column is deprecated. Use the MGMT_P5 column instead.
CPU_P6	NUMBER		This column is deprecated. Use the MGMT_P6 column instead.
CPU_P7	NUMBER		This column is deprecated. Use the MGMT_P7 column instead.
CPU_P8	NUMBER		This column is deprecated. Use the MGMT_P8 column instead.
MGMT_P1	NUMBER		Resource allocation at level 1. For share-based plans, indicates the number of shares.
MGMT_P2	NUMBER		Resource allocation at level 2.
MGMT_P3	NUMBER		Resource allocation at level 3.
MGMT_P4	NUMBER		Resource allocation at level 4.
MGMT_P5	NUMBER		Resource allocation at level 5.
MGMT_P6	NUMBER		Resource allocation at level 6.
MGMT_P7	NUMBER		Resource allocation at level 7.
MGMT_P8	NUMBER		Resource allocation at level 8.
ACTIVE_SESS_POOL_P1	NUMBER		Maximum number of calls this consumer group can run concurrently
QUEUEING_P1	NUMBER		Timeout in seconds for waits in the Active Session Limit queue

Column	Datatype	NULL	Description
PARALLEL_TARGET_PERCENTAGE	NUMBER		This column is deprecated. Use the PARALLEL_SERVER_LIMIT column instead.
PARALLEL_DEGREE_LIMIT_P1	NUMBER		Sessions in this consumer group are limited to this maximum degree of parallelism for all parallel operations
SWITCH_GROUP	VARCHAR2(128)		Group to switch to once the switch time is reached
SWITCH_FOR_CALL	VARCHAR2(5)		Indicates whether to switch back to the initial consumer group once the top call has completed (TRUE) or not (FALSE)
SWITCH_TIME	NUMBER		Amount of run time (in seconds) within a group before the session is automatically switched. As with other switch directives, if SWITCH_FOR_CALL is TRUE, the run time is accumulated from the start of a call. Otherwise, the run time is accumulated for the length of the session.
SWITCH_IO_MEGABYTES	NUMBER		The maximum megabytes of I/O within a group that will trigger the action specified by SWITCH_GROUP. As with other switch directives, if SWITCH_FOR_CALL is TRUE, the maximum megabytes of I/O is accumulated from the start of a call. Otherwise, the maximum megabytes of I/O is accumulated for the length of the session.
SWITCH_IO_REQS	NUMBER		The maximum I/O requests within a group that will trigger the action specified by SWITCH_GROUP. As with other switch directives, if SWITCH_FOR_CALL is TRUE, the maximum I/O requests is accumulated from the start of a call. Otherwise, the maximum I/O requests is accumulated for the length of the session.
SWITCH_ESTIMATE	VARCHAR2(5)		Indicates whether estimated execution time should be used for switch criteria (TRUE) or not (FALSE)
MAX_EST_EXEC_TIME	NUMBER		Maximum estimated execution time
UNDO_POOL	NUMBER		Undo pool size for the consumer group
MAX_IDLE_TIME	NUMBER		Maximum idle time for the session
MAX_IDLE_BLOCKER_TIME	NUMBER		Maximum idle time for the session when blocking other sessions
MAX_UTILIZATION_LIMIT	NUMBER		This column is deprecated. Use UTILIZATION_LIMIT instead.
PARALLEL_QUEUE_TIMEOUT	NUMBER		Time (in seconds) that a query can remain in the parallel statement queue for the consumer group before it is removed from the queue. The PQ_TIMEOUT_ACTION directive in a Resource Manager plan can be used in conjunction with the PARALLEL_QUEUE_TIMEOUT directive to either cancel or run the removed query. If the PQ_TIMEOUT_ACTION directive is not specified, the default behavior is to cancel the query with ORA-07454.
SWITCH_TIME_IN_CALL	NUMBER		This column is deprecated. Specify the time in the SWITCH_TIME directive and set SWITCH_FOR_CALL to TRUE.

Column	Datatype	NULL	Description
SWITCH_IO_LOGICAL	NUMBER		The number of logical I/Os that will trigger the action specified by SWITCH_GROUP. As with other switch directives, if SWITCH_FOR_CALL is TRUE, the number of logical I/Os is accumulated from the start of a call. Otherwise, the number of logical IOs is accumulated for the length of the session.
SWITCH_ELAPSED_TIME	NUMBER		The elapsed time that will trigger the action specified by SWITCH_GROUP. As with other switch directives, if SWITCH_FOR_CALL is TRUE, the elapsed time is accumulated from the start of a call. Otherwise, the elapsed time is accumulated for the length of the session.
PARALLEL_SERVER_LIMIT	NUMBER		Maximum percentage of the parallel target used before queuing subsequent parallel queries
UTILIZATION_LIMIT	NUMBER		Maximum resource utilization allowed, expressed in percentage
PARALLEL_STMT_CRITICAL	VARCHAR2(12)		Indicates whether parallel statements from this consumer group are eligible for queuing in the parallel statement queue: • BYPASS QUEUE - Parallel statements in this consumer group are critical and should never be queued. • QUEUE - All parallel statements in this consumer group, irrespective of the PARALLEL_DEGREE_POLICY initialization parameter value, are eligible for being queued. • FALSE - Certain parallel statements are eligible for queuing, depending on the PARALLEL_DEGREE_POLICY initialization parameter value. This is the default.
SESSION_PGA_LIMIT	NUMBER		The maximum amount of untunable PGA in MB that sessions in this consumer group can allocate before being terminated
PQ_TIMEOUT_ACTION	VARCHAR2(6)		Indicates the action to be taken on a parallel query in the parallel queue when its queue time exceeds the limit set by the Resource Manager plan's PARALLEL_QUEUE_TIMEOUT directive: • CANCEL - The statement terminates with error ORA-07454 • RUN - The statement runs immediately, and may get downgraded if parallel servers are unavailable
COMMENTS	VARCHAR2(2000)		Text comment on the plan directive
STATUS	VARCHAR2(128)		Indicates whether the plan directive is part of the pending area (PENDING) or not (NULL). **Note:** PDB resource plans must be single-level, they cannot contain subplans, and they must have 8 or fewer consumer groups. If a resource plan is imported into a PDB and it violates any of these PDB requirements, then the import will automatically convert the resource plan to a compliant version. The original, unmodified resource plan will be stored with a STATUS of LEGACY.

Column	Datatype	NULL	Description
MANDATORY	VARCHAR2(3)		Indicates whether the plan directive is mandatory (YES) or not (NO). Mandatory plans cannot be deleted.

See Also:

- "PARALLEL_DEGREE_POLICY"
- *Oracle Database Administrator's Guide* for information on resource plans in general
- *Oracle Database PL/SQL Packages and Types Reference* for information about specifying Resource Manager directive values using the DBMS_RESOURCE_MANAGER.CREATE_PLAN_DIRECTIVE procedure

6.350 DBA_RSRC_PLANS

DBA_RSRC_PLANS displays information about all resource plans in the database.

For a list of currently active plans, see "V$RSRC_PLAN".

Column	Datatype	NULL	Description
PLAN_ID	NUMBER	NOT NULL	ID number of the resource plan
PLAN	VARCHAR2(128)		Name of the resource plan
NUM_PLAN_DIRECTIVES	NUMBER		Number of plan directives for the plan
CPU_METHOD	VARCHAR2(128)		CPU resource allocation method for the plan
MGMT_METHOD	VARCHAR2(128)		Resource allocation method for the plan
ACTIVE_SESS_POOL_MTH	VARCHAR2(128)		Active session pool resource allocation method for the plan
PARALLEL_DEGREE_LIMIT_MTH	VARCHAR2(128)		Parallel degree limit resource allocation method for the plan
QUEUING_MTH	VARCHAR2(128)		Queuing resource allocation method for the plan
SUB_PLAN	VARCHAR2(3)		Indicates whether the plan is a subplan (YES) or not (NO)
COMMENTS	VARCHAR2(2000)		Text comment on the plan
STATUS	VARCHAR2(128)		Indicates whether the plan is part of the pending area (PENDING) or not (NULL)
MANDATORY	VARCHAR2(3)		Indicates whether the plan is mandatory (YES) or not (NO). Mandatory plans cannot be deleted.

> **See Also:**
>
> - *Oracle Database Administrator's Guide* for information on resource plans in general
> - *Oracle Database PL/SQL Packages and Types Reference* for more information on creating resource plans with the `DBMS_RESOURCE_MANAGER` package

6.351 DBA_RULE_SET_RULES

`DBA_RULE_SET_RULES` describes the rules in all rule sets in the database. Its columns are the same as those in `ALL_RULE_SET_RULES`.

> **See Also:**
>
> "ALL_RULE_SET_RULES"

6.352 DBA_RULE_SETS

`DBA_RULE_SETS` describes all rule sets in the database. Its columns are the same as those in `ALL_RULE_SETS`.

> **See Also:**
>
> "ALL_RULE_SETS"

6.353 DBA_RULES

`DBA_RULES` describes all rules in the database. Its columns are the same as those in `ALL_RULES`.

> **See Also:**
>
> "ALL_RULES"

6.354 DBA_SAGA_BROKERS

DBA_SAGA_BROKERS displays all saga brokers in the database.

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

> **See Also:**
>
> "ALL_SAGA_BROKERS"

6.355 DBA_SAGA_DETAILS

DBA_SAGA_DETAILS displays details for all sagas in the database. Its columns are the same as those in ALL_SAGA_DETAILS.

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

> **See Also:**
>
> "ALL_SAGA_DETAILS"

6.356 DBA_SAGA_ERRORS

DBA_SAGA_ERRORS describes errors generated by all sagas in the database. Its columns are the same as those in ALL_SAGA_ERRORS.

This view displays sagas that were either initiated in the current PDB or joined by participants in the current PDB.

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

> **See Also:**
>
> "ALL_SAGA_ERRORS"

6.357 DBA_SAGA_FINALIZATION

DBA_SAGA_FINALIZATION displays information about pending finalization actions for all sagas in the database. Its columns are the same as those in ALL_SAGA_FINALIZATION.

This view displays one row for each unique reservable table updated as part of a participant transaction for sagas that were either initiated in the current PDB or joined by participants in the current PDB.

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

> **See Also:**
>
> "ALL_SAGA_FINALIZATION"

6.358 DBA_SAGA_PARTICIPANT_SET

DBA_SAGA_PARTICIPANT_SET displays information about saga participants for all sagas in the database. Its columns are the same as those in ALL_SAGA_PARTICIPANT_SET.

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

> **See Also:**
>
> "ALL_SAGA_PARTICIPANT_SET"

6.359 DBA_SAGA_PARTICIPANTS

DBA_SAGA_PARTICIPANTS describes all saga participants in the database. Its columns are the same as those in ALL_SAGA_PARTICIPANTS.

This view displays sagas that were either initiated in the current PDB or joined by participants or coordinators in the current PDB.

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

> **See Also:**
>
> "ALL_SAGA_PARTICIPANTS"

6.360 DBA_SAGA_PENDING

`DBA_SAGA_PENDING` describes all pending sagas in the database. Its columns are the same as those in `ALL_SAGA_PENDING`.

This view displays sagas that were initiated in the current PDB.

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

> **See Also:**
>
> "ALL_SAGA_PENDING"

6.361 DBA_SAGAS

`DBA_SAGAS` describes all active sagas in the database. Its columns are the same as those in `ALL_SAGAS`.

This view displays sagas that were either initiated in the current PDB or joined by participants in the current PDB.

> **Note:**
>
> This view displays active sagas. Completed sagas are displayed in the DBA_HIST_SAGAS view.

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

> **See Also:**
>
> "ALL_SAGAS"

6.362 DBA_SCHEDULER_CHAIN_RULES

DBA_SCHEDULER_CHAIN_RULES displays information about the rules for all chains in the database. Its columns are the same as those in ALL_SCHEDULER_CHAIN_RULES.

> **See Also:**
>
> "ALL_SCHEDULER_CHAIN_RULES"

6.363 DBA_SCHEDULER_CHAIN_STEPS

DBA_SCHEDULER_CHAIN_STEPS displays information about the defined steps of all chains in the database. Its columns are the same as those in ALL_SCHEDULER_CHAIN_STEPS.

> **See Also:**
>
> "ALL_SCHEDULER_CHAIN_STEPS"

6.364 DBA_SCHEDULER_CHAINS

DBA_SCHEDULER_CHAINS displays information about all chains in the database. Its columns are the same as those in ALL_SCHEDULER_CHAINS.

> **See Also:**
>
> "ALL_SCHEDULER_CHAINS"

6.365 DBA_SCHEDULER_CREDENTIALS

DBA_SCHEDULER_CREDENTIALS displays information about all credentials in the database. Its columns are the same as those in ALL_SCHEDULER_CREDENTIALS.

> **Note:**
>
> This view is deprecated in favor of the DBA_CREDENTIALS view. Oracle recommends that you use DBA_CREDENTIALS instead. DBA_SCHEDULER_CREDENTIALS is retained for backward compatibility only.

> **See Also:**
>
> - "DBA_CREDENTIALS"
> - "ALL_SCHEDULER_CREDENTIALS"

6.366 DBA_SCHEDULER_DB_DESTS

DBA_SCHEDULER_DB_DESTS displays information about all destination objects in the database pointing to remote databases. Its columns are the same as those in ALL_SCHEDULER_DB_DESTS.

> **See Also:**
>
> "ALL_SCHEDULER_DB_DESTS"

6.367 DBA_SCHEDULER_DESTS

DBA_SCHEDULER_DESTS displays information about all destination objects for jobs in the database. Its columns are the same as those in ALL_SCHEDULER_DESTS.

> **See Also:**
>
> "ALL_SCHEDULER_DESTS"

6.368 DBA_SCHEDULER_EXTERNAL_DESTS

DBA_SCHEDULER_EXTERNAL_DESTS displays information about all destination objects in the database pointing to remote agents. Its columns are the same as those in ALL_SCHEDULER_EXTERNAL_DESTS.

> **See Also:**
>
> "ALL_SCHEDULER_EXTERNAL_DESTS"

6.369 DBA_SCHEDULER_FILE_WATCHERS

DBA_SCHEDULER_FILE_WATCHERS displays information about all scheduler file watch requests in the database. Its columns are the same as those in ALL_SCHEDULER_FILE_WATCHERS.

> **See Also:**
>
> "ALL_SCHEDULER_FILE_WATCHERS"

6.370 DBA_SCHEDULER_GLOBAL_ATTRIBUTE

DBA_SCHEDULER_GLOBAL_ATTRIBUTE displays the values of all scheduler attributes in the database (for example, DEFAULT_TIMEZONE and CURRENT_OPEN_WINDOW). Its columns are the same as those in ALL_SCHEDULER_GLOBAL_ATTRIBUTE.

> **See Also:**
>
> "ALL_SCHEDULER_GLOBAL_ATTRIBUTE"

6.371 DBA_SCHEDULER_GROUP_MEMBERS

DBA_SCHEDULER_GROUP_MEMBERS displays information about the members of all Scheduler object groups in the database. Its columns are the same as those in ALL_SCHEDULER_GROUP_MEMBERS.

> **See Also:**
>
> "ALL_SCHEDULER_GROUP_MEMBERS"

6.372 DBA_SCHEDULER_GROUPS

DBA_SCHEDULER_GROUPS displays information about all Scheduler object groups in the database. Its columns are the same as those in ALL_SCHEDULER_GROUPS.

> **See Also:**
>
> "ALL_SCHEDULER_GROUPS"

6.373 DBA_SCHEDULER_IN_MEMORY_TRACE

DBA_SCHEDULER_IN_MEMORY_TRACE displays scheduler in-memory trace information generated by all user processes in the database.

Column	Datatype	NULL	Description
USER#	NUMBER		Oracle identifier of the user generating scheduler trace
USERNAME	VARCHAR2(128)		Name of the user generating scheduler trace
PROCESSNAME	VARCHAR2(5)		Name of the process generating scheduler trace
SPID	VARCHAR2(24)		Operating system identifier of the process generating scheduler trace
PADDR	RAW(8)		Address of the process state object generating scheduler trace
TRACE_ST	CLOB		Scheduler trace generated by the process

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

6.374 DBA_SCHEDULER_INCOMPAT_MEMBER

DBA_SCHEDULER_INCOMPAT_MEMBER displays all Scheduler incompatibility resource objects members in the database. Its columns are the same as those in ALL_SCHEDULER_INCOMPAT_MEMBER.

> **See Also:**
>
> "ALL_SCHEDULER_INCOMPAT_MEMBER"

6.375 DBA_SCHEDULER_INCOMPATS

DBA_SCHEDULER_INCOMPATS displays all Scheduler incompatibility resource objects in the database. Its columns are the same as those in ALL_SCHEDULER_INCOMPATS.

> **See Also:**
>
> "ALL_SCHEDULER_INCOMPATS"

6.376 DBA_SCHEDULER_JOB_ARGS

DBA_SCHEDULER_JOB_ARGS displays information about the arguments of all Scheduler jobs in the database. Its columns are the same as those in ALL_SCHEDULER_JOB_ARGS.

> **See Also:**
>
> "ALL_SCHEDULER_JOB_ARGS"

6.377 DBA_SCHEDULER_JOB_CLASSES

DBA_SCHEDULER_JOB_CLASSES displays information about all Scheduler job classes in the database. Its columns are the same as those in ALL_SCHEDULER_JOB_CLASSES.

> **See Also:**
>
> "ALL_SCHEDULER_JOB_CLASSES"

6.378 DBA_SCHEDULER_JOB_DESTS

DBA_SCHEDULER_JOB_DESTS displays information about the state of all jobs in the database at each of their destinations. Its columns are the same as those in ALL_SCHEDULER_JOB_DESTS.

> **See Also:**
>
> "ALL_SCHEDULER_JOB_DESTS"

6.379 DBA_SCHEDULER_JOB_LOG

DBA_SCHEDULER_JOB_LOG displays log information for all Scheduler jobs in the database. Its columns are the same as those in ALL_SCHEDULER_JOB_LOG.

> **See Also:**
>
> "ALL_SCHEDULER_JOB_LOG"

6.380 DBA_SCHEDULER_JOB_ROLES

DBA_SCHEDULER_JOB_ROLES displays information about all Scheduler jobs in the database by database role.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)	NOT NULL	Owner of the Scheduler job
JOB_NAME	VARCHAR2(128)	NOT NULL	Name of the Scheduler job
JOB_SUBNAME	VARCHAR2(128)		Subname of the Scheduler job (for a job running a chain step)
JOB_CREATOR	VARCHAR2(128)		Creator of the Scheduler job
DATABASE_ROLE	VARCHAR2(16)		Name of the database role
PROGRAM_OWNER	VARCHAR2(4000)		Owner of the program associated with the job
PROGRAM_NAME	VARCHAR2(4000)		Name of the program associated with the job
JOB_TYPE	VARCHAR2(16)		Inline job action type: • PLSQL_BLOCK • STORED_PROCEDURE • EXECUTABLE • CHAIN
JOB_ACTION	VARCHAR2(4000)		Inline job action
JOB_CLASS	VARCHAR2(128)		Name of the job class associated with the job
SCHEDULE_OWNER	VARCHAR2(4000)		Owner of the schedule that the job uses (can be a window or a window group)
SCHEDULE_NAME	VARCHAR2(4000)		Name of the schedule that the job uses (can be a window or a window group)
SCHEDULE_TYPE	VARCHAR2(12)		Type of the schedule that the job uses: • IMMEDIATE - Start date and repeat interval are NULL • ONCE - Repeat interval is NULL • PLSQL - PL/SQL expression used as schedule • CALENDAR - Oracle calendaring expression used as schedule • EVENT - Event schedule • NAMED - Named schedule • WINDOW - Window used as schedule • WINDOW_GROUP - Window group used as schedule

Column	Datatype	NULL	Description
START_DATE	TIMESTAMP(6) WITH TIME ZONE		Original scheduled start date of the job (for an inline schedule)
REPEAT_INTERVAL	VARCHAR2(4000)		Inline schedule PL/SQL expression or calendar string
END_DATE	TIMESTAMP(6) WITH TIME ZONE		Date after which the job will no longer run (for an inline schedule)
LAST_START_DATE	TIMESTAMP(6) WITH TIME ZONE		Last date on which the job ran
ENABLED	VARCHAR2(5)		Indicates whether the job is enabled (TRUE) or disabled (FALSE)
STATE	VARCHAR2(15)		Current state of the job: • DISABLED • RETRY SCHEDULED • SCHEDULED • RUNNING • COMPLETED • BROKEN • FAILED • REMOTE • SUCCEEDED • CHAIN_STALLED
COMMENTS	VARCHAR2(4000)		Comments on the job

6.381 DBA_SCHEDULER_JOB_RUN_DETAILS

DBA_SCHEDULER_JOB_RUN_DETAILS displays log run details for all Scheduler jobs in the database. Its columns are the same as those in ALL_SCHEDULER_JOB_RUN_DETAILS.

> **See Also:**
>
> "ALL_SCHEDULER_JOB_RUN_DETAILS"

6.382 DBA_SCHEDULER_JOBS

DBA_SCHEDULER_JOBS displays information about all Scheduler jobs in the database. Its columns are the same as those in ALL_SCHEDULER_JOBS.

> **See Also:**
>
> "ALL_SCHEDULER_JOBS"

6.383 DBA_SCHEDULER_NOTIFICATIONS

DBA_SCHEDULER_NOTIFICATIONS displays information about the E-mail notifications for all jobs in the database. Its columns are the same as those in ALL_SCHEDULER_NOTIFICATIONS.

> **See Also:**
>
> "ALL_SCHEDULER_NOTIFICATIONS"

6.384 DBA_SCHEDULER_PROGRAM_ARGS

DBA_SCHEDULER_PROGRAM_ARGS displays information about the arguments of all Scheduler programs in the database. Its columns are the same as those in ALL_SCHEDULER_PROGRAM_ARGS.

> **See Also:**
>
> "ALL_SCHEDULER_PROGRAM_ARGS"

6.385 DBA_SCHEDULER_PROGRAMS

DBA_SCHEDULER_PROGRAMS displays information about all Scheduler programs in the database. Its columns are the same as those in ALL_SCHEDULER_PROGRAMS.

> **See Also:**
>
> "ALL_SCHEDULER_PROGRAMS"

6.386 DBA_SCHEDULER_REMOTE_DATABASES

DBA_SCHEDULER_REMOTE_DATABASES displays information about all remote databases that have been registered as sources and destinations for remote database jobs. Its columns are the same as those in ALL_SCHEDULER_REMOTE_DATABASES.

> **See Also:**
>
> "ALL_SCHEDULER_REMOTE_DATABASES"

6.387 DBA_SCHEDULER_REMOTE_JOBSTATE

DBA_SCHEDULER_REMOTE_JOBSTATE displays information about the state of all jobs at remote databases. Its columns are the same as those in ALL_SCHEDULER_REMOTE_JOBSTATE.

> **See Also:**
>
> "ALL_SCHEDULER_REMOTE_JOBSTATE"

6.388 DBA_SCHEDULER_RESOURCES

DBA_SCHEDULER_RESOURCES displays all scheduler resource objects in the database. Its columns are the same as those in ALL_SCHEDULER_RESOURCES.

> **See Also:**
>
> "ALL_SCHEDULER_RESOURCES"

6.389 DBA_SCHEDULER_RSC_CONSTRAINTS

DBA_SCHEDULER_RSC_CONSTRAINTS lists all Oracle Scheduler resource constraint members in the database. Its columns are the same as those in ALL_SCHEDULER_RSC_CONSTRAINTS.

> **See Also:**
>
> "ALL_SCHEDULER_RSC_CONSTRAINTS"

6.390 DBA_SCHEDULER_RUNNING_CHAINS

DBA_SCHEDULER_RUNNING_CHAINS displays information about the chain steps of all running chains in the database. Its columns are the same as those in ALL_SCHEDULER_RUNNING_CHAINS.

> **See Also:**
>
> "ALL_SCHEDULER_RUNNING_CHAINS"

6.391 DBA_SCHEDULER_RUNNING_JOBS

DBA_SCHEDULER_RUNNING_JOBS displays information about all running Scheduler jobs in the database. Its columns are the same as those in ALL_SCHEDULER_RUNNING_JOBS.

> **See Also:**
>
> "ALL_SCHEDULER_RUNNING_JOBS"

6.392 DBA_SCHEDULER_SCHEDULES

DBA_SCHEDULER_SCHEDULES displays information about all Scheduler schedules in the database. Its columns are the same as those in ALL_SCHEDULER_SCHEDULES.

> **See Also:**
>
> "ALL_SCHEDULER_SCHEDULES"

6.393 DBA_SCHEDULER_WINDOW_DETAILS

DBA_SCHEDULER_WINDOW_DETAILS displays log details for all Scheduler windows in the database. Its columns are the same as those in ALL_SCHEDULER_WINDOW_DETAILS.

> **See Also:**
>
> "ALL_SCHEDULER_WINDOW_DETAILS"

6.394 DBA_SCHEDULER_WINDOW_GROUPS

DBA_SCHEDULER_WINDOW_GROUPS displays information about all Scheduler window groups in the database. Its columns are the same as those in ALL_SCHEDULER_WINDOW_GROUPS.

> **See Also:**
>
> "ALL_SCHEDULER_WINDOW_GROUPS"

6.395 DBA_SCHEDULER_WINDOW_LOG

DBA_SCHEDULER_WINDOW_LOG displays log information for all Scheduler windows in the database. Its columns are the same as those in ALL_SCHEDULER_WINDOW_LOG.

> **See Also:**
>
> "ALL_SCHEDULER_WINDOW_LOG"

6.396 DBA_SCHEDULER_WINDOWS

DBA_SCHEDULER_WINDOWS displays information about all Scheduler windows in the database. Its columns are the same as those in ALL_SCHEDULER_WINDOWS.

> **See Also:**
>
> "ALL_SCHEDULER_WINDOWS"

6.397 DBA_SCHEDULER_WINGROUP_MEMBERS

DBA_SCHEDULER_WINGROUP_MEMBERS displays the members of all Scheduler window groups in the database. Its columns are the same as those in ALL_SCHEDULER_WINGROUP_MEMBERS.

> **See Also:**
>
> "ALL_SCHEDULER_WINGROUP_MEMBERS"

6.398 DBA_SCHEMA_PRIVS

DBA_SCHEMA_PRIVS describes schema privileges granted to users and roles.

Column	Datatype	NULL	Description
GRANTEE	VARCHAR2(128)		Grantee name, user, or role receiving the grant
PRIVILEGE	VARCHAR2(40)		Schema privilege
SCHEMA	VARCHAR2(128)		Schema on which the privilege was granted
ADMIN_OPTION	VARCHAR2(3)		Indicates whether the grant was with the ADMIN option (YES) or not (NO)

Column	Datatype	NULL	Description
COMMON	VARCHAR2(3)		Indicates how the grant was made. Possible values: • YES if the privilege was granted commonly (CONTAINER=ALL was used) • NO if the privilege was granted locally (CONTAINER=ALL was not used)
INHERITED	VARCHAR2(3)		Indicates whether the grant was inherited from another container (YES) or not (NO)

Note:

This view is available starting with Oracle Database 23ai.

See Also:

"USER_SCHEMA_PRIVS"

6.399 DBA_SEC_RELEVANT_COLS

DBA_SEC_RELEVANT_COLS describes the security relevant columns of all security policies in the database. Its columns are the same as those in ALL_SEC_RELEVANT_COLS.

See Also:

"ALL_SEC_RELEVANT_COLS"

6.400 DBA_SECONDARY_OBJECTS

DBA_SECONDARY_OBJECTS provides information about all secondary objects that are associated with domain indexes in the database.

This view is only relevant in the context of domain indexes. Its columns are the same as those in "ALL_SECONDARY_OBJECTS".

6.401 DBA_SEGMENTS

DBA_SEGMENTS describes the storage allocated for all segments in the database.

Related View

USER_SEGMENTS describes the storage allocated for the segments owned by the current user's objects. This view does not display the OWNER, HEADER_FILE, HEADER_BLOCK, or RELATIVE_FNO columns.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)		Username of the segment owner
SEGMENT_NAME	VARCHAR2(128)		Name, if any, of the segment
PARTITION_NAME	VARCHAR2(128)		Object Partition Name (Set to NULL for nonpartitioned objects)
SEGMENT_TYPE	VARCHAR2(18)		Type of segment: • NESTED TABLE • TABLE • TABLE PARTITION • CLUSTER • LOBINDEX • INDEX • INDEX PARTITION • LOBSEGMENT • TABLE SUBPARTITION • INDEX SUBPARTITION • LOB PARTITION • LOB SUBPARTITION • ROLLBACK • TYPE2 UNDO • DEFERRED ROLLBACK • TEMPORARY • CACHE • SPACE HEADER • UNDEFINED
SEGMENT_SUBTYPE	VARCHAR2(10)		Subtype of LOB segment: SECUREFILE, ASSM, MSSM, and NULL
TABLESPACE_NAME	VARCHAR2(30)		Name of the tablespace containing the segment
HEADER_FILE	NUMBER		Absolute file number of the data file containing the segment header
HEADER_BLOCK	NUMBER		ID of the block containing the segment header
BYTES	NUMBER		Size, in bytes, of the segment
BLOCKS	NUMBER		Size, in Oracle blocks, of the segment
EXTENTS	NUMBER		Number of extents allocated to the segment
INITIAL_EXTENT	NUMBER		Size in bytes requested for the initial extent of the segment at create time. (Oracle rounds the extent size to multiples of 5 blocks if the requested size is greater than 5 blocks.)
NEXT_EXTENT	NUMBER		Size in bytes of the next extent to be allocated to the segment
MIN_EXTENTS	NUMBER		Minimum number of extents allowed in the segment
MAX_EXTENTS	NUMBER		Maximum number of extents allowed in the segment
MAX_SIZE	NUMBER		Maximum number of blocks allowed in the segment
RETENTION	VARCHAR2(7)		Retention option for SECUREFILE segment
MINRETENTION	NUMBER		Minimum retention duration for SECUREFILE segment
PCT_INCREASE	NUMBER		Percent by which to increase the size of the next extent to be allocated
FREELISTS	NUMBER		Number of process freelists allocated to this segment

Column	Datatype	NULL	Description
FREELIST_GROUPS	NUMBER		Number of freelist groups allocated to this segment
RELATIVE_FNO	NUMBER		Relative file number of the segment header
BUFFER_POOL	VARCHAR2(7)		Buffer pool to be used for segment blocks: • DEFAULT • KEEP • RECYCLE
FLASH_CACHE	VARCHAR2(7)		Database Smart Flash Cache hint to be used for segment blocks: • DEFAULT • KEEP • NONE Solaris and Oracle Linux functionality only.
CELL_FLASH_CACHE	VARCHAR2(7)		Cell flash cache hint to be used for segment blocks: • DEFAULT • KEEP • NONE **See Also:** Oracle Exadata Storage Server Software documentation for more information
INMEMORY	VARCHAR2(8)		Indicates whether the In-Memory Column Store (IM column store) is enabled (ENABLED) or disabled (DISABLED) for this segment
INMEMORY_PRIORITY	VARCHAR2(8)		Indicates the priority for In-Memory Column Store (IM column store) population: • LOW • MEDIUM • HIGH • CRITICAL • NONE • NULL
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)		Indicates the compression level for the IM column store: • NO MEMCOMPRESS • FOR DML • FOR QUERY [LOW | HIGH] • FOR CAPACITY [LOW | HIGH] • AUTO • NULL

Column	Datatype	NULL	Description
CELLMEMORY[11]	VARCHAR2(24)		The value for columnar compression in the storage cell flash cache. Possible values: • ENABLED: Oracle Exadata Storage will decide automatically whether to cache in columnar form • DISABLED: Oracle Exadata Storage is prevented from caching in columnar form • NO CACHECOMPRESS: Oracle Exadata Storage will cache in HCC format (no recompression) • FOR QUERY: Oracle Exadata Storage will recompress and cache in INMEMORY query high format • FOR CAPACITY: Oracle Exadata Storage will recompress and cache in INMEMORY capacity low format

1 This column is intended for use with Oracle Exadata

See Also:

"USER_SEGMENTS"

6.402 DBA_SEGMENTS_OLD

DBA_SEGMENTS_OLD lists information about storage allocated for all database segments.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)		Username of the segment owner
SEGMENT_NAME	VARCHAR2(128)		Name, if any, of the segment
PARTITION_NAME	VARCHAR2(128)		Name of the partition
SEGMENT_TYPE	VARCHAR2(18)		Type of segment: INDEX PARTITION, TABLE PARTITION, TABLE, CLUSTER, INDEX, ROLLBACK, DEFERRED ROLLBACK, TEMPORARY, CACHE, LOBSEGMENT and LOBINDEX
TABLESPACE_NAME	VARCHAR2(30)		Name of the tablespace containing the segment
HEADER_FILE	NUMBER		ID of the file containing the segment header
HEADER_BLOCK	NUMBER		ID of the block containing the segment header
BYTES	NUMBER		Size, in bytes, of the segment
BLOCKS	NUMBER		Size, in Oracle blocks, of the segment
EXTENTS	NUMBER		Number of extents allocated to the segment
INITIAL_EXTENT	NUMBER		Size in bytes requested for the initial extent of the segment at create time. (Oracle rounds the extent size to multiples of 5 blocks if the requested size is greater than 5 blocks.)
NEXT_EXTENT	NUMBER		Size in bytes of the next extent to be allocated to the segment
MIN_EXTENTS	NUMBER		Minimum number of extents allowed in the segment

Column	Datatype	NULL	Description
MAX_EXTENTS	NUMBER		Maximum number of extents allowed in the segment
PCT_INCREASE	NUMBER		Percent by which to increase the size of the next extent to be allocated
FREELISTS	NUMBER		Number of process freelists allocated to the segment
FREELIST_GROUPS	NUMBER		Number of freelist groups allocated to this segment
RELATIVE_FNO	NUMBER		Relative file number of the segment header
BUFFER_POOL	VARCHAR2(7)		Buffer pool for the object

6.403 DBA_SENSITIVE_COLUMN_TYPES

DBA_SENSITIVE_COLUMN_TYPES describes sensitive column types in the database.

Column	Datatype	NULL	Description
NAME	VARCHAR2(128)		The name of the sensitive column type
USER_COMMENT	VARCHAR2(4000)		User comment on the sensitive column type
SOURCE_NAME	VARCHAR2(128)		The name of the discovery source for the sensitive column type
SOURCE_TYPE	VARCHAR2(3)		The type of the discovery source: • ADM: import from ADM • DB: added within the database

See Also:

Oracle Database Security Guide for more information about transparent sensitive data protection

6.404 DBA_SENSITIVE_DATA

DBA_SENSITIVE_DATA describes the sensitive columns in the database.

Column	Datatype	NULL	Description
SENSITIVE#	NUMBER	NOT NULL	Dictionary ID for the sensitive data
SCHEMA_NAME	VARCHAR2(128)	NOT NULL	The schema containing the sensitive data
TABLE_NAME	VARCHAR2(128)	NOT NULL	The table containing the sensitive data
COLUMN_NAME	VARCHAR2(128)	NOT NULL	The name of the column identified as sensitive
SENSITIVE_TYPE	VARCHAR2(128)		The sensitive column type of the data
SOURCE_NAME	VARCHAR2(128)		The name of the discovery source for the sensitive data
USER_COMMENT	VARCHAR2(4000)		User comment on the sensitive data
TS	TIMESTAMP(6)		The time when the data was identified as sensitive in the database

> **See Also:**
>
> *Oracle Database Security Guide* for more information about transparent sensitive data protection

6.405 DBA_SEQUENCES

`DBA_SEQUENCES` describes all sequences in the database. Its columns are the same as those in `ALL_SEQUENCES`.

> **See Also:**
>
> "ALL_SEQUENCES"

6.406 DBA_SERVER_REGISTRY

`DBA_SERVER_REGISTRY` displays information about all server components in the database that are loaded into the component registry.

Column	Datatype	NULL	Description
COMP_ID	VARCHAR2(30)	NOT NULL	Component identifier
COMP_NAME	VARCHAR2(255)		Component name
VERSION	VARCHAR2(30)		Component version loaded
VERSION_FULL	VARCHAR2(30)		Component full version
STATUS	VARCHAR2(11)		Component status: • INVALID • VALID • LOADING • LOADED • UPGRADING • UPGRADED • DOWNGRADING • DOWNGRADED • REMOVING • REMOVED
MODIFIED	VARCHAR2(20)		Time when the component was last modified
CONTROL	VARCHAR2(128)	NOT NULL	User that created the component entry
SCHEMA	VARCHAR2(128)	NOT NULL	User that contains the objects for the component
PROCEDURE	VARCHAR2(61)		Validation procedure
STARTUP	VARCHAR2(8)		Indicates whether the component requires a startup after the upgrade (`REQUIRED`) or not
PARENT_ID	VARCHAR2(30)		Parent component identifier
OTHER_SCHEMAS	VARCHAR2(4000)		A list of ancillary schema names associated with the component

6.407 DBA_SERVICES

DBA_SERVICES displays all services in the database. The view excludes rows marked for deletion. Its columns are the same as those in ALL_SERVICES.

> **See Also:**
>
> "ALL_SERVICES"

6.408 DBA_SODA_COLLECTIONS

DBA_SODA_COLLECTIONS describes all Simple Oracle Document Access (SODA) collections in the database.

Related View

USER_SODA_COLLECTIONS describes the SODA collections owned by the current user. This view does not display the OWNER column.

Column	Datatype	NULL	Description
URI_NAME	NVARCHAR2(255)	NOT NULL	Collection name
OWNER	VARCHAR2(128)	NOT NULL	Owner of the collection
OBJECT_TYPE	VARCHAR2(10)	NOT NULL	Indicates whether the collection is table-based (TABLE) or view-based (VIEW)
OBJECT_SCHEMA	VARCHAR2(128)	NOT NULL	Name of the schema that includes the table or view on which the collection is based
OBJECT_NAME	VARCHAR2(128)	NOT NULL	Name of the table or view on which the collection is based
CREATED_ON	TIMESTAMP(6)	NOT NULL	Collection creation time
CREATE_MODE	VARCHAR2(10)	NOT NULL	Creation mode. Possible values: • DDL: A new table was created at collection creation time • MAP: The collection was created by mapping a preexisting table or view **Note:** For view-based collections, the value of this column is always MAP. New views are not created for view-based collections. A view-based collection is always created by mapping a preexisting view.
JSON_DESCRIPTOR	VARCHAR2(4000)	NOT NULL	Collection metadata, expressed in JavaScript Object Notation (JSON)

> **See Also:**
>
> "USER_SODA_COLLECTIONS"

6.409 DBA_SOURCE

DBA_SOURCE describes the text source of all stored objects in the database. Its columns are the same as those in ALL_SOURCE.

> **See Also:**
>
> "ALL_SOURCE"

6.410 DBA_SOURCE_AE

DBA_SOURCE_AE describes the text source of all stored objects (across all editions) in the database. Its columns are the same as those in ALL_SOURCE_AE.

> **See Also:**
>
> "ALL_SOURCE_AE"

6.411 DBA_SQL_ERROR_MITIGATIONS

DBA_SQL_ERROR_MITIGATIONS describes automatic error mitigations for SQL statements.

Column	Datatype	NULL	Description
SQL_ID	VARCHAR2(13)		Identifier for the relevant SQL statement
SIGNATURE	NUMBER		Unique SQL identifier generated from normalized SQL text
FORCE_MATCHING	VARCHAR2(3)		Indicates whether the signature is force matching (YES) or exact matching (NO)
SQL_TEXT	CLOB		SQL text
INCIDENT_ID	NUMBER		ID for the incident associated with the error mitigation
PROBLEM_KEY	VARCHAR2(64)		Description of the problem
PROBLEM_TYPE	VARCHAR2(17)		Problem type: • COMPILATION ERROR • EXECUTION ERROR
CREATED	TIMESTAMP(6) WITH TIME ZONE		Date and time at which the first automatic error mitigation was created on this SQL
START_TIME	TIMESTAMP(6) WITH TIME ZONE		Start date and time of automatic error mitigation
END_TIME	TIMESTAMP(6) WITH TIME ZONE		End date and time of automatic error mitigation
NOTES	CLOB		Mitigation notes

Column	Datatype	NULL	Description
MITIGATION_TYPE	VARCHAR2(14)		Mitigation type: • ALTERNATE PLAN • SQL PATCH
MITIGATION_NAME	VARCHAR2(128)		Mitigation name
MITIGATION_DETAILS	CLOB		Mitigation details
STATUS	CHAR(7)		Mitigation status: • SUCCESS • FAILURE
LAST_VERIFIED	TIMESTAMP(6) WITH TIME ZONE		Date and time at which the auto patch was last verified

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

6.412 DBA_SQL_FIREWALL_ALLOW_LISTS

DBA_SQL_FIREWALL_ALLOW_LISTS displays information about SQL Firewall allow-lists.

Column	Datatype	NULL	Description
USERNAME	VARCHAR2(128)		Name of the target user
GENERATED_ON	TIMESTAMP(6) WITH TIME ZONE	NOT NULL	Date and time of allow-list generation
STATUS	VARCHAR2(8)		Allow-list status (ENABLED or DISABLED)
STATUS_UPDATED_ON	TIMESTAMP(6) WITH TIME ZONE	NOT NULL	Date and time of the most recent allow-list status change
TOP_LEVEL_ONLY	VARCHAR2(14)		Indicates whether the allow-list should be enforced on only top-level SQL commands, that is, SQL commands issued directly from the user (Y) or on all SQL commands, including top-level SQL commands and SQL commands from PL/SQL units (N)
ENFORCE	VARCHAR2(15)		Option of the allow-list enforcement: • ENFORCE_CONTEXT - Allowed contexts will be checked and enforced during database connection • ENFORCE_SQL - Allowed SQLs will be checked and enforced for every SQL statement execution • ENFORCE_ALL - Both allowed contexts and allowed SQLs will be checked and enforced
BLOCK	VARCHAR2(14)		If the allow-list is enabled, indicates whether the allow-list is enabled in blocking mode (Y) or non-blocking mode (N)

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

6.413 DBA_SQL_FIREWALL_ALLOWED_IP_ADDR

DBA_SQL_FIREWALL_ALLOWED_IP_ADDR lists the IP addresses that SQL Firewall target users are allowed to use to connect to the database.

Column	Datatype	NULL	Description
USERNAME	VARCHAR2(128)		Name of the target user
IP_ADDRESS	VARCHAR2(128)	NOT NULL	Allowed client IP address

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

6.414 DBA_SQL_FIREWALL_ALLOWED_OS_PROG

DBA_SQL_FIREWALL_ALLOWED_OS_PROG lists the OS programs that SQL Firewall target users are allowed to use to connect to the database.

Column	Datatype	NULL	Description
USERNAME	VARCHAR2(128)		Name of the target user
OS_PROGRAM	VARCHAR2(128)	NOT NULL	Allowed OS program name

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

6.415 DBA_SQL_FIREWALL_ALLOWED_OS_USER

DBA_SQL_FIREWALL_ALLOWED_OS_USER lists the OS user names that SQL Firewall target users are allowed to use to connect to the database.

Column	Datatype	NULL	Description
USERNAME	VARCHAR2(128)		Name of the target user
OS_USER	VARCHAR2(128)	NOT NULL	Allowed OS user name

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

6.416 DBA_SQL_FIREWALL_ALLOWED_SQL

DBA_SQL_FIREWALL_ALLOWED_SQL displays information about allowed SQL for SQL Firewall allow-lists.

Column	Datatype	NULL	Description
USERNAME	VARCHAR2(128)		Name of the target user
ALLOWED_SQL_ID	NUMBER	NOT NULL	Unique ID of allowed SQL entry for the target user
SQL_SIGNATURE	VARCHAR2(64)	NOT NULL	Allowed SQL signature
SQL_TEXT	VARCHAR2(4000)		SQL text (up to 1000 characters) To view the full SQL text, query the SQL_TEXT column of the DBA_SQL_FIREWALL_SQL_LOGS view.
ACCESSED_OBJECTS	VARCHAR2(4000)		List of accessed objects (up to 1000 characters) To view the full list of accessed objects, query the ACCESSED_OBJECTS column of the DBA_SQL_FIREWALL_SQL_LOGS view.
CURRENT_USER	VARCHAR2(128)		Name of the user who invoked the SQL command
TOP_LEVEL	VARCHAR2(9)		Indicates whether the allowed SQL entry is a top-level SQL command, that is, a SQL command issued directly from the user (Y) or a SQL command from a PL/SQL unit (N)
VERSION	NUMBER	NOT NULL	Version number for the allow-list when the allowed SQL was added You can use this value to determine whether specific allowed SQLs were added to the allow-list in the same batch.

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

> **See Also:**
>
> "DBA_SQL_FIREWALL_SQL_LOGS"

6.417 DBA_SQL_FIREWALL_CAPTURE_LOGS

DBA_SQL_FIREWALL_CAPTURE_LOGS displays SQL Firewall capture logs.

Column	Datatype	NULL	Description
USERNAME	VARCHAR2(128)		Name of the target user
SESSION_ID	NUMBER	NOT NULL	SQL Firewall session ID
COMMAND_TYPE	VARCHAR2(64)		Type of SQL command
SQL_SIGNATURE	VARCHAR2(64)		SQL signature
SQL_TEXT	VARCHAR2(4000)		SQL text (up to 1000 characters) To view the full SQL text, query the SQL_TEXT column of the DBA_SQL_FIREWALL_SQL_LOGS view.
ACCESSED_OBJECTS	VARCHAR2(4000)		List of accessed objects (up to 1000 characters) To view the full list of accessed objects, query the ACCESSED_OBJECTS column of the DBA_SQL_FIREWALL_SQL_LOGS view.
CURRENT_USER	VARCHAR2(128)		Name of the user who invoked the SQL command
TOP_LEVEL	VARCHAR2(9)		Indicates whether the SQL command is a top-level SQL command, that is, a SQL command issued directly from the user (Y) or a SQL command from a PL/SQL unit (N)
CLIENT_PROGRAM	VARCHAR2(84)		Name of the client program
OS_USER	VARCHAR2(128)		Name of the OS user of the client process
IP_ADDRESS	VARCHAR2(48)		Client IP address

Note:

This view is available starting with Oracle Database 23ai.

See Also:

"DBA_SQL_FIREWALL_SQL_LOGS"

6.418 DBA_SQL_FIREWALL_CAPTURES

DBA_SQL_FIREWALL_CAPTURES displays information about SQL Firewall captures.

Column	Datatype	NULL	Description
USERNAME	VARCHAR2(128)		Name of the target user
TOP_LEVEL_ONLY	VARCHAR2(14)		Indicates whether SQL Firewall should capture only top-level SQL commands, that is, SQL commands issued directly from the user (Y) or all SQL commands, including top-level SQL commands and SQL commands from PL/SQL units (N)
STATUS	VARCHAR2(8)		Capture status (ENABLED or DISABLED)

Column	Datatype	NULL	Description
LAST_STARTED_ON	TIMESTAMP(6) WITH TIME ZONE		Date and time at which the capture was last started
LAST_STOPPED_ON	TIMESTAMP(6) WITH TIME ZONE		Date and time at which the capture was last stopped

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

6.419 DBA_SQL_FIREWALL_SESSION_LOGS

DBA_SQL_FIREWALL_SESSION_LOGS displays SQL Firewall session logs.

Column	Datatype	NULL	Description
SESSION_ID	NUMBER	NOT NULL	SQL Firewall session ID
USERNAME	VARCHAR2(128)		Name of the target user
LOGIN_TIME	TIMESTAMP(6) WITH TIME ZONE		Date and time at which the target user logged in to the database
IP_ADDRESS	VARCHAR2(48)		Client IP address
CLIENT_PROGRAM	VARCHAR2(84)		Name of the client program
OS_USER	VARCHAR2(128)		Name of the OS user of the client process

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

6.420 DBA_SQL_FIREWALL_SQL_LOGS

DBA_SQL_FIREWALL_SQL_LOGS displays information about SQL logs for SQL Firewall.

Column	Datatype	NULL	Description
COMMAND_TYPE	VARCHAR2(64)		Type of SQL command
SQL_SIGNATURE	VARCHAR2(64)		SQL signature
SQL_TEXT	CLOB		SQL text
ACCESSED_OBJECTS	CLOB		List of accessed objects
CHARSET	VARCHAR2(64)		Character set of the SQL text

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

6.421 DBA_SQL_FIREWALL_STATUS

DBA_SQL_FIREWALL_STATUS displays the status of SQL Firewall.

Column	Datatype	NULL	Description
STATUS	VARCHAR2(8)		SQL Firewall status (ENABLED or DISABLED)
STATUS_UPDATED_ON	TIMESTAMP(6) WITH TIME ZONE	NOT NULL	Date and time of the most recent SQL Firewall status change
EXCLUDE_JOBS	VARCHAR2(12)		Indicates whether the SQL Firewall will capture or enforce allow-lists for database connections and SQL executions of Oracle scheduler job sessions (Y) or not (N)

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

6.422 DBA_SQL_FIREWALL_VIOLATIONS

DBA_SQL_FIREWALL_VIOLATIONS lists SQL Firewall violations.

Column	Datatype	NULL	Description
USERNAME	VARCHAR2(128)		Name of the target user
COMMAND_TYPE	VARCHAR2(64)		Type of SQL command
SQL_SIGNATURE	VARCHAR2(64)		SQL signature
SQL_TEXT	VARCHAR2(4000)		SQL text (up to 1000 characters) To view the full SQL text, query the SQL_TEXT column of the DBA_SQL_FIREWALL_SQL_LOGS view.
ACCESSED_OBJECTS	VARCHAR2(4000)		List of accessed objects (up to 1000 characters) To view the full list of accessed objects, query the ACCESSED_OBJECTS column of the DBA_SQL_FIREWALL_SQL_LOGS view.
CURRENT_USER	VARCHAR2(128)		Name of the user who invoked the SQL command
TOP_LEVEL	VARCHAR2(9)		Indicates whether the SQL command is a top-level SQL command, that is, a SQL command issued directly from the user (Y) or a SQL command from a PL/SQL unit (N)
IP_ADDRESS	VARCHAR2(48)		Client IP address
CLIENT_PROGRAM	VARCHAR2(84)		Name of the client program

Column	Datatype	NULL	Description
OS_USER	VARCHAR2(128)		Name of the OS user of the client process
CAUSE	VARCHAR2(17)		Cause of the violation: • Context violation • SQL violation
FIREWALL_ACTION	VARCHAR2(7)		Indicates whether the SQL Firewall action was Allowed or Blocked
OCCURRED_AT	TIMESTAMP(6) WITH TIME ZONE		Date and time of the violation

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

> **See Also:**
>
> "DBA_SQL_FIREWALL_SQL_LOGS"

6.423 DBA_SQL_MANAGEMENT_CONFIG

DBA_SQL_MANAGEMENT_CONFIG displays the configuration parameters of the SQL management base.

You must have the DBA role in order to change the configuration parameter values.

Column	Datatype	NULL	Description
PARAMETER_NAME	VARCHAR2(128)	NOT NULL	Name of the configuration parameter: • AUTO_CAPTURE_ACTION • AUTO_CAPTURE_MODULE • AUTO_CAPTURE_PARSING_SCHEMA_NAME • AUTO_CAPTURE_SQL_TEXT • PLAN_RETENTION_WEEKS • SPACE_BUDGET_PERCENT
PARAMETER_VALUE	VARCHAR2(4000)		Value of the configuration parameter
PARAMETER_VALUE_CLOB	CLOB		Value of the configuration parameter, in CLOB format
LAST_MODIFIED	TIMESTAMP(6)		Time the parameter value was last updated
MODIFIED_BY	VARCHAR2(128)		User who last updated the parameter value

6.424 DBA_SQL_QUARANTINE

DBA_SQL_QUARANTINE displays information about SQL Quarantine configurations.

Each row in this view represents a quarantine configuration for a SQL plan.

Column	Datatype	NULL	Description
SIGNATURE	NUMBER	NOT NULL	Unique SQL identifier generated from normalized SQL text
NAME	VARCHAR2(128)	NOT NULL	Unique plan identifier in string form as a search key
SQL_TEXT	CLOB	NOT NULL	Un-normalized SQL text
PLAN_HASH_VALUE	NUMBER		Unique plan identifier in numeric form as a search key
CPU_TIME	VARCHAR2(4000)		CPU time threshold (in seconds)
IO_MEGABYTES	VARCHAR2(4000)		I/O threshold (in megabytes)
IO_REQUESTS	VARCHAR2(4000)		Physical I/O threshold (number of physical I/O requests)
ELAPSED_TIME	VARCHAR2(4000)		Elapsed time threshold (in seconds)
IO_LOGICAL	VARCHAR2(4000)		Logical I/O threshold (number of logical I/O requests)
CREATOR	VARCHAR2(128)		User who created the quarantine configuration
ORIGIN	VARCHAR2(16)		Method by which the quarantine configuration was created. The only possible value is RESOURCE-MANAGER, which indicates that the quarantine configuration was created by the Resource Manager.
DESCRIPTION	VARCHAR2(500)		Text description
CREATED	TIMESTAMP(6)	NOT NULL	Time at which the quarantine configuration was created
LAST_EXECUTED	TIMESTAMP(6)		Time at which the quarantine configuration was last used
ENABLED	VARCHAR2(3)		Indicates whether the quarantine configuration is enabled (YES) or disabled (NO)
AUTOPURGE	VARCHAR2(3)		Indicates whether the quarantine configuration is auto-purged (YES) or not (NO)

See Also:

Oracle Database SQL Tuning Guide for more information about quarantined SQL plans

6.425 DBA_SQL_PATCHES

DBA_SQL_PATCHES displays the set of SQL patches.

Column	Datatype	NULL	Description
NAME	VARCHAR2(128)	NOT NULL	Name of the SQL patch
CATEGORY	VARCHAR2(128)	NOT NULL	Category of the SQL patch
SIGNATURE	NUMBER	NOT NULL	Unique identifier generated from normalized SQL text
SQL_TEXT	CLOB	NOT NULL	Un-normalized SQL text
CREATED	TIMESTAMP(6)	NOT NULL	Timestamp when the SQL patch was created
LAST_MODIFIED	TIMESTAMP(6)		Timestamp when the SQL patch was last modified

Column	Datatype	NULL	Description
DESCRIPTION	VARCHAR2(500)		Text description provided for the SQL patch
ORIGIN	VARCHAR2(22)		Origin of the patch: • AUTO-FOREGROUND-REPAIR • PATCH • UNKNOWN
STATUS	VARCHAR2(8)		Status of the SQL patch: • ENABLED • DISABLED
FORCE_MATCHING	VARCHAR2(3)		Indicates whether the signature is force matching (YES) or exact matching (NO)
TASK_ID	NUMBER		Advisor task ID that generated the SQL patch
TASK_EXEC_NAME	VARCHAR2(128)		Advisor execution name for the SQL patch
TASK_OBJ_ID	NUMBER		Advisor object ID for the SQL patch
TASK_FND_ID	NUMBER		Advisor finding ID for the SQL patch
TASK_REC_ID	NUMBER		Advisor recommendation ID for the SQL patch

6.426 DBA_SQL_PLAN_BASELINES

DBA_SQL_PLAN_BASELINES displays information about the SQL plan baselines currently created for specific SQL statements.

Column	Datatype	NULL	Description
SIGNATURE	NUMBER	NOT NULL	Unique SQL identifier generated from normalized SQL text
SQL_HANDLE	VARCHAR2(30)	NOT NULL	Unique SQL identifier in string form as a search key
SQL_TEXT	CLOB	NOT NULL	Un-normalized SQL text
PLAN_NAME	VARCHAR2(128)	NOT NULL	Unique plan identifier in string form as a search key
CREATOR	VARCHAR2(128)		User who created the plan baseline
ORIGIN	VARCHAR2(29)		How the plan baseline was created: • ADDM-SQLTUNE • AUTO-CAPTURE • AUTO-SQLTUNE • EVOLVE-AUTO-INDEX-LOAD • EVOLVE-CREATE-FROM-ADAPTIVE • EVOLVE-LOAD-FROM-AWR • EVOLVE-LOAD-FROM-CURSOR-CACHE • EVOLVE-LOAD-FROM-STS • FOREGROUND-CAPTURE • MANUAL-LOAD • MANUAL-LOAD-FROM-AWR • MANUAL-LOAD-FROM-CURSOR-CACHE • MANUAL-LOAD-FROM-STS • MANUAL-SQLTUNE • STORED-OUTLINE
PARSING_SCHEMA_NAME	VARCHAR2(128)		Name of the parsing schema
DESCRIPTION	VARCHAR2(500)		Text description provided for the plan baseline

Column	Datatype	NULL	Description
VERSION	VARCHAR2(64)		Database version at the time of plan baseline creation
CREATED	TIMESTAMP(6)	NOT NULL	Timestamp when the plan baseline was created
LAST_MODIFIED	TIMESTAMP(6)		Timestamp when the plan baseline was last modified
LAST_EXECUTED	TIMESTAMP(6)		Timestamp when the plan baseline was last executed **Note:** For performance reasons, this column is not updated immediately after each execution of the plan baseline. Therefore, the plan baseline may have been executed more recently than the value of this column indicates.
LAST_VERIFIED	TIMESTAMP(6)		Timestamp when the plan baseline was last verified
ENABLED	VARCHAR2(3)		Indicates whether the plan baseline is enabled (YES) or disabled (NO)
ACCEPTED	VARCHAR2(3)		Indicates whether the plan baseline is accepted (YES) or not (NO)
FIXED	VARCHAR2(3)		Indicates whether the plan baseline is fixed (YES) or not (NO)
REPRODUCED	VARCHAR2(3)		Indicates whether the optimizer was able to reproduce the plan (YES) or not (NO). The value of this column is set to YES when a plan is initially added to the plan baseline.
AUTOPURGE	VARCHAR2(3)		Indicates whether the plan baseline is auto-purged (YES) or not (NO)
ADAPTIVE	VARCHAR2(3)		Indicates whether a plan that is automatically captured by SQL plan management is marked adaptive or not. When a new adaptive plan is found for a SQL statement that has an existing SQL plan baseline, that new plan will be added to the SQL plan baseline as an unaccepted plan, and the ADAPTIVE column will be marked YES. When this new plan is verified (either manually or via the auto evolve task), the plan will be test executed and the final plan determined at execution will become an accepted plan if its performance is better than the existing plan baseline. At this point, the value of the ADAPTIVE column is set to NO since the plan is no longer adaptive, but resolved.
OPTIMIZER_COST	NUMBER		Optimizer cost at the time the plan baseline was created
MODULE	VARCHAR2(64)		Application module name
ACTION	VARCHAR2(64)		Application action
EXECUTIONS[1]	NUMBER		Number of executions at the time the plan baseline was created
ELAPSED_TIME[1]	NUMBER		Total elapsed time (in microseconds) at the time the plan baseline was created
CPU_TIME[1]	NUMBER		Total CPU time (in microseconds) at the time the plan baseline was created
BUFFER_GETS[1]	NUMBER		Total buffer gets at the time the plan baseline was created
DISK_READS[1]	NUMBER		Total disk reads at the time the plan baseline was created

Column	Datatype	NULL	Description
DIRECT_WRITES[1]	NUMBER		Total direct writes at the time the plan baseline was created
ROWS_PROCESSED[1]	NUMBER		Total rows processed at the time the plan baseline was created
FETCHES[1]	NUMBER		Total number of fetches at the time the plan baseline was created
END_OF_FETCH_COUNT[1]	NUMBER		Total number of full fetches at the time the plan baseline was created
FOREGROUND_LAST_VERIFIED	TIMESTAMP(6)		Timestamp when the SQL plan baseline was verified by real-time SQL plan management
NOTES	CLOB		Internal SQL plan baseline verification information

1 If the value of the ORIGIN column is equal to AUTO-CAPTURE, then data for this column is not populated.

See Also:

- *Oracle Database SQL Tuning Guide* for more information about SQL plan baselines
- The DBMS_SQLTUNE package in *Oracle Database PL/SQL Packages and Types Reference*

6.427 DBA_SQL_PLAN_DIR_OBJECTS

DBA_SQL_PLAN_DIR_OBJECTS displays the objects created in the SQL plan directive.

Column	Datatype	NULL	Description
DIRECTIVE_ID	NUMBER		The identifier of the SQL plan directive
OWNER	VARCHAR2(128)		The username of the owner of the object in the SQL plan directive
OBJECT_NAME	VARCHAR2(128)		The name of the object in the SQL plan directive
SUBOBJECT_NAME	VARCHAR2(128)		The name of the subobject (for example, column) in the SQL plan directive
OBJECT_TYPE	VARCHAR2(6)		The type of the subobject in the SQL plan directive
NUM_ROWS	NUMBER		The number of rows in the object when the directive is created
NOTES	XMLTYPE		Other notes about the object

See Also:

- "DBA_SQL_PLAN_DIRECTIVES"
- *Oracle Database SQL Tuning Guide* for more information about SQL plan directives

6.428 DBA_SQL_PLAN_DIRECTIVES

DBA_SQL_PLAN_DIRECTIVES displays information about the SQL plan directives in the system.

Column	Datatype	NULL	Description
DIRECTIVE_ID	NUMBER	NOT NULL	The identifier of the SQL plan directive
TYPE	VARCHAR2(16)		The type of the SQL plan directive: • DYNAMIC SAMPLING: SQL plan directive • DYNAMIC SAMPLING RESULT: Dynamic sampling query results. This value appears only in Oracle Database 12*c* Release 2 (12.2.0.1) and later releases. • UNKNOWN: Unknown
ENABLED	VARCHAR2(3)		Indicates whether the SQL plan directive is enabled. Possible values: • YES: The SQL plan directive is enabled. • NO: The SQL plan directive is not enabled.
STATE	VARCHAR2(10)		The state of the SQL plan directive. Possible values include: • SUPERSEDED: This value indicates that the corresponding column or groups have an extension or histogram, or that another SQL plan directive exists that can be used for the directive. • USABLE: This value indicates that the SQL plan directive is usable for the optimizer.
AUTO_DROP	VARCHAR2(3)		If YES, the SQL plan directive gets dropped when unused beyond SPD_RETENTION_WEEKS
REASON	VARCHAR2(36)		The reason for creating the SQL plan directive
CREATED	TIMESTAMP(6)		The creation timestamp of the SQL plan directive
LAST_MODIFIED	TIMESTAMP(6)		The timestamp of most recent modification of the SQL plan directive
LAST_USED	TIMESTAMP(9)		The timestamp of most recent usage of the SQL plan directive
NOTES	XMLTYPE		Extra information about the SQL plan directive

See Also:

- "DBA_SQL_PLAN_DIR_OBJECTS"
- *Oracle Database SQL Tuning Guide* for more information about SQL plan directives

6.429 DBA_SQL_PROFILES

DBA_SQL_PROFILES displays information about SQL profiles currently created for specific SQL statements.

Column	Datatype	NULL	Description
NAME	VARCHAR2(128)	NOT NULL	Name of the SQL profile
CATEGORY	VARCHAR2(128)	NOT NULL	Category of the SQL profile
SIGNATURE	NUMBER	NOT NULL	Unique identifier generated from normalized SQL text
SQL_TEXT	CLOB	NOT NULL	Un-normalized SQL text
CREATED	TIMESTAMP(6)	NOT NULL	Timestamp when the SQL profile was created
LAST_MODIFIED	TIMESTAMP(6)		Timestamp when the SQL profile was last modified
DESCRIPTION	VARCHAR2(500)		Text description provided for the SQL profile
TYPE	VARCHAR2(7)		Type of the SQL profile (how it was created): • MANUAL • AUTO
STATUS	VARCHAR2(8)		Status of the SQL profile: • ENABLED • DISABLED • VOID
FORCE_MATCHING	VARCHAR2(3)		If YES, this causes SQL Profiles to target all SQL statements which have the same text after normalizing all literal values to bind variables. If a combination of literal values and bind variables is used in the same SQL text, then no transformation occurs. This is analogous to the matching algorithm use by the FORCE option of the CURSOR_SHARING parameter. If NO, literals are not transformed. This is analogous to the matching algorithm used by the EXACT option of the CURSOR_SHARING parameter.
TASK_ID	NUMBER		Advisor task ID that generated the SQL profile
TASK_EXEC_NAME	VARCHAR2(128)		Advisor execution name for the SQL profile
TASK_OBJ_ID	NUMBER		Advisor object ID for the SQL profile
TASK_FND_ID	NUMBER		Advisor finding ID for the SQL profile
TASK_REC_ID	NUMBER		Advisor recommendation ID for the SQL profile
TASK_CON_DBID	NUMBER		Database ID for the PDB tuning task generating the SQL profile

> **See Also:**
>
> The `DBMS_SQLTUNE` package in *Oracle Database PL/SQL Packages and Types Reference*

6.430 DBA_SQL_TRANSLATION_PROFILES

`DBA_SQL_TRANSLATION_PROFILES` describes all SQL translation profiles in the database. Its columns are the same as those in `ALL_SQL_TRANSLATION_PROFILES`.

> **See Also:**
>
> "ALL_SQL_TRANSLATION_PROFILES"

6.431 DBA_SQL_TRANSLATIONS

`DBA_SQL_TRANSLATIONS` describes all SQL translations in the database. Its columns are the same as those in `ALL_SQL_TRANSLATIONS`.

> **See Also:**
>
> "ALL_SQL_TRANSLATIONS"

6.432 DBA_SQLJ_TYPE_ATTRS

`DBA_SQLJ_TYPE_ATTRS` describes the attributes of all SQLJ object types in the database. Its columns are the same as those in `ALL_SQLJ_TYPE_ATTRS`.

> **See Also:**
>
> "ALL_SQLJ_TYPE_ATTRS"

6.433 DBA_SQLJ_TYPE_METHODS

`DBA_SQLJ_TYPE_METHODS` describes the methods of all SQLJ object types in the database. Its columns are the same as those in `ALL_SQLJ_TYPE_METHODS`.

> **See Also:**
>
> "ALL_SQLJ_TYPE_METHODS"

6.434 DBA_SQLJ_TYPES

`DBA_SQLJ_TYPES` describes all SQLJ object types in the database. Its columns are the same as those in `ALL_SQLJ_TYPES`.

> **See Also:**
>
> "ALL_SQLJ_TYPES"

6.435 DBA_SQLSET

`DBA_SQLSET` displays information about all SQL tuning sets in the database. Its columns are the same as those in `ALL_SQLSET`.

> **See Also:**
>
> "ALL_SQLSET"

6.436 DBA_SQLSET_BINDS

`DBA_SQLSET_BINDS` displays the bind values associated with all SQL tuning sets in the database. Its columns are the same as those in `ALL_SQLSET_BINDS`.

> **See Also:**
>
> "ALL_SQLSET_BINDS"

6.437 DBA_SQLSET_PLANS

`DBA_SQLSET_PLANS` describes captured plans in the SQL tuning sets in the database. Its columns are the same as those in `ALL_SQLSET_PLANS`.

> **See Also:**
>
> "ALL_SQLSET_PLANS"

6.438 DBA_SQLSET_REFERENCES

DBA_SQLSET_REFERENCES describes whether or not all SQL tuning sets in the database are active. A SQL tuning set cannot be dropped if it is referenced. Its columns are the same as those in ALL_SQLSET_REFERENCES.

> **See Also:**
>
> "ALL_SQLSET_REFERENCES"

6.439 DBA_SQLSET_STATEMENTS

DBA_SQLSET_STATEMENTS displays information about the SQL statements, along with their statistics, that form all SQL tuning sets in the database. Its columns, except for PARSING_SCHEMA_ID, are the same as those in ALL_SQLSET_STATEMENTS.

Column	Datatype	NULL	Description
SQLSET_NAME	VARCHAR2(128)	NOT NULL	Name of the SQL tuning set for the statement
SQLSET_OWNER	VARCHAR2(128)		User name of the SQL tuning set owner
SQLSET_ID	NUMBER	NOT NULL	ID of the SQL tuning set for the statement
CON_DBID	NUMBER	NOT NULL	The database ID of the PDB
SQL_ID	VARCHAR2(13)	NOT NULL	SQL identifier of the parent cursor in the library cache
FORCE_MATCHING_SIGNATURE	NUMBER	NOT NULL	The signature used when the CURSOR_SHARING parameter is set to FORCE
SQL_TEXT	CLOB		Full text for the SQL statement exposed as a CLOB column.
PARSING_SCHEMA_NAME	VARCHAR2(128)		Name of the user in whose schema the statement was parsed
PARSING_SCHEMA_ID	NUMBER		ID of the schema in which the statement was parsed
PLAN_HASH_VALUE	NUMBER	NOT NULL	Hash value for the plan corresponding to statistics in this record
BIND_DATA	RAW(2000)		Bind data
BINDS_CAPTURED	CHAR(1)		Binds captured
MODULE	VARCHAR2(64)		Contains the name of the module that was executing at the time that the SQL statement was first parsed, which is set by calling DBMS_APPLICATION_INFO.SET_MODULE
ACTION	VARCHAR2(64)		Contains the name of the action that was executing at the time that the SQL statement was first parsed, which is set by calling DBMS_APPLICATION_INFO.SET_ACTION
ELAPSED_TIME	NUMBER		Elapsed time (in microseconds) used by this cursor for parsing, executing, and fetching
CPU_TIME	NUMBER		CPU time (in microseconds) used by this cursor for parsing, executing, and fetching
BUFFER_GETS	NUMBER		Number of buffer gets for this child cursor

Column	Datatype	NULL	Description
DISK_READS	NUMBER		Number of disk reads for this child cursor
DIRECT_WRITES	NUMBER		Number of direct writes for this child cursor
ROWS_PROCESSED	NUMBER		Total number of rows that the parsed SQL statement returns
FETCHES	NUMBER		Number of fetches associated with the SQL statement
EXECUTIONS	NUMBER		Number of executions that took place on this object since it was brought into the library cache
END_OF_FETCH_COUNT	NUMBER		Number of times this cursor was fully executed since the cursor was brought into the library cache. The value of this statistic in not incremented when the cursor is partially executed, either because it failed during the execution or because only the first few rows produced by this cursor are fetched before the cursor is closed or re-executed. By definition, the value of the END_OF_FETCH_COUNT column should be less than, or equal to, the value of the EXECUTIONS column.
OPTIMIZER_COST	NUMBER		Cost of this query, given by the optimizer
OPTIMIZER_ENV	RAW(2000)		Optimizer environment
PRIORITY	NUMBER		User-defined priority
COMMAND_TYPE	NUMBER		Oracle command type definition
FIRST_LOAD_TIME	VARCHAR2(19)		Timestamp of the parent creation time
STAT_PERIOD	NUMBER		Period of time (in seconds) during which the statistics of the SQL statement were collected
ACTIVE_STAT_PERIOD	NUMBER		Effective period of time (in seconds) during which the SQL statement was active
OTHER	CLOB		Client data, specified by the user, for this statement
PLAN_TIMESTAMP	DATE		Timestamp for the plan corresponding to the statistics in this record
SQL_SEQ	NUMBER	NOT NULL	SQL sequence
LAST_EXEC_START_TIME	VARCHAR2(19)		For SQLs captured from the cursor cache, this is the time when the most recent execution of this SQL started

See Also:

- "ALL_SQLSET_STATEMENTS"
- *Oracle Database PL/SQL Packages and Types Reference* for more information about the DBMS_APPLICATION_INFO.SET_MODULE procedure
- *Oracle Database PL/SQL Packages and Types Reference* for more information about the DBMS_APPLICATION_INFO.SET_ACTION procedure

6.440 DBA_SQLTUNE_BINDS

DBA_SQLTUNE_BINDS displays the bind values associated with all tuned SQL statements in the database.

Related View

USER_SQLTUNE_BINDS displays the bind values associated with the tuned SQL statements owned by the current user.

Column	Datatype	NULL	Description
TASK_ID	NUMBER(38)	NOT NULL	Tuning task identifier
OBJECT_ID	NUMBER(38)	NOT NULL	Advisor framework object identifier
POSITION	NUMBER(38)	NOT NULL	Bind position
VALUE	ANYDATA		Bind value. This column is NULL for PL/SQL bind types.

> **See Also:**
>
> "USER_SQLTUNE_BINDS"

6.441 DBA_SQLTUNE_PLANS

DBA_SQLTUNE_PLANS displays information about the execution plans generated for all SQL statements in the database during a SQL tuning session.

Related View

USER_SQLTUNE_PLANS displays information about the execution plans generated for the SQL statements owned by the current user during a SQL tuning session.

Column	Datatype	NULL	Description
TASK_ID	NUMBER(38)	NOT NULL	Advisor task ID
EXECUTION_NAME	VARCHAR2(128)	NOT NULL	Advisor task execution
OBJECT_ID	NUMBER(38)	NOT NULL	Advisor object ID
ATTRIBUTE	VARCHAR2(34)		Text string identifying the type of the execution plan: • Original - Original plan of the query • Original with adjusted cost - Same as Original but with adjusted cost • Using SQL profile - Plan with SQL profile applied • Using new indices - Plan with indexes applied
STATEMENT_ID	VARCHAR2(30)		Optional statement identifier specified in the EXPLAIN PLAN statement
PLAN_HASH_VALUE	NUMBER	NOT NULL	Numerical representation of the execution plan
PLAN_ID	NUMBER		Plan identifier

Column	Datatype	NULL	Description
TIMESTAMP	DATE		Date and time when the EXPLAIN PLAN statement was issued
REMARKS	VARCHAR2(4000)		Place for comments that can be added to the steps of the execution plan
OPERATION	VARCHAR2(30)		Name of the operation performed at this step
OPTIONS	VARCHAR2(255)		Options used for the operation performed at this step
OBJECT_NODE	VARCHAR2(128)		Name of the database link used to reference the object
OBJECT_OWNER	VARCHAR2(128)		Owner of the object
OBJECT_NAME	VARCHAR2(128)		Name of the object
OBJECT_ALIAS	VARCHAR2(261)		Object alias
OBJECT_INSTANCE	NUMBER(38)		Numbered position of the object name in the original SQL statement
OBJECT_TYPE	VARCHAR2(30)		Descriptive modifier that further describes the type of object
OPTIMIZER	VARCHAR2(255)		Current mode of the optimizer
SEARCH_COLUMNS	NUMBER		Number of index columns with start and stop keys (that is, the number of columns with matching predicates)
ID	NUMBER(38)	NOT NULL	Identification number for this step in the execution plan
PARENT_ID	NUMBER(38)		ID of the next step that operates on the results of this step
DEPTH	NUMBER(38)		Depth
POSITION	NUMBER(38)		Order of processing for steps with the same parent ID
COST	NUMBER(38)		Cost of the current operation estimated by the cost-based optimizer (CBO)
CARDINALITY	NUMBER(38)		Number of rows returned by the current operation (estimated by the CBO)
BYTES	NUMBER(38)		Number of bytes returned by the current operation

Column	Datatype	NULL	Description
OTHER_TAG	VARCHAR2(255)		Describes the function of the SQL text in the OTHER column. Values for OTHER_TAG are: • SERIAL - SQL is the text of a locally-executed, serial query plan. Currently, SQL is not loaded in OTHER for this case. • SERIAL_FROM_REMOTE - SQL text shown in the OTHER column will be executed at a remote site • PARALLEL_COMBINED_WITH_PARENT - Parent of this operation is a DFO that performs both operations in the parallel execution plan • PARALLEL_COMBINED_WITH_CHILD - Child of this operation is a DFO that performs both operations in the parallel execution plan. • PARALLEL_TO_SERIAL - SQL text shown in the OTHER column is the top-level of the parallel plan. • PARALLEL_TO_PARALLEL - SQL text shown in the OTHER column is executed and output in parallel • PARALLEL_FROM_SERIAL - Operation consumes data from a serial operation and outputs it in parallel
PARTITION_START	VARCHAR2(255)		Start partition of a range of accessed partitions
PARTITION_STOP	VARCHAR2(255)		Stop partition of a range of accessed partitions
PARTITION_ID	NUMBER(38)		Step that has computed the pair of values of the PARTITION_START and PARTITION_STOP columns
OTHER	LONG		Information about parallel execution servers and parallel queries
DISTRIBUTION	VARCHAR2(30)		Distribution method
CPU_COST	NUMBER(38)		User-defined CPU cost
IO_COST	NUMBER(38)		User-defined I/O cost
TEMP_SPACE	NUMBER(38)		Temporary space usage of the operation (sort or hash-join) as estimated by the CBO
ACCESS_PREDICATES	VARCHAR2(4000)		Predicates used to locate rows in an access structure. For example, start or stop predicates for an index range scan.
FILTER_PREDICATES	VARCHAR2(4000)		Predicates used to filter rows before producing them
PROJECTION	VARCHAR2(4000)		Expressions produced by the operation
TIME	NUMBER(38)		Elapsed time (in seconds) of the operation as estimated by the CBO
QBLOCK_NAME	VARCHAR2(128)		Name of the query block

Column	Datatype	NULL	Description
OTHER_XML	CLOB		Provides extra information specific to an execution step of the execution plan. The content of this column is structured using XML, which allows multiple pieces of information to be stored, including the following: • Name of the schema against which the query was parsed • Release number of the Oracle Database that produced the explain plan • Hash value associated with the execution plan • Name (if any) of the outline or the SQL profile used to build the execution plan • Indication of whether or not dynamic statistics were used to produce the plan • The outline data, a set of optimizer hints that can be used to regenerate the same plan • Additional data that describes the relationship between rows in the plan table and subplans of adaptive plans

See Also:

"USER_SQLTUNE_PLANS"

6.442 DBA_SQLTUNE_RATIONALE_PLAN

DBA_SQLTUNE_RATIONALE_PLAN displays the association between rationales and operations in the execution plan of all SQL statements in the database.

Related View

USER_SQLTUNE_RATIONALE_PLAN displays the association between rationales and operations in the execution plan of the SQL statements owned by the current user.

Column	Datatype	NULL	Description
TASK_ID	NUMBER(38)	NOT NULL	Tuning task identifier
EXECUTION_NAME	VARCHAR2(128)	NOT NULL	The name of the task execution with which this entry (row) is associated
RATIONALE_ID	NUMBER(38)	NOT NULL	Rationale identifier
OBJECT_ID	NUMBER(38)	NOT NULL	Advisor framework object identifier
OPERATION_ID	NUMBER(38)	NOT NULL	Operation identifier
PLAN_ATTRIBUTE	VARCHAR2(34)		Type of the execution plan: • Original - Original plan of the query • Original with adjusted cost - Same as Original but with adjusted cost • Using SQL profile - Plan with SQL profile applied • Using new indices - Plan with indexes applied

> **See Also:**
>
> "USER_SQLTUNE_RATIONALE_PLAN"

6.443 DBA_SQLTUNE_STATISTICS

DBA_SQLTUNE_STATISTICS displays statistics associated with all SQL statements in the database.

Related View

USER_SQLTUNE_STATISTICS displays statistics associated with the SQL statements owned by the current user.

Column	Datatype	NULL	Description
TASK_ID	NUMBER(38)	NOT NULL	Tuning task identifier
OBJECT_ID	NUMBER(38)	NOT NULL	Advisor framework object identifier
PARSING_SCHEMA_ID	NUMBER		Schema under which the SQL is parsed
MODULE	VARCHAR2(64)		Last application module recorded for the SQL
ACTION	VARCHAR2(64)		Last application action recorded for the SQL
ELAPSED_TIME	NUMBER		Elapsed time for the SQL statement
CPU_TIME	NUMBER		CPU time for the SQL
BUFFER_GETS	NUMBER		Number of buffer gets
DISK_READS	NUMBER		Number of disk reads
DIRECT_WRITES	NUMBER		Number of disk writes
ROWS_PROCESSED	NUMBER		Number of rows processed by the SQL
FETCHES	NUMBER		Number of fetches
EXECUTIONS	NUMBER		Number of executions
END_OF_FETCH_COUNT	NUMBER		End of fetch count
OPTIMIZER_COST	NUMBER		Optimizer cost for the SQL
OPTIMIZER_ENV	RAW(2000)		Optimizer environment
COMMAND_TYPE	NUMBER		Command type

> **See Also:**
>
> "USER_SQLTUNE_STATISTICS"

6.444 DBA_SR_GRP_STATUS

DBA_SR_GRP_STATUS provides information on the current refresh operations for the current synchronous refresh groups in the database. It has the same columns as the DBA_SR_GRP_STATUS_ALL view.

Refresh operations are controlled using the DBMS_SYNC_REFRESH package.

Related View

USER_SR_GRP_STATUS provides information on the current refresh operations for the current synchronous refresh groups in the database which are owned by the current user. Its columns are the same as those in DBA_SR_GRP_STATUS.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)	NOT NULL	Owner of the refresh operation, which is the user who launched the operation
GROUP_ID	NUMBER	NOT NULL	Group ID of the synchronous refresh group
OPERATION	VARCHAR2(7)		The phase of the refresh operation performed: • PREPARE • EXECUTE
STATUS	VARCHAR2(10)		The status of the refresh operation: • RUNNING • NOT PROCESSED • COMPLETE • ERROR-SOFT • ERROR-HARD • ABORT • PARTIAL
NUM_TBLS	NUMBER	NOT NULL	The number of tables in the synchronous refresh group
NUM_MVS	NUMBER	NOT NULL	The number of materialized views in the synchronous refresh group
BASE_TBLS_REFR_STATUS	VARCHAR2(13)		Indicates the refresh status of base tables in the synchronous refresh group. The possible values are: • NOT PROCESSED • COMPLETE • ABORT
NUM_MVS_COMPLETED	NUMBER		The number of materialized views which have completed refresh in the synchronous refresh group
NUM_MVS_ABORTED	NUMBER		The number of materialized views which have terminated refresh in the synchronous refresh group
ERROR_NUMBER	NUMBER		Error number of the run (if any)
ERROR_MESSAGE	VARCHAR2(4000)		Error message of the run (if any)
PREPARE_START_TIME	DATE		Time that the PREPARE_REFRESH phase of the run started
PREPARE_END_TIME	DATE		Time that the PREPARE_REFRESH phase of the run ended
EXECUTE_START_TIME	DATE		Time that the EXECUTE_REFRESH phase of the run started

Column	Datatype	NULL	Description
EXECUTE_END_TIME	DATE		Time that the EXECUTE_REFRESH phase of the run ended

See Also:

- "USER_SR_GRP_STATUS"
- *Oracle Database PL/SQL Packages and Types Reference* for more information about the DBMS_SYNC_REFRESH package

6.445 DBA_SR_GRP_STATUS_ALL

DBA_SR_GRP_STATUS_ALL provides information on the refresh operations on the synchronous refresh groups in the database.

A refresh operation is also called a run, and it has two phases: PREPARE_REFRESH and EXECUTE_REFRESH. These phases are controlled using the DBMS_SYNC_REFRESH package.

Each row in this view provides information on a run of a group, identified by its GROUP_ID. The view contains information on the status of the objects of both current and past runs of both current and defunct synchronous refresh groups. Therefore, this view can be used to examine the history of synchronous refresh operations.

The current run of a group is the most recent run of a group; a current group is a currently valid group, which is capable of being refreshed. A group becomes defunct when it is unregistered for any reason, either explicitly by the user or implicitly as a side-effect when the user registers materialized views related to the materialized views in the group.

To view the status of refresh operations for the most recent runs of only the current groups, use the DBA_SR_GRP_STATUS view.

Related View

USER_SR_GRP_STATUS_ALL provides information on the refresh operations on the synchronous refresh groups in the database which are owned by the current user. Its columns are the same as those in DBA_SR_GRP_STATUS_ALL.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)	NOT NULL	Owner of the refresh operation, which is the user who launched the operation
GROUP_ID	NUMBER	NOT NULL	Group ID of the synchronous refresh group
OPERATION	VARCHAR2(7)		The phase of the refresh operation performed: • PREPARE • EXECUTE

Column	Datatype	NULL	Description
STATUS	VARCHAR2(10)		The status of the refresh operation: • RUNNING • NOT PROCESSED • COMPLETE • ERROR-SOFT • ERROR-HARD • ABORT • PARTIAL
CURRENT_RUN	VARCHAR2(1)		Indicates whether the record is for the most recent refresh-operation on the group: • Y - Yes • N - No
CURRENT_GROUP	VARCHAR2(1)		Indicates whether the record is for a current group: • Y - Yes • N - No
NUM_TBLS	NUMBER	NOT NULL	The number of tables in the synchronous refresh group
NUM_MVS	NUMBER	NOT NULL	The number of materialized views in the synchronous refresh group
BASE_TBLS_REFR_STATUS	VARCHAR2(13)		Indicates the refresh status of base tables in the synchronous refresh group. The possible values are: • NOT PROCESSED • COMPLETE • ABORT
NUM_MVS_COMPLETED	NUMBER		The number of materialized views which have completed refresh in the synchronous refresh group
NUM_MVS_ABORTED	NUMBER		The number of materialized views which have terminated refresh in the synchronous refresh group
ERROR_NUMBER	NUMBER		Error number of the run (if any)
ERROR_MESSAGE	VARCHAR2(4000)		Error message of the run (if any)
PREPARE_START_TIME	DATE		Time that the PREPARE_REFRESH phase of the run started
PREPARE_END_TIME	DATE		Time that the PREPARE_REFRESH phase of the run ended
EXECUTE_START_TIME	DATE		Time that the EXECUTE_REFRESH phase of the run started
EXECUTE_END_TIME	DATE		Time that the EXECUTE_REFRESH phase of the run ended

See Also:

- "USER_SR_GRP_STATUS_ALL"
- *Oracle Database PL/SQL Packages and Types Reference* for more information about the DBMS_SYNC_REFRESH package

6.446 DBA_SR_OBJ

DBA_SR_OBJ provides information on the objects registered for synchronous refresh for current groups.

Related View

USER_SR_OBJ provides information on the objects registered for synchronous refresh for current groups for the current user. Its columns are the same as those in DBA_SR_OBJ.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)	NOT NULL	Owner of the synchronous refresh object
NAME	VARCHAR2(128)	NOT NULL	Name of the synchronous refresh object
TYPE	VARCHAR2(5)		Type of synchronous refresh object: • MVIEW • TABLE
GROUP_ID	NUMBER	NOT NULL	Group ID of the synchronous refresh group to which this object belongs
STAGING_LOG_NAME	VARCHAR2(128)		Name of the staging log for tables. This column has a value of NULL for materialized views.

> **See Also:**
>
> "USER_SR_OBJ"

6.447 DBA_SR_OBJ_ALL

DBA_SR_OBJ_ALL provides information on the objects registered for synchronous refresh for current and defunct groups.

To see information on the objects registered for synchronous refresh for only the current groups, use the DBA_SR_OBJ view.

Related View

USER_SR_OBJ_ALL provides information on the objects registered for synchronous refresh for current and defunct groups for the current user. Its columns are the same as those in DBA_SR_OBJ_ALL.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)	NOT NULL	Owner of the synchronous refresh object
NAME	VARCHAR2(128)	NOT NULL	Name of the synchronous refresh object
TYPE	VARCHAR2(5)		Type of synchronous refresh object: • MVIEW • TABLE
GROUP_ID	NUMBER	NOT NULL	Group ID of the synchronous refresh group to which this object belongs

Column	Datatype	NULL	Description
CURRENT_GROUP	VARCHAR2(1)		Indicates whether the record is for a current group: • Y - Yes • N - No
STAGING_LOG_NAME	VARCHAR2(128)		Name of the staging log for tables. This column has a value of NULL for materialized views.

> **See Also:**
>
> "USER_SR_OBJ_ALL"

6.448 DBA_SR_OBJ_STATUS

DBA_SR_OBJ_STATUS provides information on the status of objects registered for synchronous refresh for the current refresh operations for the current synchronous refresh groups in the database.

To see information on the status of objects registered for synchronous refresh, use the DBA_SR_OBJ_STATUS_ALL view.

Related View

USER_SR_OBJ_STATUS provides information on the status of objects registered for synchronous refresh for the current refresh operations for the current synchronous refresh groups in the database which are owned by the current user. Its columns are the same as those in DBA_SR_OBJ_STATUS.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)	NOT NULL	Owner of the synchronous refresh object
NAME	VARCHAR2(128)	NOT NULL	Name of the synchronous refresh object
TYPE	VARCHAR2(5)		Type of synchronous refresh object: • MVIEW • TABLE
GROUP_ID	NUMBER	NOT NULL	Group ID of the synchronous refresh group to which this object belongs
STATUS	VARCHAR2(13)		Status of the synchronous refresh object: • NOT PROCESSED • COMPLETE • ABORT
ERROR_NUMBER	NUMBER		Error number of the run (if any)
ERROR_MESSAGE	VARCHAR2(4000)		Error message of the run (if any)
LAST_MODIFIED_TIME	DATE		Last modification time of the synchronous refresh object

> **See Also:**
>
> "USER_SR_OBJ_STATUS"

6.449 DBA_SR_OBJ_STATUS_ALL

DBA_SR_OBJ_STATUS_ALL provides information on the status of objects registered for synchronous refresh.

The view contains information on the status of the objects of both the current and past runs of both current and defunct groups. Therefore, this view can be used to examine the history of synchronous refresh operations. The current run of a group is the most recent run of a group; a current group is a currently valid group, which is capable of being refreshed. A group becomes defunct when it is unregistered for any reason, either explicitly by the user or implicitly as a side-effect when the user registers materialized views related to the materialized views in the group.

To see information on the status of refresh operations for the most recent runs of only the current groups, use the DBA_SR_OBJ_STATUS view.

Related View

USER_SR_OBJ_STATUS_ALL provides information on the status of objects registered for synchronous refresh in the database which are owned by the current user. Its columns are the same as those in DBA_SR_OBJ_STATUS_ALL.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)	NOT NULL	Owner of the synchronous refresh object
NAME	VARCHAR2(128)	NOT NULL	Name of the synchronous refresh object
TYPE	VARCHAR2(5)		Type of synchronous refresh object: • MVIEW • TABLE
GROUP_ID	NUMBER	NOT NULL	Group ID of the synchronous refresh group to which this object belongs
STATUS	VARCHAR2(13)		Status of the synchronous refresh object: • NOT PROCESSED • COMPLETE • ABORT
CURRENT_RUN	VARCHAR2(1)		Indicates whether the record is for the most recent refresh operation on the group: • Y - Yes • N - No
CURRENT_GROUP	VARCHAR2(1)		Indicates whether the record is for a current group: • Y - Yes • N - No
ERROR_NUMBER	NUMBER		Error number of the run (if any)
ERROR_MESSAGE	VARCHAR2(4000)		Error message of the run (if any)
LAST_MODIFIED_TIME	DATE		Last modification time of the synchronous refresh object

> **See Also:**
>
> "USER_SR_OBJ_STATUS_ALL"

6.450 DBA_SR_PARTN_OPS

DBA_SR_PARTN_OPS provides information on the partition operations registered on the base tables of the materialized views registered for synchronous refresh.

These rows last only as long as the registrations are active; that is, they disappear after EXECUTE_REFRESH or ABORT_REFRESH of the base table by the DBMS_SYNC_REFRESH package.

Related View

USER_SR_PARTN_OPS provides information on the partition operations registered on the base tables of the materialized views registered for synchronous refresh belonging to the current user. Its columns are the same as those in DBA_SR_PARTN_OPS.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)	NOT NULL	Owner of the base table registered for synchronous refresh
TABLE_NAME	VARCHAR2(128)	NOT NULL	Name of the table
PARTITION_OP	VARCHAR2(128)	NOT NULL	Type of partition operation: • DROP • EXCHANGE • TRUNCATE
PARTITION_NAME	VARCHAR2(128)	NOT NULL	Name of the partition to be changed
OUTSIDE_TABLE_SCHEMA	VARCHAR2(128)		Schema in which the outside table (for EXCHANGE PARTITION) was created
OUTSIDE_TABLE_NAME	VARCHAR2(128)		Name of the outside table (for EXCHANGE PARTITION)

> **See Also:**
>
> - "USER_SR_PARTN_OPS"
> - *Oracle Database PL/SQL Packages and Types Reference* for more information about the DBMS_SYNC_REFRESH package

6.451 DBA_SR_STLOG_EXCEPTIONS

DBA_SR_STLOG_EXCEPTIONS provides information on the exceptions in the staging logs for the tables processed by DBMS_SYNC_REFRESH.PREPARE_STAGING_LOG.

Related View

USER_SR_STLOG_EXCEPTIONS provides information on the exceptions in the staging logs for the tables belonging to the current user processed by DBMS_SYNC_REFRESH.PREPARE_STAGING_LOG. Its columns are the same as those in DBA_SR_STLOG_EXCEPTIONS.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)	NOT NULL	Owner of the base table registered for synchronous refresh
TABLE_NAME	VARCHAR2(128)	NOT NULL	Name of the base table registered for synchronous refresh
STAGING_LOG_NAME	VARCHAR2(128)	NOT NULL	Name of the staging log for tables. This column has a value of NULL for materialized views.
BAD_ROWID	ROWID	NOT NULL	Row ID of the staging log row causing the exception for the synchronous refresh
ERROR_NUMBER	NUMBER		Error number of the exception for the synchronous refresh
ERROR_MESSAGE	VARCHAR2(4000)		Error message associated with the ERROR_NUMBER for the synchronous refresh

See Also:

- "USER_SR_STLOG_EXCEPTIONS"
- *Oracle Database PL/SQL Packages and Types Reference* for more information about the DBMS_SYNC_REFRESH package

6.452 DBA_SR_STLOG_STATS

DBA_SR_STLOG_STATS provides information on the statistics in the staging logs for the tables processed by DBMS_SYNC_REFRESH.PREPARE_STAGING_LOG.

These three statistics columns in the staging log are filled in PREPARE_STAGING_LOG:

- The number of inserts (NUM_INSERTS)
- The number of deletes (NUM_DELETES)
- The number of updates (NUM_UPDATES)

After the data in the staging logs of a synchronous refresh group have been processed by PREPARE_REFRESH and EXECUTE_REFRESH, the statistics columns for the tables in the group are cleared and appear as NULL.

Related View

USER_SR_STLOG_STATS provides information on the statistics in the staging logs for the tables belonging to the current user processed by DBMS_SYNC_REFRESH.PREPARE_STAGING_LOG.

Column	Datatype	NULL	Description
OWNER	VARCHAR2(128)	NOT NULL	Owner of the base table registered for synchronous refresh
TABLE_NAME	VARCHAR2(128)	NOT NULL	Name of the table
STAGING_LOG_NAME	VARCHAR2(128)	NOT NULL	Name of the staging log for tables. NULL for materialized views
NUM_INSERTS	NUMBER	NOT NULL	The number of inserts in the staging log
NUM_DELETES	NUMBER	NOT NULL	The number of deletes in the staging log
NUM_UPDATES	NUMBER	NOT NULL	The number of updates in the staging log
PSL_MODE	VARCHAR2(33)		The mode specified by the user in running DBMS_SYNC_REFRESH.PREPARE_STAGING_LOG. Possible values: • DELETE_TRUSTED • DELETE_TRUSTED and UPDATE_TRUSTED • ENFORCED • INSERT_TRUSTED • INSERT_TRUSTED and DELETE_TRUSTED • TRUSTED • UPDATE_TRUSTED • UPDATE_TRUSTED and INSERT_TRUSTED

See Also:

- "USER_SR_STLOG_STATS"
- *Oracle Database PL/SQL Packages and Types Reference* for more information about the DBMS_SYNC_REFRESH package

6.453 DBA_SSCR_CAPTURE

DBA_SSCR_CAPTURE displays session state capture statistics.

Column	Datatype	NULL	Description
DB_NAME	VARCHAR2(4000)		Database name of captured session
INST_NAME	VARCHAR2(4000)		Instance name of captured session
INST_ID	NUMBER		Instance ID of captured session
SESSION_ID	NUMBER		Session ID of captured session
SESSION_SERIAL#	NUMBER		Session serial number of captured session
USER_NAME	VARCHAR2(128)	NOT NULL	User name of captured session
SCHEMA_NAME	VARCHAR2(128)	NOT NULL	Schema name of captured session
SEQUENCE#	NUMBER		Sequence number of captured session

Column	Datatype	NULL	Description
CAPTURE_MODE	VARCHAR2(7)		Mode of capture operation
CAPTURE_SCOPE	VARCHAR2(7)		Scope of capture operation
CAPTURE_FORMAT	VARCHAR2(9)		Format of capture files
CAPTURE_DIR	VARCHAR2(128)		Directory object of capture files
CAPTURE_LOCATOR	RAW(64)		Locator of primary capture file
CAPTURE_TIME	TIMESTAMP(6)		Timestamp of capture operation

6.454 DBA_SSCR_RESTORE

DBA_SSCR_RESTORE displays session state restore statistics.

Column	Datatype	NULL	Description
DB_NAME	VARCHAR2(4000)		Database name of restored session
INST_NAME	VARCHAR2(4000)		Instance name of restored session
INST_ID	NUMBER		Instance ID of restored session
SESSION_ID	NUMBER		Session ID of restored session
SESSION_SERIAL#	NUMBER		Session serial number of restored session
USER_NAME	VARCHAR2(128)	NOT NULL	User name of restored session
SCHEMA_NAME	VARCHAR2(128)	NOT NULL	Schema name of restored session
SEQUENCE#	NUMBER		Sequence number of restore operation
RESTORE_MODE	VARCHAR2(7)		Mode of restore operation
RESTORE_SCOPE	VARCHAR2(7)		Scope of restore operation
RESTORE_FORMAT	VARCHAR2(9)		Format of restore files
RESTORE_DIR	VARCHAR2(128)		Directory object of restore files
RESTORE_LOCATOR	RAW(64)		Locator of primary restore file
RESTORE_TIME	TIMESTAMP(6)		Timestamp of restore operation

6.455 DBA_STAT_EXTENSIONS

DBA_STAT_EXTENSIONS displays information about all optimizer statistics extensions in the database.

> **See Also:**
>
> "ALL_STAT_EXTENSIONS"

6.456 DBA_STATEMENTS

DBA_STATEMENTS Statements in stored objects accessible to sys. Its columns are the same as those in ALL_STATEMENTS.

> **See Also:**
>
> "ALL_STATEMENTS"

6.457 DBA_STMT_AUDIT_OPTS

DBA_STMT_AUDIT_OPTS describes current system auditing options across the system and by user.

> **Note:**
>
> This view is deprecated and applies only to traditional auditing. Traditional auditing is desupported starting in Oracle Database 23ai. Though traditional auditing is desupported, any current traditional audit settings that you have will still be honored and are viewable with this view. See *Oracle Database Security Guide* for more information about how this desupport works.

Column	Datatype	NULL	Description
USER_NAME	VARCHAR2(128)		User name if by user auditing; ANY CLIENT if access by a proxy on behalf of a client is being audited; NULL for system-wide auditing
PROXY_NAME	VARCHAR2(128)		Name of the proxy user which is performing an operation for the client; NULL if the client is performing the operation directly
AUDIT_OPTION	VARCHAR2(40)	NOT NULL	Name of the system auditing option
SUCCESS	VARCHAR2(10)		Mode for WHENEVER SUCCESSFUL system auditing
FAILURE	VARCHAR2(10)		Mode for WHENEVER NOT SUCCESSFUL system auditing

6.458 DBA_STORED_SETTINGS

DBA_STORED_SETTINGS lists information about the persistent parameter settings for stored PL/SQL units for which the current user has execute privileges.

It also returns parameter information for all objects in the database and is accessible only to users with the SELECT_CATALOG_ROLE privilege. Its columns are the same as those in ALL_STORED_SETTINGS.

See Also:

"ALL_STORED_SETTINGS"